Putting Back the Stops: Integrating Syntax with Neural Topic Models

Mayank Nagda , Sophie Fellenz
RPTU Kaiserslautern-Landau, Germany
nagda@cs.uni-kl.de , fellenz@cs.uni-kl.de

Abstract

Syntax and semantics are two key concepts for language understanding. Topic models typically represent the semantics of a text corpus, while removing syntactic information during preprocessing. Without preprocessing, the generated topics become uninterpretable because the syntactic words dominate generated topics. To learn interpretable topics while keeping valuable syntactic information, we propose a novel framework that can simultaneously learn both syntactic and semantic topics from the corpus without requiring any preprocessing. A context network leverages textual dependencies to distinguish between syntactic and semantic words, while a composite VAE topic model learns two sets of topics. We demonstrate on seven datasets that our proposed method effectively captures both syntactic and semantic representations of a corpus while outperforming state-of-the-art neural topic models and statistical topic models in terms of topic quality.

1 Introduction

Topic models [Blei *et al.*, 2003] are a family of generative models that estimate probability distributions over a vocabulary of words, called topics. Topic models are used in many application areas, such as political science [Grimmer and Stewart, 2013], bioinformatics [Liu *et al.*, 2016], and digital humanities [Meeks and Weingart, 2012]. The best-known model of this kind is latent Dirichlet allocation (LDA) [Blei *et al.*, 2003], a generative statistical model representing each document as a mixture of different topics. More recently, neural topic models (NTMs) have been introduced based on variational auto-encoders (VAEs), allowing documents to be represented as a product of different topics [Srivastava and Sutton, 2017], and leading to better overall topic quality.

With the existing topic models, removing stop words and high-frequency words as preprocessing is crucial to generate useful topics [Blei *et al.*, 2003; Griffiths *et al.*, 2004]. Often, but not always, these words are syntactic words such as 'the', 'and', 'to', 'for' etc. For example, a corpus on cooking recipes also might have frequent words such as 'ingredients' or 'cups' that are removed because they would otherwise dominate all topics. In another corpus, these words might be important to identify a topic on cooking in general. Because of this, stop word lists are often manually adapted for each corpus, leading to inconsistent benchmarking [Hoyle *et al.*, 2021], and removing a large part of (not only syntactic) information from the model.

The main idea of our model is that syntactic dependencies tend to be short-range, while semantic dependencies tend to be long-range. In essence, the syntactic structure of a sentence often involves relationships between adjacent or closely positioned words or phrases. On the other hand, semantic relationships tend to extend beyond immediate word pairs, encompassing more distant words or even spanning across entire sentences or paragraphs. This distinction is supported by numerous results in the domain of cognitive linguistics and neuroscience [Neville *et al.*, 1992; Brown, 1973; Redington *et al.*, 1998] and also served as the basis for several ML approaches in the past [Wu and Khudanpur, 1999; Griffiths *et al.*, 2004; Wang *et al.*, 2005; Yingzhen and Mandt, 2018]. For long-range dependencies, a BoW representation of the documents is sufficient in a topic model, whereas for short-range dependencies we use a simple language model. By combining the two components we can automatically distinguish semantic and syntactic words and learn topics for both categories. This could be the basis for future models that automatically disentangle more fine-grained syntactical categories or indicators for stylistic properties of a text.

In summary, the major contributions of this paper are as follows [1]:

- We introduce a VAE-based neural topic model that can generate syntactic and semantic topics from a document corpus in Section 3.2.
- We demonstrate in Section 4.4 the utility of syntactic topics to better understand the data. The proposed model also achieves state-of-the-art topic quality and is successful in detecting stop words without predefined lists (Section 4.5 and Section 4.6).
- The proposed framework can potentially accommodate all VAE-based NTMs, indicating its versatility and applicability. The extent of this generalizability is demonstrated in Section 4.

[1] Code and supplementary material available at: https://github.com/mayanknagda/integrating-syntax-with-neural-topic-models

2 Related Work

In this section, we first give an overview of syntactic and semantic generative models, then review work on combining syntactic and semantic generative models.

2.1 Syntactic Generative Models

Syntactic dependencies refer to relationships between words in a sentence that are short-range, spanning multiple words within a sentence but not extending beyond it. An n-gram language model (LM), a type of probabilistic Markov model of order $n-1$, can be used to predict the next word in a sequence based on the previous $n-1$ words [Broder *et al.*, 1997]. These models are particularly useful for capturing short-range syntactic dependencies and producing grammatically correct text, but they may not generate meaningful and coherent text [Cavnar *et al.*, 1994]. The probability of a word sequence w using an n-gram model is $P(w) = \prod_{i=1}^{L} P(w_i|w_{i-(n-1)}, \ldots, w_{i-1})$, where each word is predicted by the $n-1$ preceeding words.

Neural networks have been investigated for learning n-gram LMs [Chelba *et al.*, 2017]. Recent attempts use recurrent neural networks (RNNs) and Transformers to learn syntax-sensitive dependencies [Bhatt *et al.*, 2020; Currey and Heafield, 2019; Duan *et al.*, 2019]. However, RNNs and Transformers are not restricted to a context of size n, and also capture long-range dependencies, making it difficult to distinguish between syntactic and semantic words.

Our goal is to develop a model that solely focuses on modeling syntactic dependencies and minimizes semantic dependencies. To achieve this, we have adopted the n-gram language model proposed by [Chelba *et al.*, 2017].

2.2 Semantic Generative Models

In contrast to syntactic models, topic models use a Bag of Words (BoW) input instead of sequential input, as the word order is usually not relevant to the broader topics of a text. Statistical topic models, such as LDA [Blei *et al.*, 2003] have separate parameters for each document, whereas NTMs learn a function amortized over all documents to map each document to its topic distribution [Miao *et al.*, 2016].

NTMs [Zhao *et al.*, 2021] can be divided into topic models based on VAEs [Srivastava and Sutton, 2017], generative adversarial networks (GANs) [Wang *et al.*, 2020], DocNADE [Larochelle and Lauly, 2012], and other frameworks (e.g. based on pretrained language models [Thompson and Mimno, 2020]), where the VAE-based NTMs are by far the most wide-spread models, which we also use as a basis for our work.

