# Bivariate Beta-LSTM

**Kyungwoo Song, JoonHo Jang, Seung jae Shin, Il-Chul Moon**
Korea Advanced Institute of Science and Technology (KAIST), Korea
{gtshs2, adkto8093, tmdwo0910, icmoon}@kaist.ac.kr

## Abstract

Long Short-Term Memory (LSTM) infers the long term dependency through a cell state maintained by the input and the forget gate structures, which models a gate output as a value in [0,1] through a sigmoid function. However, due to the graduality of the sigmoid function, the sigmoid gate is not flexible in representing multi-modality or skewness. Besides, the previous models lack modeling on the correlation between the gates, which would be a new method to adopt inductive bias for a relationship between previous and current input. This paper proposes a new gate structure with the bivariate Beta distribution. The proposed gate structure enables probabilistic modeling on the gates within the LSTM cell so that the modelers can customize the cell state flow with priors and distributions. Moreover, we theoretically show the higher upper bound of the gradient compared to the sigmoid function, and we empirically observed that the bivariate Beta distribution gate structure provides higher gradient values in training. We demonstrate the effectiveness of the bivariate Beta gate structure on the sentence classification, image classification, polyphonic music modeling, and image caption generation.

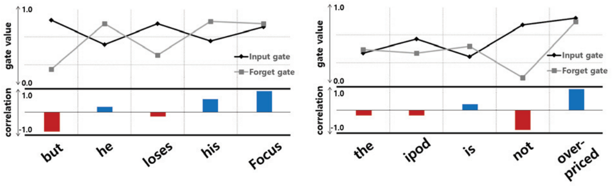
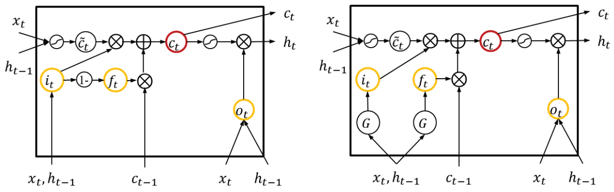


Figure 1: An illustrative example of the input gate, the forget gate, and their correlation for part of a given sentence in sentiment classification datasets. Blue and Red bars denote the positive and negative correlations, respectively. (Left) The new context starts from the word ”but”, so the negative correlation imposes the large input gate value and the small forget gate value to focus on the current context. (Right) To infer the semantic meaning between the word ”not” and ”over-priced”, the positive correlation occurs at ”over-priced”, so the positive correlation makes the input gate and the forget gate have large values. The high forget gate value denotes the importance of the previous context in our model.

## Introduction

One of the most commonly used Recurrent Neural Network (RNN) variants is *Long Short-Term Memory* (LSTM) (Hochreiter and Schmidhuber 1997), which introduces additional gate structures for controlling cell states. LSTM controls the information flow from a sequence with an input, a forget, and an output gate. The input and the forget gates decide the ratio of mixture between the current and the previous information at each time step. The sigmoid function is defined to be bounded and monotonically increasing, so the sigmoid has been a popular choice for such gate mechanisms.

In spite of the prevalence of sigmoid functions, there has been a question on the utility and the efficiency of the sigmoid function used for the gates in LSTM. For instance, the confined gate value range, which is narrower than the 0-1 bound, means that the majority of gate values may fall into the narrower range, and this makes that the gate values lose potential discrimination power (Li et al. 2018). Some tried to use additional hyper-parameters to sharpen the sigmoid function, i.e., the sigmoid function with temperature parameter (Li et al. 2018), but these would be limited to the support for the sigmoid function without fundamental innovations. From this perspective, there are few works to probabilistically model the flexibility of the gate structure, i.e., $G^2$-LSTM (Li et al. 2018) with the Bernoulli distribution, but the current probabilistic model missed the graduality of the gate value change. Moreover, it has been known that the gates could be correlated (Greff et al. 2017), and the performance can be improved by exploiting this covariance structure. One common conjecture is the correlation between the input and the forget gate values in LSTM. However, the structure of LSTM does not explicitly model such correlation, so its enforcing structure was handled at the technical implementation level. For instance, CIFG-LSTM (Greff et al. 2017) enforces the negative correlation between the input and the forget gate values. CIFG-LSTM shows competitive performance with reduced parameters because of the correlation modeling. However, CIFG-LSTM enforces the strict negative correlation, -1 only; and it needs to be generalized



(a) CIFG-LSTM

(b) $G^2$-LSTM

Figure 2: The cell structure of CIFG-LSTM and $G^2$-LSTM

by a model. We improve the correlation structure adaptable to datasets flexibly.

