# FET-LM: Flow-Enhanced Variational Autoencoder for Topic-Guided Language Modeling

Haoqin Tu, Zhongliang Yang, Jinshuai Yang, Linna Zhou, and Yongfeng Huang, *Senior Member, IEEE*

***Abstract*— Variational autoencoder (VAE) is widely used in tasks of unsupervised text generation due to its potential of deriving meaningful latent spaces, which, however, often assumes that the distribution of texts follows a common yet poor-expressed isotropic Gaussian. In real-life scenarios, sentences with different semantics may not follow simple isotropic Gaussian. Instead, they are very likely to follow a more intricate and diverse distribution due to the inconformity of different topics in texts. Considering this, we propose a flow-enhanced VAE for topic-guided language modeling (FET-LM). The proposed FET-LM models topic and sequence latent separately, and it adopts a normalized flow composed of householder transformations for sequence posterior modeling, which can better approximate complex text distributions. FET-LM further leverages a neural latent topic component by considering learned sequence knowledge, which not only eases the burden of learning topic without supervision but also guides the sequence component to coalesce topic information during training. To make the generated texts more correlative to topics, we additionally assign the topic encoder to play the role of a discriminator. Encouraging results on abundant automatic metrics and three generation tasks demonstrate that the FET-LM not only learns interpretable sequence and topic representations but also is fully capable of generating high-quality paragraphs that are semantically consistent.**

***Index Terms*— Controllable generation, normalizing flow, text generation, topic modeling, variational autoencoder (VAE).**

## Nomenclature

| | |
|---|---|
| $\boldsymbol{X}$ | Unlabeled text training document. |
| $\hat{\boldsymbol{X}}$ | Topic word corpus output. |
| $\boldsymbol{Y}$ | Reconstructed text document. |
| $x_i$ | $i$th word from $\boldsymbol{X}$. |
| $\hat{x}_i$ | $i$th topic word from $\hat{\boldsymbol{X}}$. |
| $y_i$ | $i$th word from $\boldsymbol{Y}$. |
| $\boldsymbol{z}$ | Latent variable of variational autoencoder. |
| $\boldsymbol{z_t}$ | Topic latent variable of FET-LM. |
| $T$ | Latent dimension of $\boldsymbol{z_t}$. |
| $\boldsymbol{z_s}$ | Sequence latent variable of FET-LM. |
| $\mathcal{D}_i$ | $i$th transformed distribution in the flow. |
| $f_i$ | $i$th flow transformation. |
| $\boldsymbol{v}$ | Householder vector. |
| $H$ | Householder transformation matrix. |
| $f_{h(i)}$ | $i$th householder flow (HF) transform. |
| $\boldsymbol{z}_{s(i)}$ | $i$th sequence latent variable in HF. |
| $\boldsymbol{z}_i$ | $i$th latent variable in the flow. |
| $\boldsymbol{h_i}$ | $i$th hidden state of the decoder. |
| $d$ | BoW representation from topic encoder. |
| $c$ | Vocabulary size of training corpus. |
| $b_i$ | $i$th word representation in discriminator. |
| $\beta_i$ | $i$th topic word output probability. |
| $p_\theta(\cdot)$ | Prior parameterized by $\theta$. |
| $q_\phi(\cdot)$ | Posterior parameterized by $\phi$. |

## I. Introduction

As DEEP learning methods are gradually introduced to resolve language modeling problems, language model (LM) is becoming a key constituent of various natural language processing (NLP) tasks, such as machine translation [1], [2], automatic text summarization [3], and dialogue system [4], [5]. Text generation as an elementary task in NLP aims to generate authentic and plausible textual content that is realistic-looking [6]. Natural language generation (NLG) is an inherently complex task, which requires abundant linguistic and domain knowledge at multiple levels, including syntax, semantics, morphology, phonology, and pragmatics. In real life, it is easy for us to realize that textual contexts carry different meanings for different audiences. Therefore, the automatically generated texts should be tailored to their specific audiences in terms of appropriateness of content and terminology use [7], as well as for customized network environment and transparency reasons [8]. The goal of controllable text generation aims at generating coherent and grammatically correct texts whose attributes can be controlled and/or abide by user-defined rules, which reflects the particular interests of system users [9]. The attributes to control range from being stylistic such as politeness, sentiment, and formality; demographic attributes of the person writing the text such as gender and age; content such as information, keywords, and entities; and ordering of information, events, such as plot summaries.

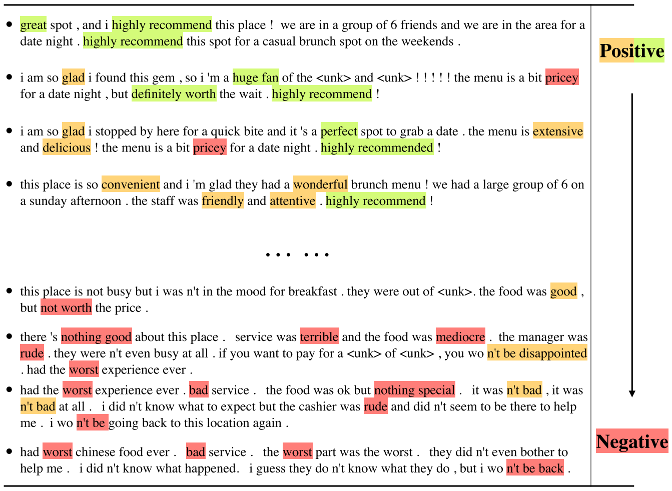
Manuscript received 27 July 2021; revised 6 November 2022 and 13 January 2023; accepted 13 February 2023. This work was supported in part by the National Key Research and Development Program of China under Grant 2022YFC3302100 and in part by the National Natural Science Foundation of China under Grant U1936216 and Grant 61862002. *(Corresponding author: Zhongliang Yang.)*

Haoqin Tu and Jinshuai Yang are with the Department of Electronic Engineering, Tsinghua University, Beijing 100084, China (e-mail: tuisaac163@gmail.com).

Zhongliang Yang and Linna Zhou are with the School of Cyberspace Security, Beijing University of Posts and Telecommunications, Beijing 100876, China (e-mail: yangzl@bupt.edu.cn).

Yongfeng Huang is with the Zhongguancun Laboratory, Beijing 100094, China, and also with the Department of Electronic Engineering, Tsinghua University, Beijing 100084, China (e-mail: yfhuang@mail.tsinghua.edu.cn).

Digital Object Identifier 10.1109/TNNLS.2023.3249253



- great spot , and i highly recommend this place ! we are in a group of 6 friends and we are in the area for a date night . highly recommend this spot for a casual brunch spot on the weekends .
- i am so glad i found this gem , so i 'm a huge fan of the <unk> and <unk> ! ! ! ! ! the menu is a bit pricey for a date night , but definitely worth the wait . highly recommend !
- i am so glad i stopped by here for a quick bite and it 's a perfect spot to grab a date . the menu is extensive and delicious ! the menu is a bit pricey for a date night . highly recommended !
- this place is so convenient and i 'm glad they had a wonderful brunch menu ! we had a large group of 6 on a sunday afternoon . the staff was friendly and attentive . highly recommend !

… …

- this place is not busy but i was n't in the mood for breakfast . they were out of <unk>. the food was good , but not worth the price .
- there 's nothing good about this place . service was terrible and the food was mediocre . the manager was rude . they were n't even busy at all . if you want to pay for a <unk> of <unk> , you wo n't be disappointed . had the worst experience ever .
- had the worst experience ever . bad service . the food was ok but nothing special . it was n't bad , it was n't bad at all . i did n't know what to expect but the cashier was rude and did n't seem to be there to help me . i wo n't be going back to this location again .
- had worst chinese food ever . bad service . the worst part was the worst . they did n't even bother to help me . i did n't know what happened. i guess they do n't know what they do , but i wo n't be back .

Fig. 1. Proposed FET-LM can learn different topic representations in an unsupervised manner. We present an example on text style transfer task. Contiguously generated texts share a similar structure while showing different sentiments. For instance, there is a changing trend of text sentiment from very positive (light green) or positive (orange) to negative (red).

We define the task of controllable text generation as finding a function $f$ to generate sentences that obey certain generation rules or conditions. This can be formally defined as follows: given a set of $n$ conditions $\mathbf{C} = \{\mathbf{c}_i\}_1^n \in \mathcal{C}$, where $\mathcal{C}$ denotes the condition space. The goal of controllable generation is formalized as learning a function $f$ such that $f(\mathbf{C}) = \mathbf{Z}, \mathbf{Z} \in \mathcal{Z}$. In general, the controlled sentence generation task can be divided into two strategies according to the way in which restrictions are imposed: generation with soft constraint and hard constraint. First, soft constraint text generation requires the generated sentences to be semantically similar to the given constraints (e.g., topic and style), rather than explicitly enforcing certain concepts or rules to appear in the contents. The mapping function $f$ mentioned above serves as a measurement to find sentences with the highest semantic similarity with given constraints [10], [11], [12], [13], [14], [15], [16], [17]. Second, hard constraint focuses on controlling specific tokens or textual structures (e.g., keywords and sentence length) during generation and thus is more fine-grained compared with the soft one. It indicates the compulsive inclusion of given constraints in the output. Hence, the function $f$ here is a binary sign on a specified controlling level (e.g., token and syntax) to eliminate the possibility of producing unqualified features on such level [18], [19], [20].

However, generating text under specific lexical constraints is challenging [21]. While hard constraint generative models handle given conditions with higher proficiency by placing explicit restrictions on independent attribute controls, they have difficulty in dealing with several issues, such as unitary syntax, semantical inconsistency [18], [20], [22], as well as excessively rigorous model architectures [9]. The other way around, soft constraint generation can not only produce authentic texts with certain attributes but also largely benefits downstream tasks (text summarization [13], style transfer [11], [14], and so on) from its ability to capture explainable text representation effectively.

In the past few years, a large number of researchers have tried to use different methods for controlled text generation with soft constraints. Intuitively, the target of producing topic-specified sentences can fall into three courses: topic extraction, text sequence learning, and joint generation. Therefore, both topic and sequence models are of great importance in analyzing and creating controllable texts. Compared with other approaches to produce textual contents, such as those based on generative adversarial networks (GANs) [23] or plain recurrent neural networks (RNNs) [24], variational autoencoder (VAE) is suitable for text generation with implicit constraints because its flexible latent spaces capture integral properties of inputs, such as content styles and high-level linguistic or semantic features, being beneficial for controllable generation [25]. Besides, the latent knowledge that originates from a VAE can help mitigate against model misspecification [26], can derive beneficial hidden knowledge for various domains [27], [28], and can also enable interesting structures to emerge [12], [29].

However, other problems arise in practice that may limit the modeling capacity and empirical performance of VAE-based models. KL collapse is one of the major challenges that are widely concerned [30]. Various approaches have been devised to handle this issue, including optimizing decoder architectures [31], [32], inventing auxiliary objectives [10], [33], [34], novel encoder–decoder training schedule [35], [36], and flexible latent code posterior [13], [37]. These methods generally share the same goal: to impair the ability of a powerful recurrent decoder and strengthen the expression of latent space. The second issue associated with a VAE model to generate topic-specified texts is rooted in the assumption of its variational distribution, which usually accepts a spherical Gaussian with diagonal covariance matrix. This leads to the following.

1) *Latent Constraint for the Plain Text VAEs:* The true posterior of the VAE can only be well approximated by variational inference when it is in the exact same family as the assumed one [38].
2) *Latent Vacancy Dilemma [39] for Controllable Generation:* A text VAE (textVAE) with one monopolistic latent space is notoriously unsuitable for direct controllable generation because of the deficiency in its latent presentation.

To address such plight, external help from more than one continuous latent space in VAE was considered [10], but its training schedule cannot be regarded as end-to-end. As a fixup, methods that extract both text syntax and topic information simultaneously were proposed [11], [40], but they suffered from an oversimplified representation in sequence component for analogous samples (i.e., isotropic Gaussian). Flexible latent modeling had also attracted attention [13], [34], whereas they confused the text syntax knowledge and topic information in a unified latent space, which makes the models less interpretable.

These methods ignore the nature that topic-specified sentences are not analogous and, thus, their representations are unlike to fit in an isotropic space, and may confuse topic and sequence modeling in a holistic continuous space, which makes them suffer from poor interpretability and mode collapse issues for controllable generation.

To tackle these puzzles, we propose flow-enhanced VAE for topic-guided language modeling (FET-LM) in this article. FET-LM essentially consists of a topic modeling part and a sequence modeling part, which equip their own continuous latent spaces and are both optimized based on VAE. In detail, FET-LM discards the spherical Gaussian assumption of latent sequence component and models its distribution with a more flexible Gaussian using the householder flow (HF). In order to maximize the utilization of such powerful sequence latent, we also propose to condition the topic latent space on expressive learned sequential information, which acts like a prophet in the topic learning process and brings progress on both language and topic learning stages. Moreover, we adjust the topic encoder as a discriminator to augment the topic expression in sentences explicitly. Through manipulating the value of latent variables, our model is also able to produce textual content in progressively altered sentiments, as shown in Fig. 1. The sentiment of generated sentences shows a tendency from positive to negative, while these texts remain the analogical linguistic structure between neighboring manipulations, demonstrating that FET-LM is well designed for textual representation understanding and unsupervised generation.

*Contributions:* First, we present FET-LM, a novel approach to document topic modeling and controllable text generation based on the VAE model. Second, we clearly separate the topic modeling and text generation process of the model and propose to condition the topic latent on flexible sequence latent distribution parameterized by HF. Third, we adapt a topic discriminator term to regularize topic learning and further verify its effectiveness in multitasks. Fourth, the effectiveness of FET-LM is validated by consistently remarkable results on language and topic modeling, classification, and three text generation tasks. Our model reaches the state-of-the-art performance on text perplexity for better quality of output content and the topic latent classification accuracy for higher interpretability of topic learning.

## II. Backgrounds

In this section, we first share a brief insight into the inference and training of the latent variable model (LVM), the fashion in which FET-LM is constructed. We then go over the generative process of normalizing flows [41] for modeling an arbitrarily complicated distribution. Finally, we review related works of unsupervised controllable generative methods.