The first VAE-based NTM was NVDM [Miao *et al.*, 2016] which used a Gaussian prior. ProdLDA [Srivastava and Sutton, 2017] was the first NTM to use a Dirichlet prior instead, leading to superior topic coherence. Other models improved on the reparameterization of the Dirichlet prior, among them D-VAE [Burkhardt and Kramer, 2019], implicit reparameterization gradients [Figurnov *et al.*, 2018], inverse CDF [Joo *et al.*, 2020], Weibull distribution as a replacement [Zhang *et al.*, 2018], and D-VAE with pathwise gradients [Hoyle *et al.*, 2021]. Based on several of these mentioned base models, extensions have been proposed. ETM [Dieng *et al.*, 2020] extends the model by incorporating word embeddings, thereby improving performance for unknown words. Other extensions include reinforcement learning [Gui *et al.*, 2019], adversarial learning [Wang *et al.*, 2020], contrastive learning [Nguyen and Luu, 2021], and hierarchical VAEs [Li *et al.*, 2022]. In our experiments we compare to D-VAE and ETM, the current state-of-the-art models [Hoyle *et al.*, 2021].

2.3 Combining Syntactic and Semantic Generative Models

There are various approaches for combining topic models with sequential models to take word order into account. [Gupta *et al.*, 2019] extend DocNADE with LSTMs. [Nallapati *et al.*, 2017] propose a sequential NTM that is able to generate text conditioned on the topic using an RNN. [Panwar *et al.*, 2021] process documents as a sequence using an LSTM encoder with attention. [Zaheer *et al.*, 2017] condition the topic of one word on the topic of the previous word using RNNs. [Rezaee and Ferraro, 2020] explicitly model the topic of each individual word, combining NTMs with RNNs. [Thompson and Mimno, 2020] incorporate contextual dependencies, but their approach excludes syntactic words by using pre-processing. In contrast to our model, none of these models produce syntactic topics or automatically filter stop words or syntactic words.

[Wu and Khudanpur, 1999] present a language model that integrates local n-gram dependencies with long-range dependencies from both syntactic structure and the topic of a sentence. It is demonstrated that topic dependencies are particularly effective in predicting semantically related words that fall outside the scope of n-grams. Similarly, [Wang *et al.*, 2005] present a directed Markov random field (MRF) model that combines n-gram models, probabilistic context-free grammars (PCFGs), and probabilistic latent semantic indexing (PLSI). While [Wu and Khudanpur, 1999] and [Wang *et al.*, 2005] do not report syntactic or semantic topics, they demonstrate the efficacy of combining n-gram language modeling and topic modeling to capture both syntactic and semantic dependencies.

[Griffiths *et al.*, 2004] present a non-neural generative model that leverages both syntactic and semantic dependencies without any explicit representation of syntax or semantics beyond statistical dependence. The model employs Hidden Markov Models (HMM) to capture syntactic dependencies, and LDA to capture semantic dependencies. This work serves as the primary inspiration for our model.

[Boyd-Graber and Blei, 2008] introduce Syntactic Topic Models, which estimate topics for both syntax and semantics, but requires POS-tagged input and does not learn syntactic topics for unlabeled text. [Dieng *et al.*, 2016] present TopicRNN, an RNN-based topic model capturing global semantic meaning and local syntactic dependencies. TopicRNN does not differentiate syntactic and semantic words automatically, utilizes stop words, and lacks syntactic topic generation.

3 Method

In the section, we describe the problem setup and present proposed neural topic modeling approach integrating syntax with neural topic models and learning both syntactic and semantic topics.

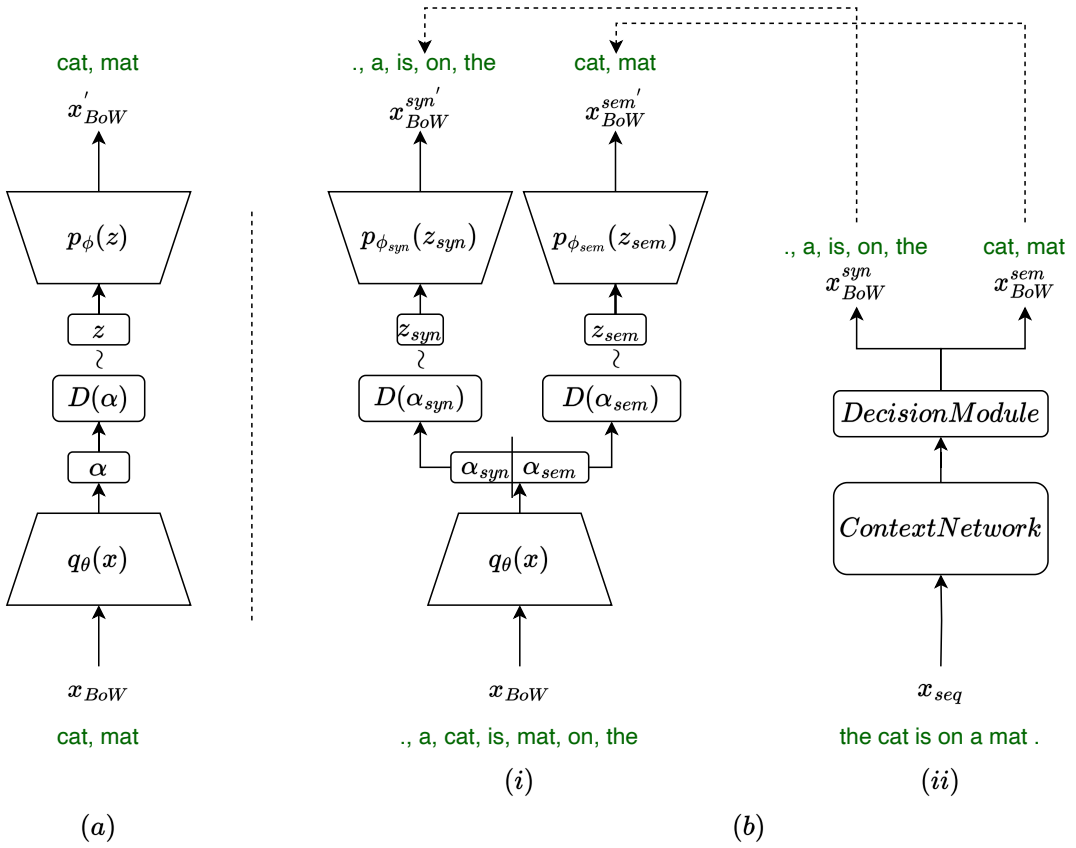

Figure 1: This figure shows (a) the standard architecture of a neural topic model and (b) our proposed model. It consists of a VAE topic model (left) and a context network (right). The topic model has two decoders, one for syntactic words and one for semantic words. The decision module decides, using the input from the context network, which words are semantic or syntactic.