This correlation modeling is frequently included in the data domains, such as texts and images. For text datasets, there are the syntactic and the semantic relationships between words in a sentence (Harabagiu 2004), so the information flow to the cell structure should reflect the semantic relationship. Similarly, for image datasets, there is a relationship between pixels in a short-range, as well as pixels in a long-range within a single image (Kampffmeyer et al. 2019). The relation modeling is one of the effective inductive biases for deep learning models (Battaglia et al. 2018), which can handle the property of datasets.

In a general setting, let us assume that we prefer a large input value and a large forget gate value when both of the current and the previous information are important. Then, a positive correlation can be an effective inductive bias for modeling the idiom such as *rely on*, to control the gate value, efficiently. For the opposite case, A negative correlation can be a good inductive bias to model the sentence ”...but he loses his focus”. To reflect the change of context, the input gate and the forget gate should have a large and small value, respectively, at ”but” as shown in Figure 1.

We propose a bivariate Beta LSTM (bBeta-LSTM), which improves the sigmoid function in the input gate and the forget by substituting the sigmoid function with a bivariate Beta distribution. bBeta-LSTM has three advantages over the LSTM. First, the Beta distribution can represent values of [0,1] flexibly, since a Beta distribution is a generalized distribution of the uniform distribution, the power function, and the Bernoulli distribution with 0.5 probability. The Beta distribution can represent either symmetric or skewed shape by adjusting two shape parameters. Second, the bivariate Beta distribution can represent the covariance structure of the input and the forget gates because a bivariate Beta distribution shares the Gamma random variables, which make the correlation between two sampled values. We utilized the property of the bivariate Beta distribution for modeling the input and the forget gates in bBeta-LSTM. The bivariate Beta distribution could be further elaborated by expanding the probabilistic model, i.e., adding a common prior to the input gate and the forget gate distributions. Third, the bivariate Beta distribution can alleviate the gradient vanishing problem of LSTM. Under a certain condition, we verify that the derivative of gates in bBeta-LSTM is greater than those of LSTM, experimentally and theoretically.

## Preliminary: Stochastic Gate in RNN

Since RNN is a deterministic model, it is difficult to prevent overfitting, and it is infeasible to generate diverse outputs. Therefore, multiple methods were explored to model the stochasticity in sequence learning. First, dropout methods for RNN (Gal and Ghahramani 2016; Park et al. 2019) demonstrated that stochastic masking could improve its generalization. Second, latent variables were a good combination with the RNN structure, such as Variational RNN (VRNN) (Chung et al. 2015) and Variable Hierarchical Recurrent Encoder Decoder (VHRED) (Serban et al. 2017). Third, the gate mechanisms, which are extensively used in RNN variants due to the vanishing gradient, can be substituted with probabilistic models.

When we investigate further on the gate mechanism, there have been efforts in reducing the number of gates (Lei et al. 2018), enabling a gate structure to be a complex number (Wolter and Yao 2018), correlating gate structures (Greff et al. 2017). For instance, Gumbel Gate LSTM ($G^2$-LSTM) (Li et al. 2018) replaces the sigmoid function of the input and the forget gates with Bernoulli distributions. The Bernoulli gates in $G^2$-LSTM turns the continuous gate values to be the binary value of 0 or 1. This work can be expanded to incorporate a continuous spectrum, a multi-modality, and stochasticity, at the same time. Furthermore, $G^2$-LSTM uses a Gumbel-Softmax, which remains in the realm of sigmoid gates, so the limitations discussed in the Introduction are still applicable. Figure 2b and below equations enumerate the information flow in $G^2$-LSTM, and $G$ is the Gumbel-softmax function with a temperature parameter of $\tau$.