### A. Variational Inference and Training

In the realm of expectation–maximization (EM) algorithms, the maximal likelihood estimation (MLE) is of vital importance, which aims at minimizing the average negative log loss (NLL) of data $\boldsymbol{X}$ parameterized by $\theta$

$$\min_{\boldsymbol{\theta}\in\Theta} \frac{1}{n}\sum_{i=1}^{n} -\log p_{\boldsymbol{\theta}}(x_i) \tag{1}$$

where $\boldsymbol{X} = [x_1, x_2, \ldots, x_n]$ is described as a set of training data with length $n$. However, this probability calculation of $p_{\boldsymbol{\theta}}(x_i)$ is notoriously intractable and also cannot be differentiated directly. EM algorithms bring an estimation stage to settle this problem to some extent. In practice, variational inference introduces a latent variable $z$ and uses the parametric inference distribution (or posterior distribution) $q_\phi(z \mid \boldsymbol{X})$ to update the intractable likelihood term. Concretely speaking, the latent variable $z$ is contributed by optimizing the evidence lower bound (ELBO), which takes both reconstruction loss and a regularization loss implemented by the Kullback–Leibler divergence (KLD) into account

$$\log P_\theta(\boldsymbol{X}) \geq \underbrace{\mathbb{E}_{q_\phi(z|\boldsymbol{X})}\big[\log p_\theta(\boldsymbol{X} \mid z)\big]}_{\text{reconstruction term}} - \underbrace{\mathbb{D}_{\mathrm{KL}}\big(q_\phi(z \mid \boldsymbol{X}) \| p_\theta(z)\big)}_{\text{regularization term}}. \tag{2}$$

This objective directly optimizes the continuous latent space of VAE, helping the latent variable learn meaningful linguistic representations and further making it favorable to conduct controllable generation tasks.

### B. Generative Models With Flow

A well-expressive latent variable $z$ is essential to decouple different but somehow related topics in texts. As a result, in order to model all the complexities of sequences with various topics, the latent posterior of text representations $q_\phi(z \mid \boldsymbol{X})$ will necessarily be complex.

A normalizing flow [41] is able to transform a simple distribution (e.g., Gaussian) to a relatively complex one by a chain of invertible functions. Formally, given a simple distribution $\mathcal{D}_0$ and a variable $\boldsymbol{z_0}$ drawn from it, our goal is to find a complex distribution $\mathcal{D}_K$ by sampling a concrete $\boldsymbol{z_K}$ from it. We then define an invertible transformation $f(\cdot)$ whose scope and range are $\mathcal{D}_0$ and $\mathcal{D}_k$, respectively: $\boldsymbol{z_0} \sim \mathcal{D}_0, \boldsymbol{z_K} = f(\boldsymbol{z_0})$, where the bijection function $f(\cdot)$ can be decomposed as a set of bijection functions $\{f_k\}_{k=1}^K$ of the same kind. By stacking them into a chain and acting on $\boldsymbol{z_0}$, altogether, they play the same role as $f(\cdot)$ does. We can call it a normalizing flow on distribution $\mathcal{D}_0$.

The essence of the flow-based generative process is the constant change of the input's coordinate system. Hence, we only need a Jacobian determinant to be multiplied to every point from the distribution $\mathcal{D}_0$ to distribution $\mathcal{D}_K$

$$\mathcal{D}_k = \mathcal{D}_0 \left|\det \frac{\partial \boldsymbol{f}}{\partial \boldsymbol{z_0}}\right| \tag{3}$$

and the general formula for the $k$th transformation is the absolute determinant of Jacobian matrix at that step: $|\det(\partial \boldsymbol{f_k}/\partial \boldsymbol{z_{k-1}})|$. As we specify that the generative model follows the paradigm of a VAE, the ELBO of a VAE-based generative model derived previously in (2) additionally requires a sum of the absolute determinant of the Jacobian matrix, that is:

$$\log P_\theta(\boldsymbol{X}) \geq \mathbb{E}_{q_\phi(\boldsymbol{z_K}|\boldsymbol{X})}\big[\log(p_\theta(\boldsymbol{X} \mid \boldsymbol{z_K}))\big] - \mathbb{D}_{\mathrm{KL}}(q_\phi(\boldsymbol{z_0} \mid \boldsymbol{X}) \| p_\theta(\boldsymbol{z_K})) + \sum_{k=1}^{K} \log\left|\det \frac{\partial \boldsymbol{f_k}}{\partial \boldsymbol{z_{k-1}}}\right| \tag{4}$$

and the original latent code $z$ is substituted by $z_K$ here, which is more competent to approximate the true distribution of data.

### C. Related Work

The objectives of self-supervised models with a soft constraint can be listed from three aspects: topic representation extraction, text syntax learning, and integrative generation. For the first target, learning topic information in sentences aims at finding a low-dimensional representation, which consists of topic explanatory and generative factors of the observed texts. Effective topic models, such as latent Dirichlet allocation (LDA) [42] as well as its nonparametric Bayesian generalizations [43], are quite appropriate for extracting topics from document-level texts and then map them to a latent space. Their modeling power has been further boosted by bringing in multilayer deep neural networks [44]. These methods typically ignore words' sequential orders [45] and feed texts in the bag-of-word (BoW) manner. Unlike previous methods, Wang et al. [46] proposed a customized convolutional operator and probabilistic pooling for topic modeling, which takes word order into consideration and resoundingly catches topic knowledge as well as local words dependencies. However, their model has difficulty in capturing reasonable text sequence information and producing realistic textual content.

When digging into the sequence modeling method, a big question associated with VAE-based controllable LMs is how to alleviate KLD collapse problem [30] and meanwhile integrate learned semantic information with proper syntax rules to generate plausible texts. KL vanishing problem is caused by the strong and obligate autoregressive network for text generation, which has become an important open challenge in the NLP field. There mainly exist two kinds of solutions to this problem. The first kind tackles this problem mostly by modifying model architectures to weaken the context modeling ability of decoders, for instance, word dropout trick before feeding to the decoder [30] and nonautoregressive networks (e.g., convolutional neural networks) as the decoder [47], [48]. The second category is to modify the loss functions of VAE-based LMs, for example, various KL annealings to fully leverage the latent information [30], [35], auxiliary loss terms to compensate the KL vanishing [10], [33], [49], or improved KL distance metrics for network optimization [28], [50]. The main idea behind all these methods is the same, i.e., to force the models to be less dependent on autoregressive RNNs so as to make latent information weights more on balancing sentence features. Furthermore, incorporating topic meanings with the component for syntax modeling has been greatly explored in recent years. Das et al. [51] put forward a Gauss-based topic model and assumed that each word was generated from a Gaussian distribution. Following this thought, Xiao et al. [10] employed a similar topic module but with Dirichlet distribution. Despite their success, these learning algorithms require multistage sampling or inference, so they cannot be counted as an end-to-end mode. Wang et al. [13] proposed a series of VAE works [12], [13], in which they used either mixture-of-experts or flow-based decoder for text distribution modeling. However, they mixed sequence and topic representation at the model input, making the unsupervised models unclear to explain. Also, similar model structures are also observed for controllable generation in various domains [28], [29]. As a remedy, Tang et al. [11] proposed to adopt topic and sequence models that followed multinominal Gaussian, and produced controllable word sequences by concatenating latent codes. Nevertheless, there are some drawbacks to these methods. For example, the restricted inference assumption in previous approaches put the learning process of texts with different topics on an equal footing, which is illogical for topic-specified text modeling. They trained both components from the scratch, which increased the difficulty for topic module to learn the semantic messages. Rezaee and Ferraro [17] came up with a novel variational topic LM. They first masked word embedding to label word semantics discretely and then constructed a conditional LM to generate controllable texts. Despite its refined architecture, the statistical results were not fairly satisfactory. Most lately, Dai et al. [34] made the latent space of VAE as a complex Riemannian manifold with learnable prior and posterior to enhance VAE's expression capability.

The model proposed in this article is different from the existing works. We explicitly split FET-LM into topic and sequence modeling sections with latent conditionality. We adopt HF to depict the complex distribution of texts with certain topics. Besides, our method leverages a discriminator with BoW input, which avoids a latent code collapse problem and heightens the overall model capacity. Finally, all elements in FET-LM can be trained end-to-end.

## III. FET-LM Methodology

FET-LM essentially consists of a topic modeling component and a sequence modeling component. The topic modeling part intends to learn interpretable latent codes of topics, while the sequential strategy is built for both modeling plausible sequence knowledge and composing learned topic latent into the generative process. The model structure is shown in Fig. 2 and its corresponding graphic is shown in Fig. 3(b).

### A. Topic Modeling Component

Similar to previous works on topic models, we transform discrete sentences into a BoW representation in the first place. We define $c$ to be the vocabulary size and $d \in \mathbb{Z}_{+}^{c}$ as the BoW representation of a document $\boldsymbol{X} = [x_1, x_2, \ldots, x_n]$ with length $n$, which indicates that every document has $c$ elements with nonnegative count. We assume that there are $T$ potential topics in given documents and introduce the topic latent variable $z_t$ following a $T$-dimensional Dirichlet distribution. Resemble in LDA, each dimension of $z_t$ hypothetically represents one topic. Intuitively, learning topic information from scratch is much harder than foreseeing some knowledge about the given document. As a result, we leak the posterior information of $z_s$ from the sequence component to the topic model in order to generate $z_t$, which will be described in detail in Section III-B. The overall process above is depicted by the upper part of Fig. 2. Concretely, for the conditional prior modeling of $z_t$, we have the following.

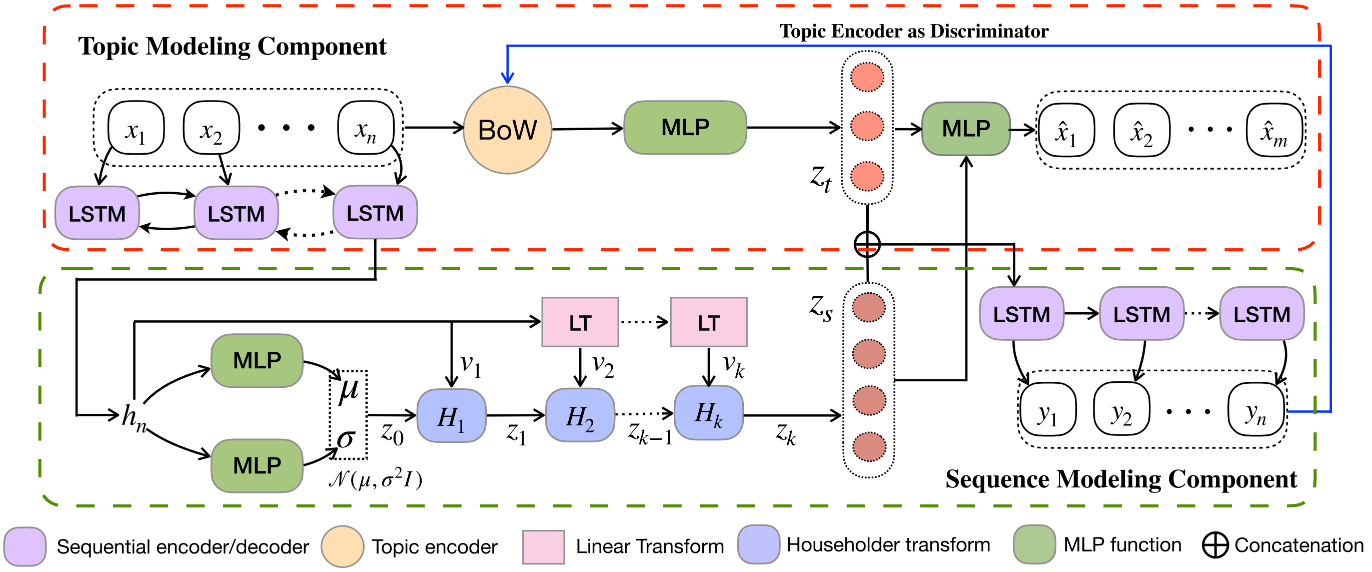


Fig. 2. Detailed explanation of the proposed FET-LM model. The overall architecture observes the encoder–decoder framework, which leverages two separate VAE models for topic and sequence modeling with the flow module and discriminator loss. These settings are in favor of learning topic information well-grounded and producing controllable texts with high qualities at the same time.

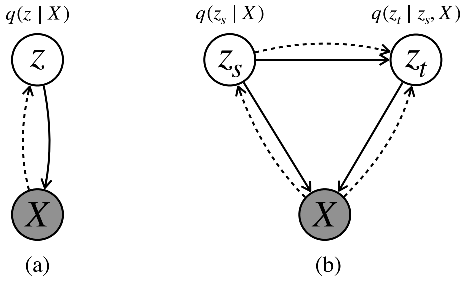


Fig. 3. Graphic model of (a) VAE and (b) FET-LM. Observed variables are in gray, while unseen variables are in white. Solid lines represent the inference process, and dashed lines work during the training process. The plain VAE equips one continuous hidden space $z$, while FET-LM separates topic and sequence latent spaces as $z_t, z_s$ with conditional assumption between them.

1) Draw $z_s$ from the sequence prior: $z_s \sim \mathcal{N}(0, I)$.
2) Draw topic prior conditioned on $z_s$ as $z_t \sim p(z_t \mid z_s)$.

Also, accordingly, the posterior of $z_t$ is parameterized as follows.

1) Draw the sequence posterior $z_s \sim q(z_s \mid \boldsymbol{X})$.
2) Draw the topic posterior $z_t \sim q(z_t \mid \boldsymbol{X}, z_s)$.

Then, the generative process of our topic part can be accomplished via the output probability of each word token, which can be specified as drawing $z_t$ from its learned distribution and then generates the output probability of topic words from topic decoder $\text{Dec}(\cdot)$: $[p(\hat{x}_1), \ldots, p(\hat{x}_m)] = \text{Dec}(z_t)$. In detail, $z_t$ follows Dirichlet and is modeled as $\text{Dir}(\text{BN}(\exp(z_s)))$, where $z_s \sim q(z_s \mid \boldsymbol{X})$, BN and Dir are the batch normalization and Dirichlet function, respectively, and $\exp(\cdot)$ is an exponential function to maintain the nonnegativity of the input Dirichlet center. Topic decoder Dec is built as linear layers: $\text{Dec}(z_t) = \text{Softmax}[\text{BN}(W_{z_t} z_t + b_{z_t})]$, $\text{Softmax}(\cdot)$ represents Softmax function, and $W_{z_t}$ and $b_{z_t}$ are learnable weights and bias, respectively.

The recovery process of topic model can be specified as

$$
\begin{aligned}
p(\hat{\boldsymbol{X}}) &= \int_{z_t} \int_{z_s} p(z_t) \left( \prod_{i=1}^{n} p(\hat{x}_i \mid z_t) p(z_t \mid z_s) p(z_s) \right) dz_s dz_t \\
&= \int_{z_t} \int_{z_s} p(\hat{\boldsymbol{X}}, z_s, z_t) dz_s dz_t. \qquad (5)
\end{aligned}
$$

Since the neural topic component is constructed in the fashion of a VAE, the ELBO of this component is in the following form:

$$
\begin{aligned}
\mathcal{L}_T &= \mathbb{E}_{q(z_s \mid X) q(z_t \mid X, z_s)} \left[ \log(p(\hat{\boldsymbol{X}} \mid z_t, z_s)) \right] \\
&\quad - \mathbb{E}_{q(z_s \mid X)} \left[ \mathbb{D}_{\text{KL}}(q(z_t \mid \boldsymbol{X}, z_s) \| p(z_t \mid z_s)) \right] \qquad (6)
\end{aligned}
$$

with $q(z_t \mid \boldsymbol{X}, z_s)$ and $p(z_t \mid z_s))$ to be the posterior and conditional prior of $z_t$, respectively.