3.1 Problem Setup

The standard architecture of existing VAE-based topic models is shown in Figure 1 (a). Let $\{x_i\}_N$ be the observed input documents in BoW form, where $x_i \in \mathbb{N}^V$ and V is the vocabulary size. The encoder, parameterized by θ, encodes the input into a latent vector $z \sim \text{Dirichlet}(\alpha)$, and reconstructs data using the decoder parameterized by ϕ [Burkhardt and Kramer, 2019; Srivastava and Sutton, 2017].

The goal is to learn the parameters θ and ϕ. The objective for β-VAE is defined as:

$$\begin{aligned}\mathcal{L}(\theta, \phi; x) = &-\mathbb{E}_{q_\theta(z|x)}\left[\log p_\phi(x \mid z)\right] \\ &+ \beta \cdot D_{KL}\left[q_\theta(z \mid x) \| p(z)\right]\end{aligned} \tag{1}$$

where the first term is the reconstruction error and the second term is the Kullback–Leibler (KL) [Kullback and Leibler, 1951] divergence that ensures the smoothness of the latent space. β is the hyperparameter balancing the reconstruction loss and the KL term [Higgins *et al.*, 2017]. $p(z)$ is the prior distribution, which in the case of topic models is typically the Dirichlet distribution with $\alpha \ll 1$ [Burkhardt and Kramer, 2019]. $q(z|x)$ is the approximated posterior modeled by a Dirichlet distribution, where α is parameterized by the encoder of the VAE. The latent vector $z_i \in \mathbb{R}^K$ represents the topic distribution of text document x_i, where K is the number of topics. $\phi \in \mathbb{R}^{K \times V}$ represents the topic-word distributions.

The existing VAE-based topic models are limited in including syntax in the latent space. Topics are hard to interpret if the model is forced to incorporate syntactic words, as shown in Table 1, in the bottom part where topics of a model trained without preprocessing are shown. This limitation has also been reported by [Griffiths *et al.*, 2004] and [Dieng *et al.*, 2020].

3.2 Composite-VAE

To overcome the limitations of the existing VAE-based topic models in including syntax, we propose a composite VAE architecture that can incorporate multiple subspaces in the latent space and learn the respective topics. The architecture of the proposed topic modeling framework is shown in Figure 1 (b). We will start by describing the general idea behind the left part of the figure (i), which shows the composite VAE topic model with both syntactic and semantic subspaces representing corresponding topics.

To incorporate multiple subspaces, we separate the output of the encoder into multiple parts, i.e., $\alpha = [\alpha_1; \alpha_2; \ldots; \alpha_n]$, where $[\cdot;\cdot]$ denotes concatenation. Each part of α is used as a parameter for a probabilistic distribution D of choice. $(z_1, z_2, \ldots, z_n)$ are then sampled from the respective distribu-

tions $(D(\alpha_1), D(\alpha_2), \ldots, D(\alpha_n))$ and used as an input to the corresponding decoders[2].

We now state the objective for the composite-VAE (C-VAE). The derivation of the objective and the graphical model of C-VAE is provided in Appendix A.

Theorem 3.1. *Let $\{x_i\}_N$ be the observed inputs, where $x_i \in \mathbb{N}^V$. Let q_θ and $(p_{\phi_1}, p_{\phi_2}, \ldots, p_{\phi_n})$ be the approximate posterior and likelihoods of C-VAE. The objective of C-VAE to find the parameters of posterior and likelihood such that the Evidence Lower Bound (ELBO) is maximized, is given by:*

$$\begin{aligned}\mathcal{L}_{\text{c-vae}}(\theta, \phi; x) &= \sum_{i=1}^{n} \mathcal{L}(\theta, \phi_i; x) \\ &= \sum_{i=1}^{n} -\mathbb{E}_{q_\theta(z_i|x)}\left[\log p_{\phi_i}(x_i \mid z_i)\right] \\ &\qquad + \beta D_{KL}\left[q_\theta(z_i \mid x) \,\|\, p(z_i)\right],\end{aligned}$$

where Equation 1 is substituted as shown in Appendix A.

The C-VAE architecture allows us to incorporate both syntactic and semantic topics in different subspaces. However, it cannot disentangle them in the text. For this, we employ a context network.

3.3 Context Network

The idea of the context network is that a network with a short-range context can only successfully predict syntactic words. As it lacks long-range context, it will fail to capture long-range semantic dependencies. Words that the network cannot predict are handed over to the semantic component of the composite VAE, whereas words that are predicted correctly, are handed over to the syntactic component. The context network predicts a target word w_i from a context of size c. The context can be symmetric or asymmetric, where symmetric means that words before and after the target word are included $(w_{i-(c-1)}, \ldots, w_{i-1}, w_{i+1}, \ldots, w_{i+(c+1)})$, whereas asymmetric means that only the words before the target word are used as context $(w_{i-(c-1)}, \ldots, w_{i-1})$. The neural architecture for the context network is illustrated in Appendix E. The objective for the context network is defined as the cross-entropy loss over the vocabulary of words:

$$\mathcal{L}_{CN} = -\sum_{i=1}^{V}\sum_{j=1}^{n} r(i,j) \log s(i,j), \tag{2}$$

where $r(i,j)$ is one if the target word at position j is equal to the ith word in the vocabulary and zero otherwise and $s(i,j)$ is the predicted probability by the context network for the word at position j to be the ith word in the vocabulary.

3.4 Decision Module

The decision module assigns each word to its respective syntactic or semantic class based on the predictions generated by the context network. We put forward two techniques for this:

Top-M: A word is assigned the syntax class if it ranks within the top-M predicted words with the highest probabilities. Here, M is a hyperparameter.