$$i_t = G(W_{xi}x_t + W_{hi}h_{t-1} + b_i, \tau) \quad (1)$$

$$f_t = G(W_{xf}x_t + W_{hf}h_{t-1} + b_f, \tau) \quad (2)$$

$$\widetilde{c}_t = tanh(W_{xc}x_t + W_{hc}h_{t-1} + b_c) \quad (3)$$

$$c_t = f_t \odot c_{t-1} + i_t \odot \widetilde{c}_t \quad (4)$$

$$o_t = \sigma(W_{xo}x_t + W_{ho}h_{t-1} + b_o) \quad (5)$$

$$h_t = o_t \odot tanh(c_t) \quad (6)$$

When we consider a stochastic expansion on gate mechanisms, it is natural to structure the random variables with conditional independence and priors. For example, the input and the forget gates are both related to the cell state in the LSTM cell so that we may conjecture their correlations through a common cause prior. To our knowledge, CIFG-LSTM in Figure 2a is the first model to introduce a structured input and forget gate modeling by assuming $f_t = 1 - i_t$. This hard assignment is not a flexible correlation modeling, so this can be further extended by adopting the flexible probabilistic gate mechanism. Our source code is available at https://github.com/gtshs2/BetaLSTM.

## Methodology

First, we improve the LSTM to have more flexible gate values, which can represent skewness and multi-modality by modeling the input and the forget gate as a Beta distribution. Second, we extend the Beta distribution to incorporate the correlation between the input gate and the forget gate with the bivariate Beta distribution. Third, we introduce the prior distribution to the gate structure to keep the stochasticity and handle the mutual dependency. Our probabilistic gate model resides in a neural network cell, as Figure 3.

## Beta-LSTM

We propose a Beta-LSTM that embeds independent Beta distributions on the input and the forget gates, instead of the sigmoid function. We construct each Beta distribution with two Gamma distributions to apply the reparametrization technique.

$$U_t^{(j)} = g_j(x_t, h_{t-1}), j = 1, ..., 4 \tag{7}$$

$$u_t^{(j)} \sim \text{Gamma}(U_t^{(j)}, 1), j = 1, .., 4 \tag{8}$$

$$i_t = \frac{u_t^{(1)}}{u_t^{(1)} + u_t^{(2)}}, f_t = \frac{u_t^{(3)}}{u_t^{(3)} + u_t^{(4)}} \tag{9}$$

We formulate $U_t^{(j)}$, the shape parameter of a Gamma distribution; as a function, $g_j$ of the current input $x_t$ and the previous hidden state $h_{t-1}$. We omit the amortized inference on the rate parameter of a Gamma distribution by setting it as a constant of 1. Each $g_j$ can be a multi-layered perceptron (MLP) that combines $x_t$ and $h_{t-1}$.

As we follow the reparameterization technique of optimal mass transport (OMT) gradient estimator (Jankowiak and Obermeyer 2018) which utilize the implicit differentiation, we can compute the stochastic gradient of random variable $u_t^{(j)}$ with respect to $U_t^{(j)}$ efficiently, without inverse CDF.

## *bivariate* Beta-LSTM

Beta-LSTM improves LSTM to have more flexible input and forget gate values, but these inputs and forget gates are modeled independently, which is the same as LSTM. However, as we surveyed in the above, there is a growing interest in modeling the correlation of the gate values. To consider the correlation efficiently, we further extended Beta-LSTM to have a structured gate modeling. We adopt the bivariate Beta distribution to reflect the correlation between input and forget gates by maintaining the flexibility of the Beta distribution. We can construct a bivariate Beta distribution with three independent random variables which follow a Gamma distribution, independently (Olkin and Liu 2003). We name the bivariate Beta LSTM with three Gamma distributions as bBeta-LSTM(3G). The formulation of $U_t^{(j)}$ and $u_t^{(j)}$ is same within Equation 7,8 for all $j$.

$$i_t = \frac{u_t^{(1)}}{u_t^{(1)} + u_t^{(3)}}, f_t = \frac{u_t^{(2)}}{u_t^{(2)} + u_t^{(3)}} \tag{10}$$

The bivariate Beta distribution utilizes Gamma random variables to handle the correlation between the input and the forget gate values, but bBeta-LSTM(3G) can only model the positive correlation between 0 and 1 (Olkin and Liu 2003). In practice, for example, natural language processing, the input, and the forget gates might show either a positive or negative correlation in cases. Sequential correlated words, i.e., idioms or phrases, would prefer a positive correlation because the cell state should include both previous and current information. On the contrary, if a new important context starts, unlike the previous context, the cell state should disregard the previous information and adapt the current information. The latter case will require a negative correlation, but bBeta-LSTM(3G) lacks this functionality.