### B. Sequence Modeling Component

The sequential information of sentences reveals the syntax structure of them. Since words in sentences are serially correlated, we thus construct a textVAE to generate words sequentially. A sequence modeling component in controllable LMs should not only be able to produce reasonable sentences but becomingly compose topic latent to its generating process. We assume that the sequence encoder infers a sequence latent code $z_s$, so the sequence decoder can generate topic-correlated texts via integrating learned topic latent codes and sequence latent codes. Specifically, a continuous variable $z_s$ is first drawn from its prior distribution $p(z_s)$. Since the semantic information associated with sentences substantially contains different subgroups (e.g., topics), we believe that the distribution of topic-specified texts is hard to be accurately captured by a standard VAE, which simply imposes an independent multivariate Gaussian prior on latent $z_s$. To fulfill the goal of generating semantically related texts, we also need to distinguish semantic expressions that are intrinsically correlated in real circumstances. Intuitively, this requires a well-expressive input of sentence decoder, and we hence make the hypothesis that $\boldsymbol{z_s}$ is sampled from a complex Gaussian prior with the full covariance matrix. $\boldsymbol{z_s}$ with a delicately designed modeling approach will not be directly used in producing topic-dependent texts. By sharing posterior knowledge with topic latent variable $\boldsymbol{z_t}$ as well as composing with it for decoder input, the backpropagation technique can update $\boldsymbol{z_s}$ in a trend of leveraging topic information into its representations. Thus, FET-LM is promising to fulfill the goal of creating controllable sentences with both $\boldsymbol{z_s}$ and $\boldsymbol{z_t}$.

In detail, we adopt a bidirectional encoder to encode the words and take the last hidden state of the encoder (denoted as $h_n$) to fit a Gaussian mean and log variance, respectively. Here, we initialize the prior of $\boldsymbol{z_s}$ with zero mean and all one in the covariance matrix, which is helpful to stably train a deep generative flow for posterior approximation. As for decoder, we obtain the overall latent code $z$ by concatenating $\boldsymbol{z_t}$ and $\boldsymbol{z_s}$: $z = [\boldsymbol{z_t}, \boldsymbol{z_s}]$, and the decoder is utilized to reconstruct document words with latent $z$ as input. For a reconstructed document $\boldsymbol{Y}$ output from the proposed method, its probability likelihood can be calculated as follows:

$$p(\boldsymbol{Y} \mid z) = \prod_{i=1}^{n} p(y_i \mid y_{1:i-1}, z) = \prod_{i=1}^{n} p(y_i \mid \boldsymbol{h_i}, z) \tag{7}$$

where $\boldsymbol{h}_i$ is the $i$th hidden state of the decoder RNN that satisfies $\boldsymbol{h}_i = \text{Decoder}(\boldsymbol{h}_{i-1}, x_{i-1}, z)$. Overall, the ELBO of our customized sequence VAE is

$$\mathcal{L}_S = \mathbb{E}_{q(z_t, z_s|X)}\big[\log(p(\boldsymbol{Y} \mid \boldsymbol{z_t}, \boldsymbol{z_s}))\big] - \mathbb{D}_{\text{KL}}(q(\boldsymbol{z_s} \mid \boldsymbol{X}) \| p(\boldsymbol{z_s})). \tag{8}$$

Here, we have $q(\boldsymbol{z_s} \mid \boldsymbol{X})$ and $p(\boldsymbol{z_s})$ to be the posterior and prior of $\boldsymbol{z_s}$, respectively.

### C. BoW Discriminator

Though the decoder of the sequence part in FET-LM composes both semantic and sequential features by sharing partial parameters for $p(\boldsymbol{z_t})$ modeling, there still stands a chance that the sequence part is not able to fully leverage $\boldsymbol{z_t}$. Following [11], we introduce a topic discriminator to aggregate the semantic expression of generated sentences. Our goal is to compel the model to generate topic-coherent texts with $\boldsymbol{z_t}$. In another word, the more alike of topic distribution between generated texts and original texts, the better it achieves our expectations. To do so, we empower the BoW encoder with the role of a discriminator. Since we are going to improve the topic coherence of sentences generated from the sequence decoder, the output of the sequence modeling component should be the input of our discriminator. However, the discrete property of generated texts is not friendly with the backpropagation process of the discriminator. Thus, we resort to the Gumbel-Softmax [52] distribution to approximate discrete samples. Specifically, we obtain the distribution of the whole corpus $p(\boldsymbol{Y} \mid \boldsymbol{z_t}, \boldsymbol{z_s}) = [\beta_1, \beta_2, \ldots, \beta_m]$ with $\beta_i$ to be the output probability of the $i$th topic word at any time step; then, we model word representations from the discriminator

$$b_i = \frac{\exp(\log(\beta_i) + g_1)/\tau}{\sum_{j=1}^{c} \exp\big(\log(\beta_j) + g_2\big)/\tau} \tag{9}$$

where $g_1$ and $g_2$ are drawn from the Gumbel(0, 1) distribution, $c$ is the vocabulary size and parameter $\tau$ is manually selected in advance. Unlike discriminator in [11], which utilized the word embedding to approximate output and forced the hidden size of word embedding equal to the topic encoder size, we utilize the topic model embedding in this process. As a result, the $i$th reconstructed topic word in our implementation is approximated as: $\hat{y}_i = b_i^T W_{\text{bow}}$, where $W_{\text{bow}}$ is the trainable BoW embedding in the topic encoder.

### D. HF for Sequence Posterior Approximation

Householder transformation (or elementary reflection) [53] is an orthogonal and volume-preserving transformation that transforms the $n$-dimensional vector to any other $n$-dimensional vectors. A normalizing flow consisting of such transformation is known as HF [54], [55]. When applying to distribution estimation, it is not only capable of generating more flexible sequence posteriors due to its nature as a flow but significantly simplifies the objective of flow-based variational methods because there stands $\log|\det(\partial \boldsymbol{H_k} \boldsymbol{z_{k-1}} / \partial \boldsymbol{z_{k-1}})| = 0$ for $k \in [1, K]$. By starting from a simple posterior with the full covariance matrix $\boldsymbol{z_{s(0)}}$ from sequence encoder, a $K$-layer HF is inflicted to it in order to better approximate the true posterior that befits various topics. The loss function of our sequence part in (8) should be modified as

$$\mathbb{E}_{q(z_t, z_{s(0)}|X)}\big[\log(p(\boldsymbol{X} \mid \boldsymbol{z_t}, \boldsymbol{z_{s(K)}}))\big] - \mathbb{D}_{\text{KL}}(q(\boldsymbol{z_{s(0)}} \mid \boldsymbol{X}) \| p(\boldsymbol{z_{s(K)}})). \tag{10}$$

Since HF is volume-preserving [41], the type of distribution will not change after the transformation. In the case of assuming that sequence prior follows a multivariate Gaussian, distribution after transformation is still a Gaussian but crucially with an intricate full covariance matrix. This property can approximate more complex sequence posterior with different semantics than isotropic Gaussian. Note that, though we only use normalizing flow to directly produce sequence posterior, the approximation method is also conducive to the topic latent $\boldsymbol{z_t}$ due to its conditional assumption on $\boldsymbol{z_s}$.

Distinct from TGVAE [13], which also utilizes HF but does not divide topic and sequence modeling and requires the Gaussian mixture model (GMM) to parameterize the hidden spaces, our method is more simple and effective to employ (check Section IV for experimental results).

### E. Training Losses

For both the topic and sequence modeling components, we adopt AutoEncoding variation Bayes (AEVB) to achieve posterior inference and parameter learning. As a result, $\mathcal{L}_S$ and $\mathcal{L}_T$ consist of the reconstruction term and regularization term concerning $\boldsymbol{z_s}$ and $\boldsymbol{z_t}$, respectively. Then, the training objective for the model is $\mathcal{L}_{\text{VAE}} = \mathcal{L}_S + \mathcal{L}_T$, whose regularization terms are corelative and can be summed to a unified KLD to form the regularization term of a holistic VAE

$$\begin{aligned}\mathcal{L}_{\text{VAE}} &= \mathcal{L}_S + \mathcal{L}_T \\ &= \mathbb{E}_{q(z_t, z_{s(0)} \mid X)}\big[\log(p(\boldsymbol{Y} \mid z_t, z_{s(K)}))\big] \\ &\quad + \mathbb{E}_{q(z_{s(0)} \mid X) q(z_t \mid X, z_{s(0)})}\big[\log(p(\hat{\boldsymbol{X}} \mid z_t, z_{s(K)}))\big] \\ &\quad - \underbrace{\mathbb{D}_{\text{KL}}(q(z_t, z_{s(0)} \mid \boldsymbol{X}) \| p(z_t, z_{s(K)}))}_{\text{regularization term}} \qquad (11)\end{aligned}$$

where $z_{s(k)}$ represents the sequence latent variable after the $k$th householder transformation as mentioned in Section III-D. The goal of making the topic encoder as a discriminator is to narrow the gap of semantics between the original texts and generated texts. This idea can be approximated by the log-likelihood maximization of $z_t$

$$\mathcal{L}_D = \mathbb{E}_{p(z_s) p(z_t)}\big[\log(q(z_t \mid \boldsymbol{Y}))\big] \qquad (12)$$

where $\boldsymbol{Y}$ is the generated sentences from the sequence decoder. Finally, the whole loss function of the FET-LM model is designed as: $\mathcal{L} = \mathcal{L}_{\text{VAE}} + \lambda_D \mathcal{L}_D$.

## IV. Experiment and Results

We evaluated FET-LM from two general perspectives: language modeling ability and topic coherence, which can well and truly reveal both the generation capacity and topic learning ability of the proposed model. We conducted several experiments on multiple datasets and compared them with numerous baselines. Specifically, we used both text perplexity (PPL) and BLEU-based metrics [56] for text modeling evaluation, while topic coherence was evaluated through normalized PMI (NPMI) [57] and a supervised classification task quantitatively. We also visualized the distribution cluster of learned sentiment in our topic latent code. Finally, from the perspective of text generation, we exhibited controllable texts, latent interpolated generation, and text style transfer task to visually illustrate the generation capacity of FET-LM. Our code is available at https://github.com/ImKeTT/FET-LM.

### A. Datasets and Model Details

Empirical studies of the text modeling performance were performed on four text datasets: APNEWS,[1] IMDB [58], BNC [59], and PTB [60]. For these four corpora, we first used SpaCy[2] to tokenize the sentences and lowercase all word tokens. Then, we followed previous works [11], [13], [17] to filter out the words whose occurrence frequency was less than 2 times (8 for BNC to accelerate the training procedure). For the evaluation of topic learning, we added Yelp15[3] dataset. With sentiment labels, Yelp15 allows us to conduct classification and visualization of learned latent distributions.

For the purpose of keeping the topic component focusing on valuable words that represent different topics, subtracting a set of specific words (e.g., stop words, rare words, and frequent words) from the original corpus as the input of the topic model is widely accepted. This process can make our topic model more reliable. In our system, we dealt with this situation in a slightly different way. We still input the whole corpus to the topic encoder, but additionally added a postprocessing stage to eliminate specific words: counted all of them to zero for BoW import. In addition, for the input of topic modeling part, we moved out stop words in every document and removed the top 0.3% most frequent words as well as words that appear less than 100 documents. The summarized statistics of all five datasets can be found in Table I.

Regarding training details, we fixed a maximum vocabulary size of 40k and a maximum length of 80 words across the first four text datasets (APNEWS, IMDB, BNC, and PTB). Considering the relatively longer text length and much bigger vocabulary size of Yelp15, we followed [11] and set the maximum vocabulary size to 20k with a maximum text length of 150 to expedite the training process. Pretrained word vectors from GloVe [61] were first utilized to initialize word embedding with a dimension of 200, which was shared by both topic and sequence modeling components. The encoder of the topic modeling component follows the BoW manner, which was implemented with a two-layer feedforward network with 200 hidden units and softplus activation function. We set the dimension of $z_t$ to 20. The sequence encoder was a bidirectional LSTM [62] with hidden size 300 for both directions, and the decoder was a plain LSTM with hidden size 300. Also, the size of sequence latent $z_s$ was 32. We used a batch size of 32 and Adam [63] optimizer with a learning rate of $10^{-4}$ for model training. The training epoch number was set to 80 with 2000 steps per epoch for datasets except for BNC (100 epochs) and IMDB (60 epochs). The weight decay rate was set to $10^{-5}$ with a dropout ratio of 0.2 for all RNNs. To avoid gradient explosion, we set the max clip norm of the gradient to 5.0. Moreover, to take full advantage of learned latent knowledge, cyclical schedule [35] with four cycles through all training epochs was utilized for KL annealing. Finally, we set the weight of discriminator loss to 0.5 through ablation studies. For HF implementation, we followed the experiment setting from [54]. Finally, the parameter $\tau$ in the BoW discriminator was 0.02 during training and 1.0 at inference. One NVIDIA GeForce 1080Ti GPU was used for training.

### B. Language Modeling Evaluation

FET-LM is intrinsically an LM. Thus, PPL and BLEU-related metrics for text quality measurement are suitable to evaluate model capability.