Probability Threshold: A word is labeled as a syntax word if its predicted probability surpasses a defined threshold t.

We investigate these two techniques in Appendix F, and based on the results, we select the Top-M condition with M set to 5 as the configuration for our final model. It is important to note that a particular word might be classified as a syntactic word in one context and a semantic word in another. This addresses scenarios where, for instance, a single word might serve as a noun or an adjective, or even convey entirely different meanings.

3.5 SyConNTM: Integrating Syntax With NTMs

To integrate syntax with the existing neural topic modeling architecture, we utilize the proposed composite VAE and incorporate two subspaces for semantic and syntactic information respectively. From the context network and decision module, we obtain $x^{\text{syn}}_{\text{BoW}}$ and $x^{\text{sem}}_{\text{BoW}}$ as two BoW vectors where one document x is the sum of the two vectors $x = x^{\text{syn}}_{\text{BoW}} + x^{\text{sem}}_{\text{BoW}}$. Accordingly, we separate the output of the encoder into two parameter vectors α_{syn} and α_{sem}, representing syntax and semantics, respectively, i.e., $\alpha = [\alpha_{\text{syn}}; \alpha_{\text{sem}}]$, where $[\cdot;\cdot]$ denotes concatenation.

Following Theorem 3.1 the objective is defined as:

$$\mathcal{L}_{\text{sc-ntm}}(\theta, \phi; x) = \mathcal{L}_s(\theta, \phi_{\text{syn}}; x) + \mathcal{L}_c(\theta, \phi_{\text{sem}}; x), \tag{3}$$

where the two terms are the syntactic and semantic loss respectively. We also define a hyperparameter λ for balancing the KL terms of $\mathcal{L}_s$ and $\mathcal{L}_c$.

Formally, the SyConNTM loss is defined as:

$$\begin{aligned}\mathcal{L}_{\text{sc-ntm}}(\theta, \phi; x) = &-\mathbb{E}_{q(z_{\text{syn}}|x)}\left[\log p(x_{\text{syn}} \mid z_{\text{syn}})\right] \\ &- \mathbb{E}_{q(z_{\text{sem}}|x)}\left[\log p(x_{\text{sem}} \mid z_{\text{sem}})\right] \\ &+\lambda\cdot\beta\cdot\left(D_{KL}\left[q(z_{\text{syn}} \mid x) \,\|\, p(z_{\text{syn}})\right]\right) \\ &+(1-\lambda)\cdot\beta\cdot\left(D_{KL}\left[q(z_{\text{sem}} \mid x) \,\|\, p(z_{\text{sem}})\right]\right)\end{aligned} \tag{4}$$

where the first two terms are the reconstruction error followed by KL terms for syntax and semantics respectively.

In the training process of SyConNTM, for each batch $\mathcal{B}$ in dataset D, we construct both BoW (x_{BoW}) and sequential (x_{seq}) representations. The latter feeds into the *ContextNetwork* to generate word probabilities s, updated using Equation 2, employed by the *DecisionModule* to differentiate syntax and content words. This distinction facilitates the construction of BoW vectors, x^{syn}_{BoW} and x^{sem}_{BoW}. The BoW representation, x_{BoW}, is then input to the SC-NTM model, which updates its parameters per Equation 4. The context network can also be pretrained resulting in a faster convergence of SyConNTM at the cost of the pretraining overhead, but in our investigation it did not improve results.

4 Experiments

We first briefly discuss the used datasets, comparison models, and evaluation measures. Then, we evaluate the generated topics of SyConNTM in Section 4.4 and compare them to other models in Section 4.5. To further validate the semantic topics, we also discuss the application of SyConNTM as an automatic feature selector in Section 4.6.

[2]Note, that if the Dirichlet distribution is chosen, its renormalization property ensures that if z is split and z_i is renormalized, z_i is still Dirichlet distributed $\forall i$.

4.1 Datasets and Preprocessing

In our experiments, we utilize seven well-known datasets. The 20 Newsgroups (20NG) dataset features around 18K newsgroup posts across 20 classes [Lang, 1995]. The Amazon reviews (AR) dataset includes roughly 35M reviews spanning 18 years [McAuley and Leskovec, 2013]. The AG News (AGN) corpus contains over a million news articles from 2,000+ sources [Zhang *et al.*, 2015]. The GovReport Summaries (GR) dataset, as introduced by [Huang *et al.*, 2021], provides summaries of about 20K government reports. The IMDB reviews (IR) dataset incorporates 50K movie reviews [Maas *et al.*, 2011]. The Rotten Tomatoes reviews (RT) dataset presents 5,331 positive and 5,331 negative processed sentences from movie reviews [Pang and Lee, 2005]. Finally, the Yelp reviews (YR) dataset is composed of reviews from the 2015 Yelp Dataset Challenge [Zhang *et al.*, 2015]. All datasets, except for GR, are labeled.

We utilize SpaCy for tokenization [Honnibal and Montani, 2017], substituting special tokens (e.g., email addresses, numeric values, alphanumeric characters) with generic labels, and marking out-of-vocabulary tokens as unknown. For SyConNTM, no additional preprocessing is done; however, for other topic models, we eliminate stop words, punctuation, and words found in fewer than 50 documents or in over 75% of documents. Further details on datasets and preprocessing are available in Appendix D.

4.2 Models

Baselines. Following [Hoyle *et al.*, 2021]'s analysis, we adopt three baseline models: Gibbs-LDA (G-LDA) [McCallum, 2002], Dirichlet-VAE (D-VAE) [Burkhardt and Kramer, 2019], and Embedded Topic Model (ETM) [Dieng *et al.*, 2020]. G-LDA, a statistical model, employs Gibbs sampling, while D-VAE and ETM, both state-of-the-art NTMs, incorporate a Dirichlet prior and external word embeddings respectively. These baselines are popular and are validated through human evaluations [Hoyle *et al.*, 2021].

SyConNTM. The proposed framework can be easily incorporated with the existing NTMs. We propose two SyConNTM variants - one with D-VAE and another with ETM. To maintain consistency, SyConNTM adopts a Dirichlet prior, mirroring D-VAE and ETM's architecture. For further details, refer to Appendix E.