We extend the bivariate Beta distribution in bBeta-LSTM(3G) to be bBeta-LSTM(5G) that uses a bivariate Beta distribution with a more flexible covariance structure. bBeta-LSTM(5G) consists of five random variables following a Gamma distribution, and bivariate Beta distribution with five random variables can handle both negative and positive correlation (Arnold and Ng 2011). bBeta-LSTM(5G) is a generalized model of CIFG-LSTM with a probabilistic covariance model. The formulation of $U_t^{(j)}$ and $u_t^{(j)}$ is same within Equation 7,8 for all $j$.

$$i_t = \frac{u_t^{(1)} + u_t^{(3)}}{u_t^{(1)} + u_t^{(3)} + u_t^{(4)} + u_t^{(5)}} \tag{11}$$

$$f_t = \frac{u_t^{(2)} + u_t^{(4)}}{u_t^{(2)} + u_t^{(3)} + u_t^{(4)} + u_t^{(5)}} \tag{12}$$

Another advantage of using a bivariate Beta distribution as an activation function is resolving the gradient vanishing problem of LSTM. We provide a proposition that the gradient value of a gate value in bBeta-LSTM(5G) with respect to the gate parameter is larger than that of LSTM under a certain condition.

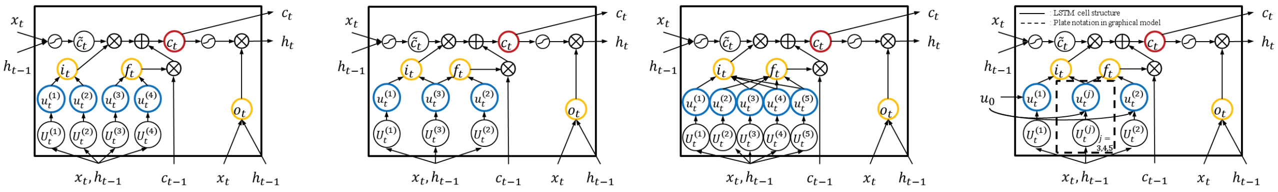
**Proposition 1.** *Let* $i_t^a(V_t)$ *and* $i_t^b(U_t^{(1:5)})$ *be the input gate of LSTM and bBeta-LSTM(5G) respectively, where* $V_t$ *and* $U_t^{(1:5)}$ *are input of each gate. Suppose that* $u_t^{(j)} < 0.8$ *or* $8 < U_t^{(j)}$ *which satisfy the* $|u_t^{(j)} - U_t^{(j)}| \leq \delta \cdot U_t^{(j)}$ *for all* $j$ *and* $\delta > 0$*. Then, for the fixed* $u_t^{(3:5)}$, $\frac{\partial i_t^b(U_t^{(1:5)})}{\partial U_t^{(1)}}|_{U_t^{(1)}=0.5}$ *has greater maximum value than* $\frac{\partial i_t^a(V_t)}{\partial V_t}$.

bBeta-LSTM(5G) considers the input gate and the forget gates as a bivariate Beta distribution, so bBeta-LSTM(5G) represents the flexible gate structure with either positive or negative correlation. Besides, stochasticity in bivariate Beta distribution can alleviate the overfitting. However, the bivariate Beta random variables with five Gamma (Eq.11,12), can has a lower variance than the bivariate Beta random variables with three Gamma (Eq.10), and the variance can be near to zero under the no regularization. The low variance can limit the advantage of stochasticity (Dieng et al. 2018), and we need additional prior model to regularize the gamma random variable $u_t^{(i)}$.