*1) Text Quality Analysis:* We quantified the quality of generated sentences in terms of text PPL, which reveals the model confidence of generating a sequence of words. The lower the PPL of a sentence is, the higher quality this sentence has. Specifically, we estimated PPL via the log-likelihood loss from the sequence decoder and normalized it by generated word number. To take a closer look at the role the BoW discriminator plays, we chose models with or without it in Table II. Also, we find that the HF is contributing to PPL

[1] https://www.ap.org/en-gb/

[2] https://spacy.io

[3] https://www.yelp.com/dataset

TABLE I

Statistical Summary of Five Datasets Used in This Article. SM Voc and TM Voc Represent the Vocabulary Size of Sequence Model and Topic Model, Respectively

| Data | #SM Voc | #TM Voc | #Training Docs | #Validation Docs | #Test Docs | #Avg Len |
|---|---|---|---|---|---|---|
| **APNEWS** | 22,760 | 7,498 | 50k | 2.0k | 2.0k | 21.4 |
| **IMDB** | 27,763 | 5,829 | 75k | 12.5k | 12.5k | 22.5 |
| **BNC** | 22,154 | 7,700 | 15k | 1.0k | 1.0k | 22.6 |
| **PTB** | 9,733 | 4,498 | 42k | 3.8k | 3.4k | 24.8 |
| **Yelp15** | 20,004 | 7,575 | 74k | 7.4k | 7.4k | 75.3 |

TABLE II

Text Quality Analysis in Terms of Text Perplexity (PPL). All Topic LMs Remain the Same Topic Latent Size (If Available) of 50

| Model | APNEWS | IMDB | BNC | PTB |
|---|---|---|---|---|
| **LSTM+LDA** | 57.05 | 69.58 | 96.42 | - |
| **Topic-RNN [15]** | 56.77 | 68.74 | 94.66 | 97.3 |
| **TDLM [57]** | 53.00 | 63.67 | 87.42 | - |
| **LSTM VAE [30]** | 75.89 | 86.16 | 105.10 | 96.0 |
| **VAE+HF** | 71.60 | 83.67 | 104.82 | - |
| **TCNLM [12]** | 52.75 | 63.98 | 87.98 | - |
| **TGVAE [13]** | 51.27 | 59.45 | 88.34 | - |
| **DVAE [10]** | - | - | - | 33.4 |
| **TATGM [11]** | 47.23 | 52.01 | 80.78 | - |
| **rGBN-RNN [16]** | 42.71 | 51.36 | 79.13 | - |
| **VRTM [17]** | 47.78 | 51.08 | 86.33 | 55.82 |
| **iVAE [25]** | - | - | - | 53.44 |
| **APo-VAE [34]** | - | - | - | 53.02 |
| **DPrior [64]** | - | - | - | 46.08 |
| **Ours** | 36.35 | **36.53** | **76.34** | **27.25** |
| **Ours w/o Dis** | **36.11** | 37.26 | 78.31 | 27.67 |

TABLE III

PPL of Our Models on Test Set With Various Number of Flow Layers (Represented by F) on Two Datasets

| Dataset | F=0 | F=5 | F=10 | F=20 |
|---|---|---|---|---|
| **IMDB** | 52.01 | 37.48 | 36.53 | 35.75 |
| **PTB** | 49.06 | 27.40 | 27.25 | 26.94 |

metric, so we present PPL values of FET-LM with different layer settings on two representative corpora in Table III.

### *C. Baseline Models*

In our experiments, we compared against two categories of baselines that mostly consider both topic and sequence information into generation. Five baselines belong to the LM-based approaches.

1) LSTM + LDA fuses the topic information from a pretrained LDA model with the hidden states of LSTM.
2) Topic-RNN [15] coalesces the topic distribution learned from an LDA component using the gate mechanism and trains jointly with the LM.
3) TDLM [57] employs a convolutional network for the topic model and also concatenates it with hidden states of RNN.
4) rGBN-RNN [16] brings a gamma belief network as a topic model and infuses learned topic information into RNN to improve model capability.

As for VAE-based models, we have the following baselines.

1) LSTM VAE [30] is a plain textVAE model whose encoder and decoder are implemented with LSTM.
2) VAE + HF is built based on plain textVAE with HF to estimate its latent distribution.
3) TCNLM [12] utilizes a neural topic model based on the VAE paradigm and a multiple experts network to generate texts.
4) TGVAE [13] consists of the same topic model of TCNLM, but a textVAE with Gaussian mixture prior and an HF to approximate its posterior.
5) DVAE [10] incorporates a simple Dirichlet latent topic model to improve textVAE.
6) TATGM [11] applies multivariant Gaussian for both topic and sequence latent codes and concatenates them for sentence generation.
7) VRTM [17] blends RNN hidden state with a binary vector sign to judge topic expression.
8) iVAE [25] parameterizes hidden space with sample method and replaces KLD with mutual information.
9) APo-VAE [34] makes the latent space a Riemannian manifold with learnable prior and posterior.
10) DPrior [64] utilizes discrete latent prior for controllable text generation with annotations.

We took baseline results from the original papers with the same topic number to our setup (if available) for fairness. According to the results in both tables, first, the proposed method takes up the top positions compared with best-performed baselines, especially on APNEWS and IMDB corpus. FET-LM precedes presently the state-of-the-art performance from [16] six and over ten points, respectively. These results demonstrate that FET-LM is fairly designed to fulfill the principle goal of an LM. Second, HF in sequence latent level decreases the PPL value by over ten absolute points on both IMDB and PTB. Besides, with the increase of flow layers, the PPL value gradually reduces. Third, FET-LM without flow can still reach competitive PPL results compared with baselines,

TABLE IV
TEXT QUALITY ANALYSIS IN TERMS OF TEST-BLEU AND BLEU-F1 SCORE. T IS THE TOPIC NUMBER AND F IS THE FLOW LAYER NUMBER

| Metrics | Methods | APNEWS | | | IMDB | | | BNC | | | PTB | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | B-2 | B-3 | B-4 | B-2 | B-3 | B-4 | B-2 | B-3 | B-4 | B-2 | B-3 | B-4 |
| *test*-BLEU↑ | **VAE** [30] | 0.564 | 0.278 | 0.192 | 0.597 | 0.315 | 0.219 | 0.479 | 0.266 | 0.169 | 0.5215 | 0.3633 | 0.2642 |
| | **VAE+HF** | 0.570 | 0.279 | 0.195 | 0.610 | 0.322 | 0.221 | 0.483 | 0.270 | 0.169 | 0.5565 | 0.3616 | 0.2529 |
| | **TGVAE (T=10)** [13] | 0.584 | 0.327 | 0.202 | 0.621 | 0.357 | 0.223 | 0.518 | 0.283 | 0.173 | - | - | - |
| | **TGVAE (T=30)** [13] | 0.627 | 0.335 | 0.207 | 0.655 | 0.369 | 0.243 | 0.528 | 0.291 | 0.182 | - | - | - |
| | **TGVAE (T=50)** [13] | 0.629 | 0.340 | 0.210 | 0.652 | 0.372 | 0.239 | 0.535 | 0.290 | 0.188 | - | - | - |
| | **Ours (F=10, T=10)** | 0.6512 | 0.3862 | 0.2358 | 0.7202 | 0.4505 | 0.2470 | 0.6997 | 0.5947 | 0.4934 | 0.6824 | 0.4847 | 0.3564 |
| | **Ours (F=10, T=30)** | 0.6434 | 0.3776 | 0.2374 | 0.7037 | 0.4347 | 0.2566 | 0.6791 | 0.5473 | 0.4502 | 0.6705 | 0.4779 | 0.3438 |
| | **Ours (F=5, T=50)** | 0.6449 | 0.3801 | 0.2241 | 0.7136 | 0.4323 | 0.2444 | 0.7397 | 0.6422 | 0.5521 | 0.6599 | 0.4710 | 0.3407 |
| | **Ours (F=10, T=50)** | **0.6757** | 0.3983 | 0.2432 | **0.7542** | **0.4753** | **0.2755** | **0.7681** | **0.6610** | **0.5672** | **0.6924** | **0.5076** | **0.3733** |
| | **Ours (F=20, T=50)** | 0.6558 | 0.3809 | 0.2187 | 0.7374 | 0.4660 | 0.2543 | 0.6744 | 0.5660 | 0.4818 | 0.6790 | 0.5001 | 0.3661 |
| | **Ours (F=10, T=50) w/o Dis.** | 0.6596 | **0.4100** | **0.2497** | 0.7447 | 0.4637 | 0.2678 | 0.7316 | 0.6234 | 0.5292 | 0.6484 | 0.4587 | 0.3297 |
| **BLEU-F1↑** | **VAE** [30] | 0.2166 | 0.3491 | 0.3071 | 0.1843 | 0.3394 | 0.3364 | 0.2273 | 0.3448 | 0.2812 | 0.2033 | 0.4055 | 0.3843 |
| | **VAE+HF** | 0.2077 | 0.3439 | 0.3121 | 0.1689 | 0.3363 | 0.3401 | 0.2242 | 0.3456 | 0.2809 | 0.2174 | 0.4292 | 0.3692 |
| | **TGVAE (T=10)** [13] | 0.2524 | 0.3916 | 0.3248 | 0.1883 | 0.3872 | 0.3446 | 0.2571 | 0.3645 | 0.2874 | - | - | - |
| | **TGVAE (T=30)** [13] | 0.2904 | 0.4081 | 0.3324 | 0.2441 | 0.4014 | 0.3693 | 0.2837 | 0.3750 | 0.2998 | - | - | - |
| | **TGVAE (T=50)** [13] | 0.2942 | 0.4124 | 0.3368 | 0.2544 | 0.4036 | 0.3651 | 0.2985 | **0.3751** | 0.3079 | - | - | - |
| | **Ours (F=10, T=10)** | 0.3720 | 0.4088 | 0.3362 | 0.3193 | 0.4265 | 0.3501 | 0.2875 | 0.3299 | 0.3513 | 0.3233 | 0.3998 | 0.4027 |
| | **Ours (F=10, T=30)** | **0.4007** | 0.4268 | 0.3484 | **0.3371** | 0.4337 | 0.3642 | 0.2933 | 0.3564 | 0.3845 | **0.3562** | 0.4350 | 0.4168 |
| | **Ours (F=5, T=50)** | 0.3671 | 0.4079 | 0.3283 | 0.3323 | 0.4289 | 0.3507 | 0.3108 | 0.3531 | 0.3802 | 0.3475 | 0.4269 | 0.4119 |
| | **Ours (F=10, T=50)** | 0.3813 | **0.4281** | 0.3487 | 0.3272 | **0.4415** | **0.3809** | **0.3358** | 0.3725 | **0.3989** | 0.3459 | 0.4246 | **0.4241** |
| | **Ours (F=20, T=50)** | 0.3603 | 0.3996 | 0.3185 | 0.3010 | 0.4300 | 0.3587 | 0.2956 | 0.3418 | 0.3623 | 0.3540 | **0.4364** | 0.4293 |
| | **Ours (F=10, T=50) w/o Dis.** | 0.3842 | 0.4228 | **0.3490** | 0.3148 | 0.4310 | 0.3709 | 0.3284 | 0.3653 | 0.3850 | 0.3287 | 0.4093 | 0.3986 |

which yields convincing effectiveness of our model design. Finally, models with the BoW discriminator reach a lower PPL in major cases, which is ascribed to the resolute guidance of our implemented discriminator.

*1) Text Relevance and Diversity Analysis:* The BLEU-related calculation is based on the $n$-gram language paradigm, by seeking for identical strings between reference and generated texts, it gives the matching precision as a similarity rating of two sentences. Following previous works [13], [16], we used *test*-BLEU to evaluate the quality of generated sentences with texts from the test sets as a reference, higher the *test*-BLEU score is, texts with more realistic-looking content are provided. Besides, we use *self*-BLEU to evaluate the diversity of generated contents [65]. Since there intrinsically exists a tradeoff between text quality and text diversity, we employ that the BLEU-F1 score involves text quality and diversity following [66]:

$$\text{BLEU-F1} = \frac{2 \times \text{test-BLEU} \times (1 - \text{self-BLEU})}{\text{test-BLEU} + (1 - \text{self-BLEU})}. \quad (13)$$

For the baseline methods, three VAE-based LMs were selected, among which VAE + HF and TGVAE are two systems utilizing HF like the proposed FET-LM. To fully explore the model capacities under different topic dimension settings, we chose to vary the model's topic number from 10 to 50. Since BLEU-related metrics require specific word output and comparison, we believe that the discriminator can play a more important role in this process because it is optimized on the word token level, and we report model performances with or without it. Formally, we carried out all the BLEU-related experiments using the benchmark tool Texygen [65]. From the *test*-BLEU and BLEU-F1 scores in Table IV, we could see that our FET-LM model is superior to the baselines in terms of BLEU-F1 and *test*-BLEU in most cases, and the discriminator is a strong performer in improving text quality (higher *test*-BLEU values in all circumstances). When the flow layer is selected to 10, our model generally performs the best, so the flow layer number is 10 for the rest experiments. Moreover, values of FET-LM on BLEU-F1 change much smoother than others from B-2 to B-3. One possible reason is that FET-LM produces more coherent texts (with lower loss in $n$-gram LMs) than other baselines do.

## D. Topic Learning Evaluation

Text quality indicators like PPL are not necessarily relevant to topic modeling ability [67]. Hence, experiments were further conducted to verify the topic modeling ability of FET-LM.

*1) NPMI Evaluation:* NPMI score measures the coherence of generated topic-related words. Following [57], $n$ was selected from $5, 10, 15,$ and $20$. Then, we averaged values across different $n$ values as the NPMI score. The PMI score is calculated based on distinguishing representative words from different topics, which not only requires words from an appointed topic but needs words from other topics as intruders. Specifically, we fixed the value of each latent dimension of $z_s$ and $z_t$ to a preset number successively while others to 0 and then output every $n$ most typical words from the topic modeling part. We sampled a topic word from another topic and appended it to the previously obtained $n$ topic words to complete the intrusion process.