4.3 Evaluation Measures

We gauge model performance using standard evaluation measures: Topic Coherence (TC), Topic Diversity (TD), and Topic Quality (TQ). TC, calculated via the C_V coherence score [Röder *et al.*, 2015], examines top word co-occurrence within a topic, using the training dataset as the reference corpus. TD measures the fraction of unique words in all topics [Dieng *et al.*, 2020], where a score near zero indicates topic redundancy and a score close to one suggests diverse topics. TQ, defined as the TC and TD product [Dieng *et al.*, 2020], reflects the interpretability and diversity of topics. Additionally, we introduced the Semantic Purity (SP) metric, which assesses purity of semantic topics by quantifying stop words. SP is computed as: $\text{SP} = 1 - \frac{\sum_{w \in \text{topics}} \mathbb{1}[w \in \text{stop words}]}{|\text{topics}|}$ [Loper and Bird, 2002], where $\mathbb{1}$ is the indicator function which is one if word w is in the stop word list and zero otherwise. A SP value near one implies minimal stop words within our semantic topics, denoting disentangled.

very slow service , our **food** came one by one within [NUM] to [NUM] **minutes** of each other .

rude **servers** and took us a **long** time to get our **bills** with lots of **mistakes** . overall very bad experience .

Figure 2: Designation of words in Yelp reviews as syntactic or semantic by SyConNTM. The green words are designated as semantic by the model, and the rest are syntactic.

4.4 Evaluating Syntactic and Semantic Topics

Setup: In this experiment, we compare the output of SyConNTM to the existing methods. For this experiment, all the selected baselines are trained for ten topics both with and without pre-processing. SyConNTM is trained for ten semantic topics and ten syntactic topics without any pre-processing.

Results: Topics from the Yelp reviews dataset are shown in Table 1 where we compare results from SyConNTM against D-VAE. The full set of topics from Yelp and 20 Newsgroups for all models is given in Appendix G, the rest can be found in the supplement. SyConNTM cleanly separates syntactic and semantic words in contrast to the baseline methods. Using preprocessing, only predefined stop words are removed such that not all relevant words (et, un, le...) are captured, whereas without preprocessing, all topics are dominated by stop words. To evaluate whether our model detects the stop words from an existing list of stop words, we report the semantic purity in Table 3, confirming the minimal presence of stop words.

We also show how the decision module assigns words in Figure 2. The highlighted words are semantic while the rest are syntactic. Interestingly, we observe, that the syntactic words produce a template for writing reviews in which the semantic words can be inserted. For example, replacing the semantic words which the model identifies in the first sentence with semantic words appropriate to some other topic, we could write: "very slow service, our *tickets* came one by one within [NUM] to [NUM] *hours* of each other".

We conclude that SyConNTM learns both syntactic and semantic topics without any pre-processing. The syntactic topics from SyConNTM provide additional information on the syntactic structure of a corpus. They are not limited to stop words but also contain stylistic elements (good, loved, great etc.) and topics frequent in the corpus. Our model also distinguishes other languages that were missed in preprocessing. The semantic topics of SyConNTM cover the content better as compared to the baselines since it has separated corpus-specific content in the syntactic topics.

		Topic 1	Topic 2	Topic 3	Topic 4	Topic 5	Topic 6	Topic 7	Topic 8	Topic 9	Topic 10
Ours	syntactic	i	good	la	ich	.	years	future	is	he	would
		me	loved	de	der	,	month	moments	has	she	could
		she	great	et	hier	!	days	chaser	always	they	did
		my	amazing	le	da	_	minutes	aspiring	than	you	when
		in	very	pas	nach	"	hour	acting	want	this	am
	semantic	theater	tacos	hotel	company	pancakes	salad	meat	game	dessert	cars
		band	bbq	rooms	office	waffle	sandwich	bbq	movie	chocolate	office
		seats	mexican	casino	wash	butter	dressing	red	games	cake	tire
		stage	burrito	bathroom	salon	breakfast	grilled	side	sports	gelato	repair
		store	wings	pool	car	cafe	bread	cut	watch	cupcakes	tires
Baseline	preprocessed	store	burger	drink	hotel	et	tacos	restaurant	love	und	buffet
		car	good	bar	rooms	un	mexican	table	store	ist	buffets
		work	fries	steak	hotels	le	salsa	food	amazing	der	rib
		shop	burgers	great	floor	de	chips	server	favorite	die	prime
		business	hot	table	vegas	pour	taco	sushi	home	das	steak
	raw	came	if	to	it	les	-	&	!	-	,
		were	get	the	the	pour	)	!	?	.	bread
		table	people	for	of	la	(	got	card	ribs	*
		food	into	if	,	des	fish	nails	said	orange	chicken
		very	room	she	for	par	happy	salon	charge	cabbage	sandwich

Table 1: The proposed model (top) produces syntactic and semantic topics, whereas the baseline neural topic model (bottom) does not. Shown are five of ten selected topics on the Yelp dataset. The current baselines without preprocessing produce uninterpretable and ineffective topics, while preprocessing results in the loss of the corpus' rich syntactic representation.