**Proposition 2.** *If all of* $u_t^{(j)}$ *have same fixed value for* $j = 1, ..., 5$*, the variance of* $i_t$ $(f_t)$ *in bBeta-LSTM(5G) is less than the variance of* $i_t$ $(f_t)$ *in bBeta-LSTM(3G).*

## *bivariate* Beta-LSTM with Structured Prior Model

Hierarchical Bayesian modeling can impose uncertainty on a model as well as a mutual dependence between variables. bBeta-LSTM(5G) has a component of probabilistic modeling, and it is easy to incorporate a prior distribution to the likelihood of the gate value. We propose bBeta-LSTM(5G)



(a) Beta-LSTM

(b) bBeta-LSTM(3G)

(c) bBeta-LSTM(5G)

(d) bBeta-LSTM(5G+p)

Figure 3: The cell structure of our proposed models. The red and yellow circle denotes a cell state and gates, respectively. The blue circle represents random variables that follow the Gamma distribution. The input and the forget gates in bBeta-LSTM(5G) and bBeta-LSTM(5G+p) shares random variables, $u^{(3)}$, $u^{(4)}$, $u^{(5)}$ and prior $u^{(0)}$.

with prior, denoted by bBeta-LSTM(5G+p), and we optimize bBeta-LSTM(5G+p) by maximizing the log marginal likelihood of the target sequence $y_{1:T}$ in Equation 13, see Figure 3d which combines the neural network gates and the random variables. Given the latent dimension of the prior, we utilize the variational method, and we optimize the evidence lower bound (ELBO) (Kingma and Welling 2014) in Equation 14 with a variational distribution, $q$, which is a feed-forward neural network with the current input $x_t$ and the previous hidden $h_{t-1}$.

$$\log p(y_{1:T}) = \log \int \prod_{t=1} p(u_t^{(1:5)}) p(y_t|u_t^{(1:5)}) du_t^{(1:5)} \quad (13)$$

$$\mathcal{L}_{ELBO} = \sum_{t=1}^{T} \mathbb{E}_{q(u_t^{(1:5)}|x_t,h_{t-1})}[p(y_t|u_t^{(1:5)})] - \mathrm{KL}[q(u_t^{(1:5)}|x_t, h_{t-1}) \,\|\, p(u_t^{(1:5)})] \quad (14)$$

We model a prior distribution of $p(u_t^{(1:5)})$, as a Gamma distribution which is a conjugate distribution of the Gamma distribution of $u_t^{(1:5)}$ in Equation 14. A Gamma distribution takes two parameters, which represent shape and rate, and our framework enables learning of the two parameters with an inference network.

**Domain Customized Structured Prior** The prior on the gate in bBeta-LSTM(5G+p) is extended to incorporate other probabilistic generative models, such as Latent Dirichlet Allocation (LDA) (Blei, Ng, and Jordan 2003) or word vector, i.e., Glove or Word2Vec. Considering the input and the forget gates reside in a LSTM cell at a certain time, $t$, the prior can be better informed by a global context extracted from $x_{1:T}$. To demonstrate this capability, we adapted Equation 14 to be as below.

$$\sum_{t=1}^{T} \mathbb{E}_{q(u_t^{(1:5)}|x_t,h_{t-1})}[p(y_t|u_t^{(1:5)})] - \lambda\{\mathrm{KL}[q(u_t^{(3:5)}|x_t, h_{t-1}) \,\|\, p(u_t^{(3:5)})] + \mathrm{KL}[q(u_t^{(1:2)}|x_t, h_{t-1}) \,\|\, p(u_t^{(1:2)}|\beta_{t-1}, \beta_t)]\}. \quad (15)$$

Here, $\lambda$ is the weight of the prior regularization, to balance the likelihood and the KL regularization term, and $\beta_t$ is the topic probability of a word at time $t$ in the sequence, which follows the definition in the original LDA. If we impose that $\beta_t$ is related prior to $u_t^{(1)}$ and $u_t^{(2)}$, which contribute to $i_t$ and $f_t$, respectively; we can directly reflect the global context to compute the input and the forget gates. To reflect the global context adequately, we compare the similarity between the topic proportion of the previous word and the current word with the radial basis kernel function (RBF). Under the prior modeling, we can also compute the input gate and the forget gates by reflecting the similarity of global topic proportions between sequential words. If the sequential words share a similar topic proportion, similar semantics, the prior with RBF kernel encourages $u_t^{(1:2)}$ to have a large value. This makes the large input and the large forget gate values, so the gating mechanisms handle the semantic composition of the previous input and the current input. While we used a pre-trained LDA model, we can learn the parameters of our proposed models and LDA parameter simultaneously. Additionally, $\beta_t$ can be substituted by a word vector, i.e., Glove.