According to our preceding experimental results, we added or discarded the BoW discriminator and assigned ten flow layers to check the model performance. As for NPMI scores of baselines, we took all the statistics from [11] and trained an LDA model with 20 topics over PTB dataset for the NPMI calculation. Based on the overall results of NPMI

TABLE V

NPMI SCORES FOR TOPIC COHERENCE EVALUATION

| Methods | APNEWS | IMDB | BNC | PTB | Yelp15 |
|---|---|---|---|---|---|
| **LDA** [42] | 0.125 | 0.084 | 0.106 | 0.118 | 0.087 |
| **TDLM** [57] | 0.149 | 0.104 | 0.102 | - | - |
| **Topic-RNN** [15] | 0.134 | 0.103 | 0.102 | - | - |
| **TCNLM** [12] | 0.159 | 0.106 | 0.114 | - | - |
| **TGVAE** [13] | 0.157 | 0.105 | 0.113 | - | - |
| **TATGM** [11] | **0.171** | **0.121** | 0.115 | - | 0.114 |
| **Ours** | 0.162 | 0.099 | **0.119** | **0.148** | **0.131** |
| **Ours w/o Dis.** | 0.163 | 0.092 | 0.116 | 0.130 | 0.129 |

TABLE VI

TOPIC WORD GENERATION OF TOP-5 REPRESENTATIVE WORDS FROM FIVE LEARNED TOPICS IN FET-LM

| Dataset | #1 | #2 | #3 | #4 | #5 |
|---|---|---|---|---|---|
| **APNEWS** | sentence | ballot | marriage | snow | probation |
| | fraud | seat | gay | wind | stabbing |
| | pornography | districts | districts | heavy | prosecutor |
| | conviction | election | laws | winds | felony |
| | stabbing | medicaid | ethics | rain | sexually |
| **IMDB** | favourite | mess | parents | screening | scientist |
| | funniest | poorly | finds | cable | mysterious |
| | easily | lame | married | toronto | killed |
| | animated | violence | sister | premiere | murders |
| | anime | ridiculous | girlfriend | tcm | murdered |
| **BNC** | court | gmt | meeting | born | chapter |
| | chapter | 1993 | work | wife | london |
| | children | thu | president | daughter | council |
| | school | jan | june | john | international |
| | darlington | nov | held | son | british |
| **PTB** | traders | director | systems | points | oct |
| | firms | article | old | dow | old |
| | dollar | women | little | priced | ms |
| | wall | robert | traders | benchmark | nov |
| | corporate | contributed | economy | mortgages | age |
| **Yelp15** | donuts | une | taco | delivery | dry |
| | cupcakes | très | pho | orders | average |
| | donut | pas | roll | attitude | mediocre |
| | burgers | des | asada | appointment | soggy |
| | bagels | avec | carne | refund | salty |

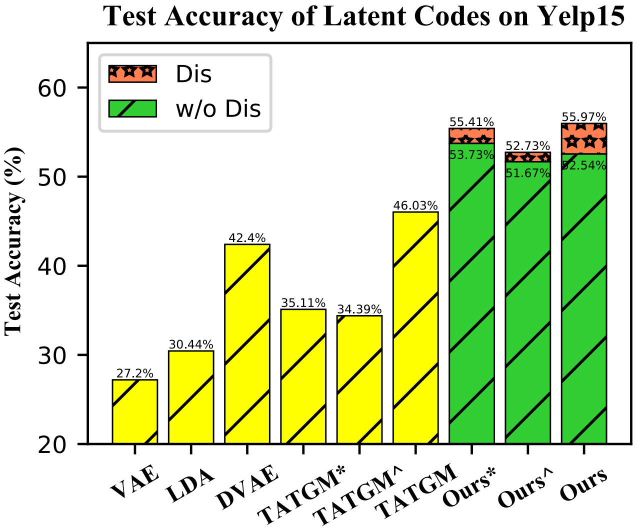


Fig. 4. Classification accuracy on Yelp15. * or ˆ represent results inferred from latent codes of the topic part and sequence part (or structural part in TATGM), respectively. The topic latent $z_t$ can learn more topic knowledge than sequence latent $z_s$ solely (i.e., higher accuracy). The discriminator is also helpful in performing the task well.

shown in Table V, the BoW discriminator is designed to enhance the topic learning capacity of FET-LM, and as a result, its impact on NPMI calculation is positive. FET-LM performs well for achieving the highest NPMI score over three datasets (BNC, PTB, and Yelp15) and being suboptimal on APNEWS.

Though the primary goal of the proposed model is to generate sentences with attributes instead of topic words, our model exhibits competitive topic learning capacity compared with baselines. As a result, the topic modeling component as an independent and qualified topic model is regarded as a side product of FET-LM.

*2) Latent Codes Classification:* Can the latent representations of FET-LM really distinguish different topics or sentiments? To further verify the topic learning ability of FET-LM, we conducted a supervised classification task on the Yelp15 dataset, each sentence from which owns a semantic label. We first obtained latent codes from the topic component and sequence component with sampled 2000 training data using a well-trained ten-flow-layer FET-LM and then constructed a two-layer linear feedforward network with softmax function as a classifier. Finally, we tested the performance of the classification model on the validation set. The higher the accuracy is, the stronger the topic extraction power a model possesses. In the case of supervised latent classification task, we made $z_s$, $z_t$, and $z = [z_s, z_t]$ as input severally, and the statistical results are presented in Fig. 4. We can get the following conclusions.

1) The classification task shows the overall superiority of FET-LM on topic learning. For instance, the best and the worst classification accuracy of FET-LM come from intact $z = [z_s, z_t]$ (Acc. = 55.97%) and single $z_s$ (Acc. = 52.73%), respectively. They are superior to the currently best-performed TATGM by almost 10% and over 6%.
2) No matter with or without BoW discriminator, the test accuracy of topic latent $z_t$ as input exceeds sequence latent variable $z_s$ as input, which exactly manifests that $z_t$ from topic component could learn more topic knowledge than $z_s$ does.
3) Models with the BoW discriminator reach a higher classification accuracy than models without it, indicating that the BoW discriminator helps latent variables to recognize different topics efficiently.

*3) Topic Latent Visualization:* Intuitively, the strong topic learning ability of FET-LM can also be captured by visualizing learned topic distribution. We randomly sampled 2000 examples from Yelp15 with labels and applied t-SNE [68] to visualize the distribution of learned $z_t$. In the clustering setting, texts with different scores were roughly grouped into negative (cyan) or positive (orange) sentiment. Note that, rating labels in Yelp15 are considered to be continuous, and it is involuntary to say that they are possibly entangled at the edge. Still, the separated sentimental distribution can be identified by the well-educated $z_t$ in our model as shown in Fig. 5, which demonstrates a structured latent pattern of $z_t$ and explains why FET-LM was yielding a decent classification performance.

TABLE VII
CONTROLLABLE TEXT GENERATION ON FIVE DATASETS

| Dataset | Topic | Sentences |
|---|---|---|
| **APNEWS** | **Crime** | • the fbi says authorities are investigating on suspicion of a fatal shooting in south of phoenix downtown. |
| | **Weather** | • tropical storm irene is warning residents of the county house that two of people affected by a tornado in south carolina. |
| **IMDB** | **Negative** | • i have seen this movie that will probably be the worst movie that is crap . |
| | **Actor** | • how can me start off for a famous performance for a very young actress and her performance in her role, so she was very funny and beautiful . |
| **BNC** | **War** | • two <unk>in the world war is being killed for a <unk> |
| | **Finance** | • the uk economy has been launched by the end of the year , according to a million contract to help out to the # 1 billion damages to boost their own business˙ |
| **PTB** | **Finance** | • third - quarter u.s . sales have been high - priced |
| | **Business** | • the world - wide business and development of credit - card business to a group ’s largest business |
| **YELP15** | **Negative** | • i do n’t know why ? this is the worst place ever ! ! ! ! ! ! ! ! ! ! ! they do n’t have the same thing they do n’t know how to do it . |
| | **Neutral** | • first time to go back . waited for a while , waited for 15 minutes. ordered a burger, which was ok . ordered a burger with fries . $ 10 for a burger was decent . |
| | **Positive** | • staff is friendly . very clean . i have been to many other locations in toronto area . i will continue to visit this location to location . |

TABLE VIII
TEXT STYLE TRANSFER GENERATION FROM NEGATIVE TO POSITIVE BY TRAVERSING LEARNED TOPIC REPRESENTATION ON YELP15

| | |
|---|---|
| **Int. 1** | •ok . the waiter was rude to us , we did n’t know what we wanted to do with our food ... we were told that they were not busy at all. |
| **Int. 2** | •very disappointing . the only thing that was not the best thing about this place is that they do n’t care about the quality of the food ! ! ! we were not impressed with the service , food was bad , service was horrible . |
| **Int. 3** | •not very disappointed . the only thing that was not the best thing about this place is that they do n’t care about the quality of the food ! ! ! we were not impressed with the service , food was good , service was horrible . we . will be back to try their <unk> |
| **Int. 4** | •not bad . the food was not bad , we had to ask for the <unk>sauce . we were told that they were not only to be able to get our food to be delivered . we were told that they were n’t even busy , but we were not impressed with the service . we will be back to try this place again ! |
| **Int. 5** | •not bad . the food was not bad , but the <unk>was not too salty . we were told that they were n’t even able to get our food to be delivered to the kitchen . we were told that they were n’t even busy . we had a great time to go to this place , the service was great ! |
| **Int. 6** | •not bad at all ! the food was not bad at all ! the only thing i would say was that the service was great . we were greeted by the owner and he was very friendly and helpful . we will be back for sure . |
| **Int. 7** | •not sure what i wanted to say about this place but the service was great . we were in the area for a few minutes and they were very nice . they were very friendly and helpful . i would recommend this place to anyone who . likes the <unk> |
| **Int. 8** | •this place is amazing and the breakfast is delicious and the staff is very friendly . i will be back . |
| **Int. 9** | •this starbucks is my favorite breakfast spot , i have been to a few times . i have a good time and i have a good time . the coffee is very good and the staff is very friendly . i will be back . |

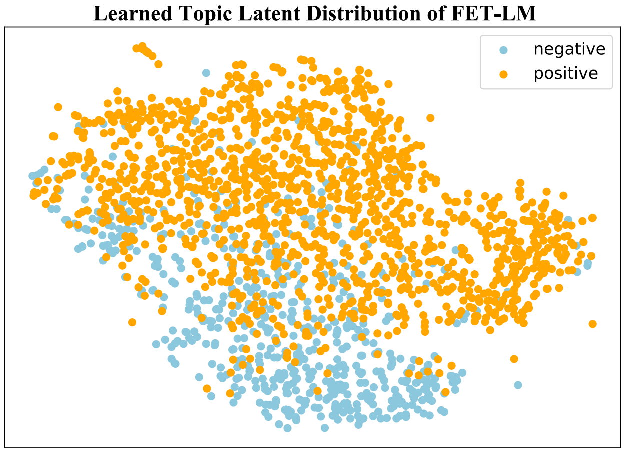


Fig. 5. Visualization of learned topic distribution from $z_t$ on Yelp15 with sentiment labels. A separation between positive and negative sentiment can be captured by $z_t$ from the clustering.

### E. Guided Text Generation

To demonstrate that the proposed FET-LM is able to generate controllable sentences, we conduct three downstream generation tasks: 1) topic word generation and controllable text generation in an unsupervised manner; 2) text style transfer; and 3) sentence interpolated generation to verify its capacities.

For unsupervised controllable generation, we selected representative topic words (Table VI) and topic-specified sentences (Table VII) from our trained model. Since every dimension of latent codes in FET-LM represents a topic or a sentiment ideally, we can easily manipulate the values of the topic and sequence latent variables to generate topic words or texts with different attributes. The model input is an one-hot $z_t$ and $z_s \sim \mathcal{N}(0, I)$ for latent spaces. We then feed both variables into the sequence decoder for controllable text generation and into the topic decoder for topic word production. From Tables VI and VII, it is clear that FET-LM can produce words belonging to certain topics or generates sentences with diverse topics on different corpora.

For text style transfer in Table VIII, the model input is $z_s$ with one certain dimension (e.g., the 1st) to be a preset number (e.g., $n$) and others to be 0, and the topic latent vector is obtained by the conditional assumption based on sequence latent. The output is the corresponding sentence. We do the

TABLE IX
INTERPOLATED SENTENCES ON IMDB

| Type | Sentences |
|---|---|
| **Ori. 1** | **i laughed at the movie !** |
| **Rec. 1** | who gets the movie , then again may be <unk>on a great movie ! ! |
| **Int. 1** | i ' ve got to tell you <unk>, why the movie is so incredibly entertaining . |
| **Int. 2** | i ' m going to tell you <unk>, the movie is so dark - making . |
| **Int. 3** | i ca n't believe it 's crap with the original game , <unk>out . |
| **Int. 4** | i ca n't believe i watched it 's own way . |
| **Rec. 2** | i ca n't believe i watched it at any time . |
| **Ori. 2** | **ca nt believe it ... .** |

TABLE X
INTERPOLATED SENTENCES ON APNEWS

| Type | Sentences |
|---|---|
| **Ori. 1** | **pricing details were n't immediately available** |
| **Rec. 1** | treasury bills were n't disclosed |
| **Int. 1** | treasury securities were n't disclosed |
| **Int. 2** | treasury securities were n't available by location and others |
| **Int. 3** | federal home loan mortgage corp freddie mac was unchanged |
| **Int. 4** | federal home loan mortgage corp freddie mac posted total of n |
| **Rec. 2** | federal home loan mortgage corp freddie mac posted yields in amounts of deposit |
| **Ori. 2** | **federal home loan mortgage corp .** |

style transfer task by traversing $n$ in a range of numbers. As shown in Table VIII, there is a sentiment transformation from negative to positive by traversing latent codes. Adjacent sentences share a similar context structure while gradually converting sentiment, that is to say, by manipulating expressive learned latent spaces, we could obtain effective implicit guidance for context generation while maintaining a consistent structure.

For the interpolated generation in Tables IX and X, the model input is two sentences from the test set (Ori. 1 and Ori. 2) and the output is a set of sentences with their latent vectors interpolated from latent codes from Ori. 1 to Ori. 2. We took FET-LM trained on IMDB as well as APNEWS for this task. We first randomly sampled two sentences from the test corpus and fed them to our FET-LM, then employed linear interpolation between the latent values inferred from given content pairs, and finally generated texts from manipulated latent codes. We can observe from the results: the latent interpolation task makes clear what FET-LM has learned for text generation from its latent space. In detail, we take the generated sentences from Table IX as an example, and Ori. 1 and Ori. 2 have distinct sentiment polarity (positive for Ori. 1, while negative for Ori. 2). Rec. 1 reconstructed from Ori. 1 maintains the positive sentiment and some feature words or symbols (e.g., word "movie," exclamation symbol). Similarly, Rec. 2 reserves doubt and negative sentiment, as well as the major structure in Ori. 2 (e.g., the statement "n't," the word "believe"). During the interpolation, the semantic feature alters from the second interpolated text with a distinctive word "dark-making," while sentence syntax structure changes progressively. As a result, we could say that the latent codes of FET-LM have properly learned text structural information as well as semantic meanings from input sentences.

## V. CONCLUSION

Generating topic-specified texts is an important, ambitious, and well-identified challenge in the literature. In this article, we propose a flow-enhanced VAE FET-LM for topic-guided language modeling, which controls text generation by incorporating a VAE-based neural sequence model and a neural topic model parameterized by the Dirichlet distribution. For scalable reasoning, we developed autoencoding variational inference based on HF, allowing efficient unsupervised end-to-end training and more accurate latent distribution estimation. Besides, the well-expressive sequence posterior is also used for conditional topic latent modeling, which releases the burden of the topic component as well as drives the LM to take full advantage of its powerful generative capacity endowed by the normalizing flow. Empirical results, including language modeling and topic learning evaluations, show clear advantages of FET-LM compared to previous works across multiple NLP tasks.

## REFERENCES

[1] D. Bahdanau, K. Cho, and Y. Bengio, "Neural machine translation by jointly learning to align and translate," in *Proc. 3rd Int. Conf. Learn. Represent. (ICLR)*, Y. Bengio and Y. LeCun, Eds. San Diego, CA, USA: May 2015, pp. 1–6.

[2] B. Zhang, D. Xiong, J. Xie, and J. Su, "Neural machine translation with GRU-gated attention model," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 31, no. 11, pp. 4688–4698, Nov. 2020.

[3] A. M. Rush, S. Chopra, and J. Weston, "A neural attention model for abstractive sentence summarization," 2015, *arXiv:1509.00685*.