	SyConNTM						G-LDA			D-VAE			ETM		
	D-VAE			ETM											
	TC	TD	TQ	TC	TD	TQ	TC	TD	TQ	TC	TD	TQ	TC	TD	TQ
50 Topics															
20NG	0.53	**0.87**	**0.46**	0.41	0.82	0.33	0.48	0.56	0.27	**0.63**	0.61	0.38	0.36	0.47	0.16
AR	**0.58**	**0.79**	**0.46**	0.34	0.72	0.24	0.40	0.38	0.15	0.55	0.68	0.37	0.46	0.61	0.27
AGN	0.50	**0.88**	**0.44**	0.30	0.84	0.25	0.44	0.18	0.08	**0.56**	0.68	0.38	0.53	0.47	0.25
GR	0.48	**0.67**	0.32	0.42	0.68	0.29	0.54	0.59	0.32	**0.57**	0.60	**0.35**	0.35	0.34	0.12
IR	0.43	**0.78**	**0.34**	**0.48**	0.67	0.33	0.41	0.52	0.21	0.41	0.43	0.18	0.30	0.49	0.15
RT	0.43	**0.88**	0.38	**0.55**	0.78	**0.43**	0.30	0.04	0.01	0.31	0.04	0.01	0.31	0.04	0.01
YR	**0.58**	**0.72**	**0.42**	0.50	0.68	0.34	0.48	0.60	0.29	0.45	0.42	0.19	0.35	0.12	0.04
200 Topics															
20NG	**0.45**	**0.25**	**0.11**	0.39	0.21	0.08	0.43	0.20	0.09	0.39	0.17	0.07	0.31	0.18	0.06
AR	**0.42**	0.38	0.16	0.41	**0.39**	**0.16**	0.36	0.10	0.03	0.34	0.09	0.03	0.31	0.04	0.01
AGN	**0.41**	0.03	**0.02**	0.32	0.02	0.01	0.26	**0.04**	0.01	0.26	0.04	0.01	0.24	0.04	0.01
GR	**0.52**	0.34	0.18	0.35	0.40	0.14	0.50	**0.41**	**0.20**	0.49	0.25	0.12	0.30	0.13	0.04
IR	0.35	**0.38**	**0.13**	**0.39**	0.32	0.12	0.37	0.36	0.13	0.37	0.19	0.07	0.29	0.04	0.01
RT	**0.49**	0.04	0.02	0.47	**0.04**	**0.02**	0.30	0.01	0.00	0.30	0.01	0.00	0.31	0.01	0.00
YR	0.47	**0.28**	0.13	**0.51**	0.27	**0.14**	0.40	0.22	0.09	0.38	0.17	0.06	0.30	0.03	0.01

Table 2: In general, SyConNTM performs better than the baselines when comparing semantic topics. This table shows topic quality (TQ), topic coherence (TC), and topic diversity (TD) for our SyConNTM and baselines on 50 topics (top) and 200 topics (bottom). The best value for each measure is shown in bold.

4.5 Benchmarking SyConNTM

Setup: We benchmark SyConNTM against baselines using topic coherence, diversity, and quality across datasets. Adhering to standard procedures, we train all models on 50 and 200 topics five times, limiting SyConNTM's syntactic topics to 10. For fairness, we compare topic quality of semantic topics from SyConNTM with topics produced by baselines *after* pre-processing.

Results: Table 2 highlights the best result for each measure in bold. In general, the proposed SyConNTM produces semantic topics with higher topic coherence, diversity, and quality than the topics produced by the baselines after pre-processing.

Datasets	50 Topics	200 Topics
20ng	0.90	0.90
Amazon Reviews	0.97	0.97
AG News	0.88	0.92
Gov_Reports	0.99	0.99
IMDB Reviews	0.95	0.96
Rotten Tomatoes	1.00	1.00
Yelp Reviews	0.96	0.97

Table 3: Semantic topics found by our model contain almost no stop words as shown by the high semantic purity.

Datasets	prepro	SWR
20ng	75.2	**89.7**
Amazon Reviews	82.1	**85.4**
AG News	72.1	**83.8**
IMDB Reviews	78.4	**81.6**
Rotten Tomatoes	59.2	**63.1**
Yelp Reviews	51.3	**68.5**

Table 4: SyConNTM is a superior feature extractor for text classification as compared to standard preprocessing. Shown is the *text classification accuracy* using Naive Bayes with two different forms of syntax word removal: 1. standard pre-processing (PREPRO) and 2. syntactic word removal (SWR) by SyConNTM.

For 50 topics D-VAE has a higher coherence on three of the datasets, for 200 topics, G-LDA performs better on only one dataset. This shows that our model performs especially well with higher numbers of topics. The versatility of the model is further demonstrated by its ability to generate competitive results when combined with both D-VAE and ETM. When using SyConNTM in conjunction with ETM, it produces higher quality outcomes for two datasets with 50 topics, as well as for four datasets with 200 topics. The improvement of the ETM variant for 200 topics can be attributed to the use of pre-trained embeddings.

4.6 Application as an Automatic Feature Detector

Another way of investigating whether our semantic topics actually capture the content of our corpus, is to use them as features for classification. Since the decision module of the proposed framework designates each word in a document as either syntactic or semantic, it can perform feature removal (based on syntactic tags) as compared to using the standard stop words and punctuation list.

Setup: We use a standard Gaussian Naive Bayes classifier. Except for the GovReport Summaries, all our datasets are labeled. For our baseline results, we use the pre-processed versions of the datasets. For automatic feature detection, we train SyConNTM and remove words designated as syntax by the decision module.

Results: The accuracy of both methods is shown in Table 4. In all cases, accuracy after removing syntactic words based on SyConNTM outperforms standard pre-processing. We suspect this is because the standard pre-processing does not account for the context in the data. This shows that our semantic topics indeed capture the content as signified by the dataset labels.

5 Conclusion

To conclude, we have presented a neural topic model that learns semantic as well as syntactic topics by integrating a language model as a context network. Our experiments show a consistent improvement in topic quality of the semantic topics over existing state-of-the-art semantic topic models. The syntactic topics decrease the reliance of our neural topic model on inconsistent preprocessing pipelines. Additionally, the syntactic topics are an additional source of information about a corpus and increase the interpretability of the learned topic model. Lastly, we emphasize that our framework is generalizable to any VAE-based neural topic model.

In the future, we plan to develop models that do both, learn topics and generate text based on these topics. We will also investigate the optimal number of syntactic and semantic topics and their relationship to each other. Other research directions are to include hierarchical levels in the topic model and more fine-grained syntactical categories in the language model part.

6 Limitations

While the proposed work demonstrates notable strengths, it is important to acknowledge several limitations. Firstly, the proposed framework primarily focuses on learning topics from syntactic words. Expanding the framework to include learning syntactic classes from the syntactic words could enhance the utility of the model. Secondly, this research primarily emphasizes the integration of syntax with neural topic models, rather than extensively exploring syntactic topics or determining the optimal number of syntactic topics to be learned within the model. Additionally, the reliance on word-level tokenizers, imposed by the topic model, imposes constraints on the proposed framework. These limitations present opportunities for future developments and research directions.

Acknowledgements

The authors acknowledge support by the Carl-Zeiss Foundation, the BMBF award 01IS20048, and the DFG awards BU 4042/2-1 and BU 4042/1-1.

References

[Bhatt *et al.*, 2020] Gantavya Bhatt, Hritik Bansal, Rishubh Singh, and Sumeet Agarwal. How much complexity does an RNN architecture need to learn syntax-sensitive dependencies? In *Proceedings of the 58th Annual Meeting of the ACL: Student Research Workshop*, pages 244–254, Online, July 2020.

[Blei *et al.*, 2003] David M Blei, Andrew Y Ng, and Michael I Jordan. Latent dirichlet allocation. *Journal of machine Learning research*, 3(Jan):993–1022, 2003.

[Boyd-Graber and Blei, 2008] Jordan Boyd-Graber and David Blei. Syntactic topic models. *Advances in neural information processing systems*, 21, 2008.

[Broder *et al.*, 1997] Andrei Z Broder, Steven C Glassman, Mark S Manasse, and Geoffrey Zweig. Syntactic clustering of the web. *Computer networks and ISDN systems*, 29(8-13):1157–1166, 1997.

[Brown, 1973] Roger Brown. A first language (pp. 54). *Cambridge, MA: Harvard University Press*, 1973.

[Burkhardt and Kramer, 2019] Sophie Burkhardt and Stefan Kramer. Decoupling sparsity and smoothness in the dirichlet variational autoencoder topic model. *J. Mach. Learn. Res.*, 20(131):1–27, 2019.

[Cavnar *et al.*, 1994] William B Cavnar, John M Trenkle, et al. N-gram-based text categorization. In *Proceedings of SDAIR-94, 3rd annual symposium on document analysis and information retrieval*, volume 161175. Las Vegas, NV, 1994.

[Chelba *et al.*, 2017] Ciprian Chelba, Mohammad Norouzi, and Samy Bengio. N-gram language modeling using recurrent neural network estimation. *arXiv preprint arXiv:1703.10724*, 2017.

[Currey and Heafield, 2019] Anna Currey and Kenneth Heafield. Incorporating source syntax into transformer-based neural machine translation. In *Proceedings of the Fourth Conference on Machine Translation (Volume 1: Research Papers)*, pages 24–33, 2019.

[Dieng *et al.*, 2016] Adji B Dieng, Chong Wang, Jianfeng Gao, and John Paisley. Topicrnn: A recurrent neural network with long-range semantic dependency. *arXiv preprint arXiv:1611.01702*, 2016.

[Dieng *et al.*, 2020] Adji B Dieng, Francisco JR Ruiz, and David M Blei. Topic modeling in embedding spaces. *Transactions of the ACL*, 8:439–453, 2020.

[Duan *et al.*, 2019] Sufeng Duan, Hai Zhao, Junru Zhou, and Rui Wang. Syntax-aware transformer encoder for neural machine translation. In *2019 International Conference on Asian Language Processing (IALP)*, pages 396–401. IEEE, 2019.

[Figurnov *et al.*, 2018] Mikhail Figurnov, Shakir Mohamed, and Andriy Mnih. Implicit reparameterization gradients. *Advances in neural information processing systems*, 31, 2018.

[Griffiths *et al.*, 2004] Thomas Griffiths, Mark Steyvers, David Blei, and Joshua Tenenbaum. Integrating topics and syntax. *Advances in neural information processing systems*, 17, 2004.

[Grimmer and Stewart, 2013] Justin Grimmer and Brandon M Stewart. Text as data: The promise and pitfalls of automatic content analysis methods for political texts. *Political analysis*, 21(3):267–297, 2013.

[Gui *et al.*, 2019] Lin Gui, Jia Leng, Gabriele Pergola, Yu Zhou, Ruifeng Xu, and Yulan He. Neural topic model with reinforcement learning. In *Proceedings of the 2019 Conference on EMNLP and the 9th IJCNLP*, pages 3478–3483, 2019.

[Gupta *et al.*, 2019] Pankaj Gupta, Yatin Chaudhary, Florian Buettner, and Hinrich Schütze. Document informed neural autoregressive topic models with distributional prior. *Proceedings of the AAAI Conference on Artificial Intelligence*, 33(01):6505–6512, Jul. 2019.

[Higgins *et al.*, 2017] Irina Higgins, Loïc Matthey, Arka Pal, Christopher P. Burgess, Xavier Glorot, Matthew M. Botvinick, Shakir Mohamed, and Alexander Lerchner. beta-vae: Learning basic visual concepts with a constrained variational framework. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*, 2017.

[Honnibal and Montani, 2017] Matthew Honnibal and Ines Montani. spaCy: Natural language understanding with Bloom embeddings, convolutional neural networks and incremental parsing. To appear, 2017.

[Hoyle *et al.*, 2021] Alexander Hoyle, Pranav Goel, Andrew Hian-Cheong, Denis Peskov, Jordan Boyd-Graber, and Philip Resnik. Is automated topic model evaluation broken? the incoherence of coherence. *Advances in Neural Information Processing Systems*, 34:2018–2033, 2021.

[Huang *et al.*, 2021] Luyang Huang, Shuyang Cao, Nikolaus Parulian, Heng Ji, and Lu Wang. Efficient attentions for long document summarization. In *Proceedings of the 2021 Conference of the North American Chapter of the ACL: Human Language Technologies*, pages 1419–1436, Online, June 2021. ACL.

[Joo *et al.*, 2020] Weonyoung Joo, Wonsung Lee, Sungrae Park, and Il-Chul Moon. Dirichlet variational autoencoder. *Pattern Recognition*, 107:107514, 2020.

[Kullback and Leibler, 1951] Solomon Kullback and Richard A Leibler. On information and sufficiency. *The annals of mathematical statistics*, 22(1):79–86, 1951.

[Lang, 1995] Ken Lang. Newsweeder: Learning to filter netnews. In *Machine Learning Proceedings 1995*, pages 331–339. Morgan Kaufmann, San Francisco (CA), 1995.

[Larochelle and Lauly, 2012] Hugo Larochelle and Stanislas Lauly. A neural autoregressive topic model. In *Advances in Neural Information Processing Systems*, volume 25. Curran Associates, Inc., 2012.

[Li *et al.*, 2022] Yewen Li, Chaojie Wang, Zhibin Duan, Dongsheng Wang, Bo Chen, Bo An, and Mingyuan Zhou. Alleviating "posterior collapse" in deep topic models via policy gradient. In *Advances in Neural Information Processing Systems*, 2022.