[4] I. Serban, A. Sordoni, Y. Bengio, A. Courville, and J. Pineau, "Building end-to-end dialogue systems using generative hierarchical neural network models," in *Proc. AAAI Conf. Artif. Intell.*, vol. 30, no. 1, 2016, pp. 1–8.

[5] T. Shi and Y. Song, "A novel two-stage generation framework for promoting the persona-consistency and diversity of responses in neural dialog systems," *IEEE Trans. Neural Netw. Learn. Syst.*, early access, Aug. 30, 2021, doi: 10.1109/TNNLS.2021.3105584.

[6] A. M. Turing, "Computing machinery and intelligence," in *Parsing the Turing Test: Philosophical and Methodological Issues in the Quest for the Thinking Computer*, R. Epstein, G. Roberts, and G. Beber, Eds. Dordrecht, The Netherlands: Springer, 2009, pp. 23–65, doi: 10.1007/978-1-4020-6710-5_3.

[7] C. Paris, *User Modelling in Text Generation*. London, U.K.: Bloomsbury Publishing, 2015.

[8] E. Mayfield et al., "Equity beyond bias in language technologies for education," in *Proc. 14th Workshop Innov. Use NLP Building Educ. Appl.*, 2019, pp. 444–460.

[9] C. Garbacea and Q. Mei, "Neural language generation: Formulation, methods, and evaluation," 2020, *arXiv:2007.15780*.

[10] Y. Xiao, T. Zhao, and W. Y. Wang, "Dirichlet variational autoencoder for text modeling," 2018, *arXiv:1811.00135*.

[11] H. Tang, M. Li, and B. Jin, "A topic augmented text generation model: Joint learning of semantics and structural features," in *Proc. Conf. Empirical Methods Natural Lang. Process. 9th Int. Joint Conf. Natural Lang. Process. (EMNLP-IJCNLP)*, 2019, pp. 5090–5099.

[12] W. Wang et al., "Topic structure-aware neural language model," in *Proc. Int. Conf. Artif. Intell. Statist.*, May 2019, pp. 356–365.

[13] W. Wang et al., "Topic-guided variational auto-encoder for text generation," 2019, *arXiv:1903.07137*.

[14] Z. Hu, Z. Yang, X. Liang, R. Salakhutdinov, and E. P. Xing, "Toward controlled generation of text," in *Proc. Int. Conf. Mach. Learn.*, 2017, pp. 1587–1596.

[15] A. B. Dieng, C. Wang, J. Gao, and J. Paisley, "TopicRNN: A recurrent neural network with long-range semantic dependency," 2016, *arXiv:1611.01702*.

[16] D. Guo, B. Chen, R. Lu, and M. Zhou, "Recurrent hierarchical topic-guided RNN for language generation," in *Proc. Int. Conf. Mach. Learn.*, 2020, pp. 3810–3821.

[17] M. Rezaee and F. Ferraro, "A discrete variational recurrent topic model without the reparametrization trick," 2020, *arXiv:2010.12055*.

[18] S. Wiseman, S. M. Shieber, and A. M. Rush, "Learning neural templates for text generation," 2018, *arXiv:1808.10122*.

[19] H. Gong, X. Feng, B. Qin, and T. Liu, "Table-to-text generation with effective hierarchical encoder on three dimensions (row, column and time)," 2019, *arXiv:1909.02304*.

[20] R. Ye, W. Shi, H. Zhou, Z. Wei, and L. Li, "Variational template machine for data-to-text generation," 2020, *arXiv:2002.01127*.

[21] Y. Zhang, G. Wang, C. Li, Z. Gan, C. Brockett, and B. Dolan, "POINTER: Constrained progressive text generation via insertion-based generative pre-training," 2020, *arXiv:2005.00558*.

[22] S. Wiseman, S. M. Shieber, and A. M. Rush, "Challenges in data-to-document generation," 2017, *arXiv:1707.08052*.

[23] L. Yu, W. Zhang, J. Wang, and Y. Yu, "SeqGAN: Sequence generative adversarial nets with policy gradient," in *Proc. AAAI Conf. Artif. Intell.*, vol. 31, no. 1, 2017, pp. 1–7.

[24] T. Mikolov, M. Karafiát, L. Burget, J. Černockỳ, and S. Khudanpur, "Recurrent neural network based language model," in *Proc. 11th Annu. Conf. Int. Speech Commun. Assoc.*, 2010, pp. 1045–1048.

[25] L. Fang, C. Li, J. Gao, W. Dong, and C. Chen, "Implicit deep latent variable models for text generation," 2019, *arXiv:1908.11527*.

[26] Y. Kim, S. Wiseman, and A. M. Rush, "A tutorial on deep latent variable models of natural language," 2018, *arXiv:1812.06834*.

[27] L. Yang, W. Fan, and N. Bouguila, "Deep clustering analysis via dual variational autoencoder with spherical latent embeddings," *IEEE Trans. Neural Netw. Learn. Syst.*, early access, Dec. 23, 2021, doi: 10.1109/TNNLS.2021.3135460.

[28] S. Karatsiolis and C. N. Schizas, "Conditional generative denoising autoencoder," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 31, no. 10, pp. 4117–4129, Oct. 2019.

[29] F. Ye and A. G. Bors, "Deep mixture generative autoencoders," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 33, no. 10, pp. 5789–5803, Oct. 2022.

[30] S. R. Bowman, L. Vilnis, O. Vinyals, A. M. Dai, R. Jozefowicz, and S. Bengio, "Generating sentences from a continuous space," 2015, *arXiv:1511.06349*.

[31] C. Li et al., "Optimus: Organizing sentences via pre-trained modeling of a latent space," 2020, *arXiv:2004.04092*.

[32] H. Tu, Z. Yang, J. Yang, and Y. Huang, "AdaVAE: Exploring adaptive GPT-2s in variational auto-encoders for language modeling," 2022, *arXiv:2205.05862*.

[33] S. Zhao, J. Song, and S. Ermon, "InfoVAE: Information maximizing variational autoencoders," 2017, *arXiv:1706.02262*.

[34] S. Dai, Z. Gan, Y. Cheng, C. Tao, L. Carin, and J. Liu, "APo-VAE: Text generation in hyperbolic space," in *Proc. Conf. North Amer. Chapter Assoc. Comput. Linguistics, Human Lang. Technol.*, 2021, pp. 416–431.

[35] H. Fu, C. Li, X. Liu, J. Gao, A. Celikyilmaz, and L. Carin, "Cyclical annealing schedule: A simple approach to mitigating KL vanishing," 2019, *arXiv:1903.10145*.

[36] H. Shao et al., "ControlVAE: Controllable variational autoencoder," in *Proc. Int. Conf. Mach. Learn.*, 2020, pp. 8655–8664.

[37] H. Tu, Z. Yang, J. Yang, S. Zhang, and Y. Huang, "PCAE: A framework of plug-in conditional auto-encoder for controllable text generation," *Knowl.-Based Syst.*, vol. 256, Nov. 2022, Art. no. 109766.

[38] C. Cremer, X. Li, and D. Duvenaud, "Inference suboptimality in variational autoencoders," in *Proc. Int. Conf. Mach. Learn.*, 2018, pp. 1078–1086.

[39] P. Xu, J. C. K. Cheung, and Y. Cao, "On variational learning of controllable representations for text without supervision," in *Proc. Int. Conf. Mach. Learn.*, 2020, pp. 10534–10543.

[40] Y. Bao et al., "Generating sentences from disentangled syntactic and semantic spaces," 2019, *arXiv:1907.05789*.

[41] D. Rezende and S. Mohamed, "Variational inference with normalizing flows," in *Proc. Int. Conf. Mach. Learn.*, 2015, pp. 1530–1538.

[42] D. M. Blei, A. Y. Ng, and M. I. Jordan, "Latent Dirichlet allocation," *J. Mach. Learn. Res.*, vol. 3, pp. 993–1022, Mar. 2003.

[43] Y. W. Teh, M. I. Jordan, M. J. Beal, and D. M. Blei, "Hierarchical Dirichlet processes," *J. Amer. Statist. Assoc.*, vol. 101, no. 476, pp. 1566–1581, Dec. 2006.

[44] Z. Gan, C. Chen, R. Henao, D. Carlson, and L. Carin, "Scalable deep Poisson factor analysis for topic modeling," in *Proc. Int. Conf. Mach. Learn.*, 2015, pp. 1823–1832.

[45] H. Zhang, B. Chen, D. Guo, and M. Zhou, "WHAI: Weibull hybrid autoencoding inference for deep topic modeling," 2018, *arXiv:1803.01328*.

[46] C. Wang, B. Chen, S. Xiao, and M. Zhou, "Convolutional Poisson gamma belief network," in *Proc. Int. Conf. Mach. Learn.*, 2019, pp. 6515–6525.

[47] Z. Yang, Z. Hu, R. Salakhutdinov, and T. Berg-Kirkpatrick, "Improved variational autoencoders for text modeling using dilated convolutions," in *Proc. Int. Conf. Mach. Learn.*, 2017, pp. 3881–3890.

[48] S. Semeniuta, A. Severyn, and E. Barth, "A hybrid convolutional variational autoencoder for text generation," 2017, *arXiv:1702.02390*.

[49] T. Zhao, R. Zhao, and M. Eskenazi, "Learning discourse-level diversity for neural dialog models using conditional variational autoencoders," 2017, *arXiv:1703.10960*.

[50] T. Shen, J. Mueller, R. Barzilay, and T. Jaakkola, "Educating text autoencoders: Latent representation guidance via denoising," in *Proc. Int. Conf. Mach. Learn.*, 2020, pp. 8719–8729.

[51] R. Das, M. Zaheer, and C. Dyer, "Gaussian lda for topic models with word embeddings," in *Proc. 53rd Annu. Meeting Assoc. Comput. Linguistics, 7th Int. Joint Conf. Natural Lang. Process.*, vol. 1, 2015, pp. 795–804.

[52] E. Jang, S. Gu, and B. Poole, "Categorical reparameterization with Gumbel–Softmax," in *Proc. 5th Int. Conf. Learn. Represent. (ICLR)*, Toulon, France, Apr. 2017, pp. 1–13.

[53] A. S. Householder, "Unitary triangularization of a nonsymmetric matrix," *J. ACM*, vol. 5, no. 4, pp. 339–342, 1958.

[54] J. M. Tomczak and M. Welling, "Improving variational auto-encoders using householder flow," 2016, *arXiv:1611.09630*.

[55] R. Zhang, C. Li, C. Chen, and L. Carin, "Learning structural weight uncertainty for sequential decision-making," in *Proc. Int. Conf. Artif. Intell. Statist.*, 2018, pp. 1137–1146.

[56] K. Papineni, S. Roukos, T. Ward, and W.-J. Zhu, "BLEU: A method for automatic evaluation of machine translation," in *Proc. 40th Annu. Meeting Assoc. Comput. Linguistics*, 2001, pp. 311–318.

[57] J. Han Lau, T. Baldwin, and T. Cohn, "Topically driven neural language model," 2017, *arXiv:1704.08012*.

[58] A. Maas, R. E. Daly, P. T. Pham, D. Huang, A. Y. Ng, and C. Potts, "Learning word vectors for sentiment analysis," in *Proc. 49th Annu. Meeting Assoc. Comput. Linguistics, Hum. Lang. Technol.*, 2011, pp. 142–150.

[59] BNC Consortium, "British national corpus, XML edition," Oxford Text Arch. Core Collection, Univ. Oxford, Oxford, U.K., 2007. [Online]. Available: http://hdl.handle.net/20.500.12024/2554

[60] M. P. Marcus, B. Santorini, and M. A. Marcinkiewicz, "Building a large annotated corpus of English: The Penn Treebank," *Comput. Linguistics*, vol. 19, no. 2, pp. 313–330, 1993. [Online]. Available: https://aclanthology.org/J93-2004

[61] J. Pennington, R. Socher, and C. Manning, "Glove: Global vectors for word representation," in *Proc. Conf. Empirical Methods Natural Lang. Process. (EMNLP)*, 2014, pp. 1532–1543.

[62] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, 1997.

[63] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," 2014, *arXiv:1412.6980*.

[64] X. Fang, J. Li, L. Shang, X. Jiang, Q. Liu, and D.-Y. Yeung, "Controlled text generation using dictionary prior in variational autoencoders," in *Findings of the Association for Computational Linguistics: ACL 2022*. Dublin, Ireland: Association for Computational Linguistics, May 2022, pp. 97–111. [Online]. Available: https://aclanthology.org/2022.findings-acl.10, doi: 10.18653/v1/2022.findings-acl.10.

[65] Y. Zhu et al., "Texygen: A benchmarking platform for text generation models," in *Proc. 41st Int. ACM SIGIR Conf. Res. Develop. Inf. Retr.*, Jun. 2018, pp. 1097–1100.

[66] X. Gu, K. Cho, J.-W. Ha, and S. Kim, "DialogWAE: Multimodal response generation with conditional Wasserstein auto-encoder," 2018, *arXiv:1805.12352*.

[67] J. Chang, S. Gerrish, C. Wang, J. L. Boyd-Graber, and D. M. Blei, "Reading tea leaves: How humans interpret topic models," in *Proc. Adv. Neural Inf. Process. Syst.*, 2009, pp. 288–296.

[68] L. Van der Maaten and G. Hinton, "Visualizing data using t-SNE," *J. Mach. Learn. Res.*, vol. 9, no. 11, pp. 1–27, 2008.

**Haoqin Tu** received the B.S. degree from the School of Mathematics, Fuzhou University, Fuzhou, Fujian, China, in 2021. He is currently pursuing the M.S. degree with the University of Chinese Academy of Sciences, Beijing, China.

He is a Student Research Intern at the Department of Electronic Engineering, Tsinghua University, Beijing. His current research interests include natural language processing and multimedia processing.

**Zhongliang Yang** received the Ph.D. degree from the Department of Electronic Engineering, Tsinghua University, Beijing, China, in 2020.

He is currently an Associate Professor with the School of Cyberspace Security, Beijing University of Posts and Telecommunications, Beijing. His research interests include multimedia network security, steganography and steganlysis, and artificial intelligence.

**Jinshuai Yang** received the B.S. degree in electronic engineering from Tsinghua University, Beijing, China, in 2020, where he is currently pursuing the Ph.D. degree in information and communication engineering.

His current research interests include natural language processing, steganography, and steganalysis.

**Linna Zhou** is currently a Full Professor and a Doctoral Supervisor with the School of Cyberspace Security, Beijing University of Posts and Telecommunications, Beijing, China. Her research interests include natural language processing and computer vision. She has published many high-level academic papers and four monographs in related research fields.

Prof. Zhou was a recipient of the State Council Special Allowance and the Head of the National Key Areas Innovation Team. She has been selected into the National Million Talents Project and awarded the honorary title of "young and middle-aged experts with outstanding contributions."