[Liu *et al.*, 2016] Lin Liu, Lin Tang, Wen Dong, Shaowen Yao, and Wei Zhou. An overview of topic modeling and its current applications in bioinformatics. *SpringerPlus*, 5(1):1–22, 2016.

[Loper and Bird, 2002] Edward Loper and Steven Bird. Nltk: The natural language toolkit. *arXiv preprint cs/0205028*, 2002.

[Maas *et al.*, 2011] Andrew L. Maas, Raymond E. Daly, Peter T. Pham, Dan Huang, Andrew Y. Ng, and Christopher Potts. Learning word vectors for sentiment analysis. In *Proceedings of the 49th Annual Meeting of the ACL: Human*

Language Technologies, pages 142–150, Portland, Oregon, USA, June 2011. ACL.

[McAuley and Leskovec, 2013] Julian McAuley and Jure Leskovec. Hidden factors and hidden topics: understanding rating dimensions with review text. In *Proceedings of the 7th ACM conference on Recommender systems*, pages 165–172, 2013.

[McCallum, 2002] Andrew Kachites McCallum. Mallet: A machine learning for languagetoolkit. *http://mallet. cs. umass. edu*, 2002.

[Meeks and Weingart, 2012] Elijah Meeks and Scott B Weingart. The digital humanities contribution to topic modeling. *Journal of Digital Humanities*, 2(1):1–6, 2012.

[Miao *et al.*, 2016] Yishu Miao, Lei Yu, and Phil Blunsom. Neural variational inference for text processing. In *ICML*, pages 1727–1736. PMLR, 2016.

[Nallapati *et al.*, 2017] Ramesh Nallapati, Igor Melnyk, Abhishek Kumar, and Bowen Zhou. Sengen: Sentence generating neural variational topic model, 2017.

[Neville *et al.*, 1992] Helen J Neville, Debra L Mills, and Donald S Lawson. Fractionating language: Different neural subsystems with different sensitive periods. *Cerebral Cortex*, 2(3):244–258, 1992.

[Nguyen and Luu, 2021] Thong Nguyen and Anh Tuan Luu. Contrastive learning for neural topic model. *Advances in Neural Information Processing Systems*, 34:11974–11986, 2021.

[Pang and Lee, 2005] Bo Pang and Lillian Lee. Seeing stars: Exploiting class relationships for sentiment categorization with respect to rating scales. In *Proceedings of the ACL*, 2005.

[Panwar *et al.*, 2021] Madhur Panwar, Shashank Shailabh, Milan Aggarwal, and Balaji Krishnamurthy. TAN-NTM: Topic attention networks for neural topic modeling. In *Proceedings of the 59th Annual Meeting of the ACL and the 11th IJCNLP(Volume 1: Long Papers)*, pages 3865–3880, Online, August 2021.

[Redington *et al.*, 1998] Martin Redington, Nick Chater, and Steven Finch. Distributional information: A powerful cue for acquiring syntactic categories. *Cognitive science*, 22(4):425–469, 1998.

[Rezaee and Ferraro, 2020] Mehdi Rezaee and Francis Ferraro. A discrete variational recurrent topic model without the reparametrization trick. In *Proceedings of the 34th International Conference on Neural Information Processing Systems*, NIPS'20, Red Hook, NY, USA, 2020. Curran Associates Inc.

[Röder *et al.*, 2015] Michael Röder, Andreas Both, and Alexander Hinneburg. Exploring the space of topic coherence measures. In *Proceedings of the eighth ACM international conference on Web search and data mining*, pages 399–408, 2015.

[Srivastava and Sutton, 2017] Akash Srivastava and Charles Sutton. Autoencoding variational inference for topic models. In *International Conference on Learning Representations*, 2017.

[Thompson and Mimno, 2020] Laure Thompson and David Mimno. Topic modeling with contextualized word representation clusters. *arXiv preprint arXiv:2010.12626*, 2020.

[Wang *et al.*, 2005] Shaojun Wang, Shaomin Wang, Russell Greiner, Dale Schuurmans, and Li Cheng. Exploiting syntactic, semantic and lexical regularities in language modeling via directed markov random fields. In *Proceedings of the 22nd ICML*, pages 948–955, 2005.

[Wang *et al.*, 2020] Rui Wang, Xuemeng Hu, Deyu Zhou, Yulan He, Yuxuan Xiong, Chenchen Ye, and Haiyang Xu. Neural topic modeling with bidirectional adversarial training. In *Proceedings of the 58th Annual Meeting of the ACL*, pages 340–350, Online, July 2020.

[Wu and Khudanpur, 1999] Jun Wu and Sanjeev Khudanpur. Combining nonlocal, syntactic and n-gram dependencies in language modeling. In *Sixth European Conference on Speech Communication and Technology*, 1999.

[Yingzhen and Mandt, 2018] Li Yingzhen and Stephan Mandt. Disentangled sequential autoencoder. In *Proceedings of the 35th ICML*, volume 80 of *PMLR*, pages 5670–5679, 10–15 Jul 2018.

[Zaheer *et al.*, 2017] Manzil Zaheer, Amr Ahmed, and Alexander J. Smola. Latent LSTM allocation: Joint clustering and non-linear dynamic modeling of sequence data. In *Proceedings of the 34th ICML*, volume 70 of *PMLR*, pages 3967–3976, 06–11 Aug 2017.

[Zhang *et al.*, 2015] Xiang Zhang, Junbo Jake Zhao, and Yann LeCun. Character-level convolutional networks for text classification. In *Advances in Neural Information Processing Systems 28: Annual Conference on Neural Information Processing Systems 2015, December 7-12, 2015, Montreal, Quebec, Canada*, pages 649–657, 2015.

[Zhang *et al.*, 2018] Hao Zhang, Bo Chen, Dandan Guo, and Mingyuan Zhou. WHAI: Weibull hybrid autoencoding inference for deep topic modeling. In *International Conference on Learning Representations*, 2018.

[Zhao *et al.*, 2021] He Zhao, Dinh Phung, Viet Huynh, Yuan Jin, Lan Du, and Wray L. Buntine. Topic modelling meets deep neural networks: A survey. *CoRR*, abs/2103.00498, 2021.