**Yongfeng Huang** (Senior Member, IEEE) received the Ph.D. degree in computer science and engineering from the Huazhong University of Science and Technology, Wuhan, China, in 2000.

He is currently a Professor with the Department of Electronic Engineering, Tsinghua University, Beijing, China. He has published six books and more than 100 research articles on computer networks, multimedia communications, and security. His research interests include multimedia network security, steganography and steganlysis, big data mining, and next-generation Internet.

## Appendix

We do the mathematical proof of the reconstruction process in the topic modeling part and the separation of KL divergence of two modeling parts in this section.

### *A. Reconstruction Process in the Topic Modeling Part*

We assume $\boldsymbol{X}$ is the input text data, $\boldsymbol{\alpha}$ is the document-level topic parameter, $\boldsymbol{Y}$ is the output of the topic modeling component. Then the reconstruction of the topic modeling part is:

$$\begin{aligned}
&p(\boldsymbol{X} \mid \boldsymbol{\alpha}) = p(\boldsymbol{Y}) = \\
&\int_{z_t}\int_{z_s} p(\boldsymbol{z_t}) \left(\prod_{i=1}^{m} p(y_i \mid \boldsymbol{z_t}) p(\boldsymbol{z_t} \mid \boldsymbol{z_s}) p(\boldsymbol{z_s})\right) d\boldsymbol{z_s} d\boldsymbol{z_t} \\
&= \int_{z_t}\int_{z_s} p(\boldsymbol{z_t}) \left(\prod_{i=1}^{m} p(y_i, \boldsymbol{z_s} \mid \boldsymbol{z_t})\right) d\boldsymbol{z_s} d\boldsymbol{z_t} \\
&= \int_{z_t}\int_{z_s} p(\boldsymbol{z_t}) p(\boldsymbol{Y}, \boldsymbol{z_s} \mid \boldsymbol{z_t}) d\boldsymbol{z_s} d\boldsymbol{z_t} \\
&= \int_{z_t}\int_{z_s} p(\boldsymbol{Y}, \boldsymbol{z_s}, \boldsymbol{z_t}) d\boldsymbol{z_s} d\boldsymbol{z_t} \\
&= \int_{z_t}\int_{z_s} p(\boldsymbol{X}, \boldsymbol{z_s}, \boldsymbol{z_t} \mid \boldsymbol{\alpha}) d\boldsymbol{z_s} d\boldsymbol{z_t},
\end{aligned} \tag{14}$$

The relation between $\boldsymbol{X}$ and $\boldsymbol{Y}$ is $\boldsymbol{Y} = \boldsymbol{X} \mid \boldsymbol{\alpha}$. The second equation above can stand because of the approximation method of the marginal probability of a word in documents: $p(y_i \mid \boldsymbol{z_t}) p(\boldsymbol{z_t} \mid \boldsymbol{z_s}) p(\boldsymbol{z_s}) = p(y_i \mid \boldsymbol{z_t}) p(\boldsymbol{z_t}, \boldsymbol{z_s}) = p(y_i, \boldsymbol{z_s} \mid \boldsymbol{z_t})$.

### *B. From the Overall KL to Separate Modes*

We will give a more intuitive explanation of the derivation of KL terms from separate modeling component (sequence and topic) in FET-LM. The overall KL term of FET-LM model under the paradigm of two VAEs can be modeled as:

$$\mathbb{D}_{\mathrm{KL}}(q(\boldsymbol{z_t}, \boldsymbol{z_s} \mid \boldsymbol{X}) \| p(\boldsymbol{z_t}, \boldsymbol{z_s})), \tag{15}$$

where we treat two different latent representations as one and calculate its regularization penalty using KL divergence. However, Eq.(15) can be factorized into two terms w.r.t. the sequence and topic latent codes respectively, that is:

$$\begin{aligned}
&\mathbb{D}_{\mathrm{KL}}(q(\boldsymbol{z_t}, \boldsymbol{z_s} \mid \boldsymbol{X}) \| p(\boldsymbol{z_t}, \boldsymbol{z_s})) \\
&= q(\boldsymbol{z_t}, \boldsymbol{z_s} \mid \boldsymbol{X}) \log \left[q(\boldsymbol{z_t}, \boldsymbol{z_s} \mid \boldsymbol{X})\right] - \log \left[p(\boldsymbol{z_t}, \boldsymbol{z_s})\right] \\
&= q(\boldsymbol{z_t}, \boldsymbol{z_s} \mid \boldsymbol{X}) \log \left[\frac{q(\boldsymbol{z_t}, \boldsymbol{z_s}, \boldsymbol{X})}{q(\boldsymbol{z_s}, \boldsymbol{X})} \cdot \frac{q(\boldsymbol{z_s}, \boldsymbol{X})}{q(\boldsymbol{X})}\right] \\
&- q(\boldsymbol{z_t}, \boldsymbol{z_s} \mid \boldsymbol{X}) \log \left[\frac{p(\boldsymbol{z_t}, \boldsymbol{z_s})}{p(\boldsymbol{z_t})} \cdot p(\boldsymbol{z_t})\right] \\
&= q(\boldsymbol{z_t}, \boldsymbol{z_s} \mid \boldsymbol{X}) \{\log \left[q(\boldsymbol{z_t} \mid \boldsymbol{z_s}, \boldsymbol{X})\right] - \log \left[p(\boldsymbol{z_t} \mid \boldsymbol{z_s})\right]\} \\
&+ q(\boldsymbol{z_s} \mid \boldsymbol{X}) \{\log \left[q(\boldsymbol{z_s} \mid \boldsymbol{X})\right] - \log p(\boldsymbol{z_s})\} \\
&= q(\boldsymbol{z_s} \mid \boldsymbol{X}) q(\boldsymbol{z_t} \mid \boldsymbol{z_s}, \boldsymbol{X}) \log \frac{q(\boldsymbol{z_t} \mid \boldsymbol{z_s}, \boldsymbol{X})}{p(\boldsymbol{z_t} \mid \boldsymbol{z_s})} \\
&+ q(\boldsymbol{z_s} \mid \boldsymbol{X}) \log \frac{q(\boldsymbol{z_s} \mid \boldsymbol{X})}{p(\boldsymbol{z_s})} \\
&= \underbrace{\mathbb{E}_{q(\boldsymbol{z_t} \mid \boldsymbol{X})} \left[\mathbb{D}_{\mathrm{KL}}(q(\boldsymbol{z_t} \mid \boldsymbol{X}, \boldsymbol{z_s}) \| p(\boldsymbol{z_t} \mid \boldsymbol{z_s}))\right]}_{\text{KL Term in Topic Modeling Component}} \\
&+ \underbrace{\mathbb{D}_{\mathrm{KL}}(q(\boldsymbol{z_s} \mid \boldsymbol{X}) \| p(\boldsymbol{z_s}))}_{\text{KL Term in Sequence Modeling Component}} .
\end{aligned} \tag{16}$$

By replacing sequence latent variable $\boldsymbol{z_s}$ in its posterior with $\boldsymbol{z_{s(0)}}$ and $\boldsymbol{z_s}$ in its prior with $\boldsymbol{z_{s(K)}}$, we can approximate this decomposition under the modeling process of normalizing flow, which leads to Eq (11) in the paper. The third equation in Eq (16) can stand because we replace $q(\boldsymbol{z_t}, \boldsymbol{z_s} \mid \boldsymbol{X})$ with $q(\boldsymbol{z_s} \mid \boldsymbol{X})$ in the second term for the third equation. At last, we discover that the overall KL term of the system is well approximated by two distinct KL penalties related to components in the FET-LM model.

### *C. Generated Topics*

For topic word generation, we used the decoder of the topic modeling part to produce the probability of each token in a corpora, then sorted words with the highest five probabilities as top-5 topic word output. We selected nine channels from FET-LM models with 50 topic latent dimensions. And generated top-5 topic words from them severally. Results are shown in Table XII.

### *D. Style Transfer Generation and Interpolated Sentences*

For well-expressive attribute representation spaces, we expect they contain distinct attributes and can be easily manipulated. For sentence generation with transferred styles, we traversed the value in one latent dimension of latent variables from $-10.0$ to $10.0$ by a step size of $2.0$. Results in Table XIII show a transformation from positive sentiment to relatively negative (i.e., with negative expressions "n't been ... twice", "overpriced"). For the interpolation task. We used a linear interpolation strategy, this process can be specified as follows:

1) Given two samples $x_i, x_j$ from train set.
2) Obtain their sequence latent code and topic latent code respectively $(\boldsymbol{z_{s(i)}}, \boldsymbol{z_{t(i)}}), (\boldsymbol{z_{s(j)}}, \boldsymbol{z_{t(j)}})$.
3) For both types of latent variables we use linear interpolation $\boldsymbol{z}_{\text{type}} = \boldsymbol{z}_{\text{type}(\boldsymbol{i})} \cdot (1 - \tau) + \boldsymbol{z}_{\text{type}(\boldsymbol{j})} \cdot \tau$ where $\boldsymbol{z}_{\text{type}} \in \{\boldsymbol{z_s}, \boldsymbol{z_t}\}$ and $\tau$ increases from $0$ to $1$ by a step size of $0.2$.

We can see there is maintenance from the original text key phrases or structure (e.g., "the company", "lawmakers are consider", inverted form) and semantics (e.g., positive, business, law) as well as a transformation between two given examples. We can observe smooth and sensible interpolation results for almost arbitrary input pairs. This demonstrates our FET-LM model learns meaningful latent spaces.

### *E. Ablation Study of Discriminator Weight w.r.t. PPL Results*

We analysis the effects of hyper-parameters $\lambda_D, \lambda_{\text{info}}$. We conducted experiments with varied $\lambda_D$ from [0.0, 0.1, 0.3, 0.5, 0.8, 1.0] in Fig. 6 w.r.t. text perplexity (PPL) and document-level entropy tasks on APNEWS dataset respectively. We find that, as the $\lambda_D$ increases, the PPL value of FET-LM generally increases, while the entropy value decreases. This yields a worse language modeling ability but better topic modeling ability of our model, also an apparent trade-off between the model's PPL and entropy values. Overall, we chose $\lambda_D = 0.5$ with a good trade-off between PPL and entropy for our model.

TABLE XII
TOP-5 TOPIC WORDS FROM NINE TOPICS GENERATED FROM 50 TOPIC FET-LM MODELS.

| Dataset | #1 | #2 | #3 | #4 | #5 | #6 | #7 | #8 | #9 |
|---|---|---|---|---|---|---|---|---|---|
| **APNEWS** | gay | iraq | 57-year | plane | tea | rain | deputies | mark | museum |
| | marriage | soldier | 19-year | crashed | gop | rains | deputy | staff | art |
| | anti | syria | collision | miles | nomination | snow | commissioners | clinton | festival |
| | ruling | troops | 21-year | wildfire | democrat | unemployment | maricopa | lead | music |
| | congress | forces | tractor | engine | challenger | storms | patrol | elections | zoo |
| **IMDB** | reviewers | poorly | debut | oscar | finished | toronto | happened | twice | grade |
| | ridiculous | cinematography | finest | terrific | remote | independent | screening | yesterday | sub |
| | total | romance | beautifully | poorly | aged | maker | makers | funniest | flicks |
| | considering | dialogue | stage | independent | maker | oscar | camera | cable | fu |
| | highly | directing | romance | talented | pre | debut | reviewers | viewed | kung |
| **BNC** | yesterday | council | conservation | voice | award | africa | international | england | environmental |
| | night | britain | environmental | yesterday | pounds | pacific | east | cup | pollution |
| | today | environmental | pollution | night | ref | council | european | voice | conservation |
| | young | meeting | council | daily | research | asia | europe | britain | council |
| | just | title | species | post | holder | east | british | league | environment |
| **PTB** | cost | composite | mortgages | gains | futures | nov | benchmark | tuesday | nasdaq |
| | fiscal | counter | adjustable | rise | traders | oct | points | notes | counter |
| | spending | volume | capped | inflation | short | priced | priced | october | s&p |
| | budget | ounce | yields | orders | gains | mature | treasury | september | activity |
| | senate | pence | rise | percentage | selling | dec | point | oct | decline |
| **Yelp15** | casino | avec | massage | beers | matcha | min | spa | cons | rooms |
| | hotels | c'est | pedicure | buffet | milk | mins | tub | pros | suite |
| | strip | des | gel | tap | bagel | tip | shower | buffet | amenities |
| | mgm | en | nail | burgers | vanilla | dirty | pool | rooms | stayed |
| | rooms | que | polish | bartender | cupcake | 40 | massage | rental | pool |

TABLE XIII
TEXT STYLE TRANSFER GENERATION FROM POSITIVE TO SLIGHTLY NEGATIVE BY TRAVERSING LEARNED TOPIC REPRESENTATIONS.

| | |
|---|---|
| **Int. 1** | •have been here twice , and i have never had a bad experience . i had the chicken salad with garlic knots . the salad was delicious ! ! ! ! ! ! ! ! ! ! ! |
| **Int. 2** | •i have been here twice , and i have never had a bad experience . i had the shrimp taco salad , which was delicious . i will be back ! ! ! ! ! ! ! ! ! ! |
| **Int. 3** | •i have been here twice and have never been disappointed . the food was delicious , the fish tacos were delicious . i had the shrimp tacos , and the chicken was cooked perfectly . |
| **Int. 4** | •i have been to this location twice and have never been disappointed . the service is very friendly and helpful . |
| **Int. 5** | •i have n't been to this location twice . the <unk>is very nice and helpful . the <unk>is located in the middle of the strip mall . |
| **Int. 6** | •i have n't been to this location twice . pros : <unk>and <unk>. the <unk>was very nice and the service was great . i was in the area for a few days and it was n't a bad experience . |
| **Int. 7** | •i have n't been to this location twice . the <unk>was very nice and the service was great . i was n't sure what to expect . |
| **Int. 8** | •i have n't been to this location twice . i would have given a lot of money in the future , but i 'm not sure why the prices are reasonable . |
| **Int. 9** | •i think it 's a bit overpriced . pros : <unk>: |

### *F. Full Results of BLEU*

We used the benchmark tool Texygen [65] to do all the BLEU-related calculations. We show results of our model only with or without the discriminator, which we believe is more important for the token-level optimization. This is because the mutual information term is directly optimized in the topic latent space $\boldsymbol{z_t}$, rather than in sequence embedding $\boldsymbol{z_s}$ or token level like the discriminator does. From the full results in Table XVI, we can see that our model outperforms all baselines in *test*-BLEU metric, yet is only superior to other models on *self*-BLEU under B-2 in major cases. This phenomenon demonstrates that the proposed model is qualified to produce texts with high quality, but has difficulty in generating texts with high diversity. Nevertheless, the overall metric BLEU-F1 shows the superiority of the FET-LM model in a well-weighted trade-off between text quality and diversity.

TABLE XIV
GENERATED SENTENCES BY INTERPOLATING LATENT CODES.

| Type | Sentences |
|---|---|
| **Org. I** | **•the company and its executives deny the charges** |
| **Rec. I** | •the company had been working with the state and financial services and the government 's plan |
| **Int. 1** | •the company had no comment on the other hand and the state department said |
| **Int. 2** | •the company wants to keep the entire computer system says the agency |
| **Int. 3** | •these guys are a good idea he says |
| **Int. 4** | •these guys is an important and financial services he says |
| **Rec. II** | •you have a lot more efficient than he says |
| **Org. II** | **•our doors are open an nbc spokesman says** |

TABLE XV
GENERATED SENTENCES BY INTERPOLATING LATENT CODES.

| Type | Sentences |
|---|---|
| **Org. 1** | **•lawmakers are considering restrictions on harvesting a hawaii seafood <unk> known as <unk>.** |
| **Rec. 1** | •lawmakers are considering a bill that would link at least two dozen dogs dead inside a local airport . |
| **Int. 1** | •lawmakers are considering a bill that would link the south carolina town of marine corps on sunday night . |
| **Int. 2** | •the state 's government will be held on a las vegas strip - based weapons ring that killed in the u.s . house , but it does n't have a chance . |
| **Int. 3** | •the city of a florida man who died after being held by a fellow military veterans affairs in the nation 's largest valley . |
| **Int. 4** | •the man who died in a shooting that killed a tennessee valley business . |
| **Rec. 2** | •the man who shot a man in a downtown philadelphia house is now that he has received a plea deal . |
| **Org. 2** | **•a man who barricaded himself in his omaha home has surrendered without incident .** |

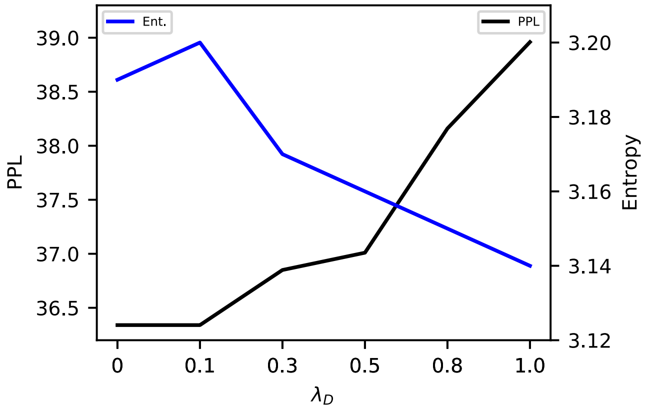


Fig. 6. Ablation analysis of $\lambda_D$ w.r.t. text perplexity (PPL) and document-level entropy on APNEWS dataset.

TABLE XVI
FULL BLEU RESULT IN TERMS OF *test*-BLEU, *self*-BLEU AND BLEU-F1 SCORES.

| **Metrics** | **Methods** | **APNEWS** | | | | **IMDB** | | | | **BNC** | | | | **PTB** | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | **B-2** | **B-3** | **B-4** | **B-5** | **B-2** | **B-3** | **B-4** | **B-5** | **B-2** | **B-3** | **B-4** | **B-5** | **B-2** | **B-3** | **B-4** | **B-5** |
| ***test*-BLEU↑** | **VAE** | 0.564 | 0.278 | 0.192 | 0.122 | 0.597 | 0.315 | 0.219 | 0.147 | 0.479 | 0.266 | 0.169 | 0.117 | 0.5215 | 0.3633 | 0.2642 | 0.1728 |
| | **VAE+HF** | 0.570 | 0.279 | 0.195 | 0.123 | 0.610 | 0.322 | 0.221 | 0.147 | 0.483 | 0.270 | 0.169 | 0.110 | 0.5565 | 0.3616 | 0.2529 | 0.1653 |
| | **TGVAE(F=10, T=10)** | 0.584 | 0.327 | 0.202 | 0.126 | 0.621 | 0.357 | 0.223 | 0.159 | 0.518 | 0.283 | 0.173 | 0.119 | - | - | - | - |
| | **TGVAE(F=10, T=30)** | 0.627 | 0.335 | 0.207 | 0.131 | 0.655 | 0.369 | 0.243 | 0.165 | 0.528 | 0.291 | 0.182 | 0.119 | - | - | - | - |
| | **TGVAE(F=10, T=50)** | 0.629 | 0.340 | 0.210 | 0.132 | 0.652 | 0.372 | 0.239 | 0.160 | 0.535 | 0.290 | 0.188 | 0.120 | - | - | - | - |
| | **Ours(F=10, T=10)** | 0.6512 | 0.3862 | 0.2358 | 0.1458 | 0.7202 | 0.4505 | 0.2470 | 0.1404 | 0.6997 | 0.5947 | 0.4934 | 0.3327 | 0.6824 | 0.4847 | 0.3564 | 0.2307 |
| | **Ours(F=10, T=30)** | 0.6434 | 0.3776 | 0.2374 | 0.1468 | 0.7037 | 0.4347 | 0.2566 | 0.1529 | 0.6791 | 0.5473 | 0.4502 | 0.3151 | 0.6705 | 0.4779 | 0.3438 | 0.2070 |
| | **Ours(F=10, T=50)** | **0.6757** | 0.3983 | 0.2432 | **0.1514** | **0.7542** | **0.4753** | **0.2755** | **0.1620** | **0.7681** | **0.6610** | **0.5672** | 0.4176 | **0.6924** | **0.5076** | 0.3733 | 0.2408 |
| | **Ours w/o Dis (F=10, T=50)** | 0.6596 | **0.4100** | **0.2497** | 0.1464 | 0.7447 | 0.4637 | 0.2678 | 0.1502 | 0.7316 | 0.6234 | 0.5292 | **0.4215** | 0.6484 | 0.4587 | 0.3297 | 0.2028 |
| | **Ours(F=5, T=50)** | 0.6449 | 0.3801 | 0.2241 | 0.1335 | 0.7136 | 0.4323 | 0.2444 | 0.1399 | 0.7397 | 0.6422 | 0.5521 | 0.3896 | 0.6599 | 0.4710 | 0.3407 | 0.2175 |
| | **Ours w/o Dis (F=5, T=50)** | 0.6531 | 0.3845 | 0.2204 | 0.1335 | 0.7221 | 0.4456 | 0.2498 | 0.1382 | 0.7283 | 0.6247 | 0.5323 | 0.4157 | 0.6870 | 0.5064 | **0.3889** | **0.2604** |
| | **Ours(F=20, T=50)** | 0.6558 | 0.3809 | 0.2187 | 0.1260 | 0.7374 | 0.4660 | 0.2543 | 0.1411 | 0.6744 | 0.5660 | 0.4818 | 0.3670 | 0.6790 | 0.5001 | 0.3661 | 0.2376 |
| | **Ours w/o Dis (F=20, T=50)** | 0.6522 | 0.3943 | 0.2274 | 0.1311 | 0.7255 | 0.4305 | 0.2418 | 0.1403 | 0.6538 | 0.5324 | 0.4265 | 0.2722 | 0.6391 | 0.4486 | 0.3149 | 0.1836 |
| ***self*-BLEU↓** | **VAE** | 0.866 | 0.531 | 0.233 | - | 0.891 | 0.632 | 0.275 | - | 0.851 | 0.510 | 0.163 | - | 0.8737 | 0.5411 | 0.2952 | 0.2359 |
| | **VAE+HF** | 0.873 | 0.552 | 0.219 | - | 0.902 | 0.648 | 0.262 | - | 0.845 | 0.520 | 0.163 | - | 0.8649 | **0.4720** | **0.3162** | **0.2181** |
| | **TGVAE(F=10, T=10)** | 0.839 | 0.512 | 0.172 | - | 0.889 | 0.577 | 0.242 | - | 0.829 | 0.488 | 0.151 | - | - | - | - | - |
| | **TGVAE(F=10, T=30)** | 0.811 | 0.478 | 0.157 | - | 0.850 | 0.560 | 0.231 | - | 0.806 | 0.473 | 0.150 | - | - | - | - | - |
| | **TGVAE(F=10, T=50)** | 0.808 | **0.476** | **0.150** | - | 0.842 | **0.559** | **0.227** | - | 0.793 | **0.469** | **0.150** | - | - | - | - | - |
| | **Ours(F=10, T=10)** | 0.7396 | 0.5659 | 0.4146 | 0.2927 | 0.7948 | 0.5950 | 0.3989 | 0.2423 | 0.8191 | 0.7718 | 0.7272 | 0.6798 | 0.7882 | 0.6598 | 0.5372 | 0.4149 |
| | **Ours(F=10, T=30)** | **0.7091** | 0.5093 | 0.3457 | **0.2173** | **0.7783** | 0.5674 | 0.3729 | 0.2298 | 0.8130 | 0.7358 | 0.6644 | 0.5924 | **0.7575** | 0.6009 | 0.4707 | 0.3413 |
| | **Ours(F=10, T=50)** | 0.7344 | 0.5373 | 0.3839 | 0.2309 | 0.7911 | 0.5878 | 0.3827 | 0.2346 | **0.7851** | 0.7407 | 0.6924 | 0.6297 | 0.7695 | 0.6350 | 0.5092 | 0.3806 |
| | **Ours w/o Dis (F=10, T=50)** | 0.7289 | 0.5635 | 0.4212 | 0.2792 | 0.8004 | 0.5973 | 0.3967 | 0.2475 | 0.7882 | 0.7417 | 0.6974 | 0.6420 | 0.7798 | 0.6305 | 0.4961 | 0.3663 |
| | **Ours(F=5, T=50)** | 0.7434 | 0.5599 | 0.3863 | 0.2590 | 0.7834 | 0.5745 | 0.3795 | 0.2392 | 0.8033 | 0.7565 | 0.7101 | 0.6524 | 0.7641 | 0.6097 | 0.4792 | 0.3546 |
| | **Ours w/o Dis (F=5, T=50)** | 0.7588 | 0.5891 | 0.4215 | 0.2773 | 0.7943 | 0.5796 | 0.3648 | **0.2107** | 0.8142 | 0.7549 | 0.6942 | 0.6306 | 0.7678 | 0.6320 | 0.5178 | 0.3854 |
| | **Ours(F=20, T=50)** | 0.7516 | 0.5799 | 0.4147 | 0.2768 | 0.8109 | 0.6008 | 0.3914 | 0.2443 | 0.8107 | 0.7552 | 0.7097 | 0.6650 | 0.7606 | 0.6127 | 0.4810 | 0.3461 |
| | **Ours w/o Dis (F=20, T=50)** | 0.7512 | 0.5735 | 0.4118 | 0.2708 | 0.7949 | 0.5723 | 0.3674 | 0.2262 | 0.8312 | 0.7788 | 0.7267 | 0.6728 | 0.7764 | 0.6442 | 0.5257 | 0.4055 |
| **BLEU-F1↑** | **VAE** | 0.2166 | 0.3491 | 0.3071 | - | 0.1843 | 0.3394 | 0.3364 | - | 0.2273 | 0.3448 | 0.2812 | - | 0.2033 | 0.4055 | 0.3843 | 0.2819 |
| | **VAE+HF** | 0.2077 | 0.3439 | 0.3121 | - | 0.1689 | 0.3363 | 0.3401 | - | 0.2242 | 0.3456 | 0.2809 | - | 0.2174 | 0.4292 | 0.3692 | 0.2729 |
| | **TGVAE(F=10, T=10)** | 0.2524 | 0.3916 | 0.3248 | - | 0.1883 | 0.3872 | 0.3446 | - | 0.2571 | 0.3645 | 0.2874 | - | - | - | - | - |
| | **TGVAE(F=10, T=30)** | 0.2904 | 0.4081 | 0.3324 | - | 0.2441 | 0.4014 | 0.3693 | - | 0.2837 | 0.3750 | 0.2998 | - | - | - | - | - |
| | **TGVAE(F=10, T=50)** | 0.2942 | 0.4124 | 0.3368 | - | 0.2544 | 0.4036 | 0.3651 | - | 0.2985 | **0.3751** | 0.3079 | - | - | - | - | - |
| | **Ours(F=10, T=10)** | 0.3720 | 0.4088 | 0.3362 | 0.2418 | 0.3193 | 0.4265 | 0.3501 | 0.2369 | 0.2875 | 0.3299 | 0.3513 | 0.3264 | 0.3233 | 0.3998 | 0.4027 | 0.3309 |
| | **Ours(F=10, T=30)** | **0.4007** | 0.4268 | 0.3484 | 0.2473 | **0.3371** | 0.4337 | 0.3642 | 0.2551 | 0.2933 | 0.3564 | 0.3845 | 0.3554 | **0.3562** | 0.4350 | 0.4168 | 0.3149 |
| | **Ours(F=10, T=50)** | 0.3813 | **0.4281** | 0.3487 | **0.2530** | 0.3272 | **0.4415** | **0.3809** | **0.2673** | **0.3358** | 0.3725 | **0.3989** | **0.3925** | 0.3459 | 0.4246 | 0.4241 | 0.3468 |
| | **Ours w/o Dis (F=10, T=50)** | 0.3842 | 0.4228 | **0.3490** | 0.2434 | 0.3148 | 0.4310 | 0.3709 | 0.2505 | 0.3284 | 0.3653 | 0.3850 | 0.3872 | 0.3287 | 0.4093 | 0.3986 | 0.3072 |
| | **Ours(F=5, T=50)** | 0.3671 | 0.4079 | 0.3283 | 0.2262 | 0.3323 | 0.4289 | 0.3507 | 0.2364 | 0.3108 | 0.3531 | 0.3802 | 0.3674 | 0.3475 | 0.4269 | 0.4119 | 0.3254 |
| | **Ours w/o Dis (F=5, T=50)** | 0.3523 | 0.3973 | 0.3193 | 0.2255 | 0.3203 | 0.4326 | 0.3586 | 0.2352 | 0.2960 | 0.3521 | 0.3884 | 0.3912 | 0.3471 | 0.4263 | **0.4335** | **0.3658** |
| | **Ours(F=20, T=50)** | 0.3603 | 0.3996 | 0.3185 | 0.2145 | 0.3010 | 0.4300 | 0.3587 | 0.2379 | 0.2956 | 0.3418 | 0.3623 | 0.3503 | 0.3540 | **0.4364** | 0.4293 | 0.3486 |
| | **Ours w/o Dis (F=20, T=50)** | 0.3602 | 0.4098 | 0.3280 | 0.2222 | 0.3197 | 0.4290 | 0.3498 | 0.2376 | 0.2683 | 0.3125 | 0.3330 | 0.2972 | 0.3313 | 0.3969 | 0.3785 | 0.2806 |