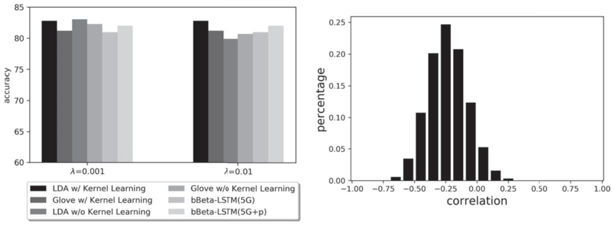


Figure 4: (Left) The sentiment classification accuracy of bBeta-LSTM variants with or without prior learning on a $0_{th}$ fold of CR dataset. (Right) The correlation of bBeta-LSTM(5G+p) on CR dataset.

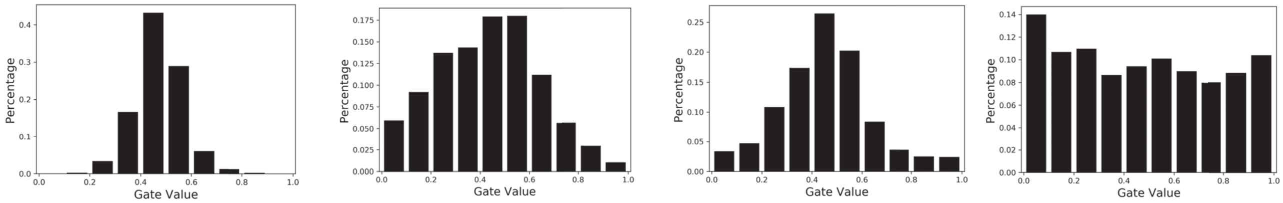
Figure 4 (Left) illustrates three insights. First, the strength of the prior should be limited by $\lambda$. Second, the prior with LDA is generally better than the prior with a static parameter, bBeta-LSTM(5G+p). Third, it is important to learn the inference model, i.e. the kernel hyperparameters used for the parameter of $p(u_t^{(1:2)}|\beta_{t-1}, \beta_t)$.

## Experiments

We compare our models and baselines, LSTM, CIFG-LSTM, $G^2$-LSTM, simple recurrent unit (SRU) (Lei et al. 2018), R-transformer (Wang et al. 2019), Batch normalized

| Models | CR | SUBJ | MR | TREC | MPQA | SST |
|---|---|---|---|---|---|---|
| LSTM | 82.91±2.40 | 92.58±0.84 | 80.37±0.98 | 94.42±1.07 | 89.38±0.55 | 88.13±0.67 |
| CIFG-LSTM | 83.28±1.79 | 92.65±0.86 | 79.86±0.91 | 94.00±0.78 | 89.14±0.91 | 87.63±0.46 |
| $G^2$-LSTM | 83.31±1.66 | 92.69±0.78 | 80.13±1.10 | 94.68±0.37 | 89.34±0.54 | 88.36±0.96 |
| Beta-LSTM | 84.45±1.87 | 93.25±0.88 | 81.12±0.93 | 94.38±0.64 | 89.66±0.49 | 88.68±0.67 |
| bBeta-LSTM(3G) | 83.63±2.14 | 93.23±0.78 | 81.47±0.92 | 94.28±0.48 | 89.41±0.91 | 88.23±0.67 |
| bBeta-LSTM(5G) | 84.48±1.96 | 92.87±0.74 | 81.05±1.05 | 94.30±0.69 | 89.42±0.66 | 88.89±0.46 |
| bBeta-LSTM(5G+p) | **84.66**±2.42 | **93.25**±0.85 | **81.59**±0.91 | **94.80**±0.47 | **89.66**±0.44 | **88.94**±0.43 |
| SRU(8-layers) | 87.00±2.24 | 93.76±0.61 | 83.14±1.53 | 94.52±0.34 | 90.39±0.55 | 89.58±0.46 |
| + bBeta-LSTM(5G+p) | **87.21**±0.78 | **93.97**±0.55 | **83.51**±1.42 | **94.80**±0.47 | **90.44**±0.79 | **89.72**±0.31 |

Table 1: Test accuracies on sentence classification task.



(a) LSTM

(b) CIFG-LSTM

(c) $G^2$-LSTM

(d) bBeta-LSTM(5G+p)

Figure 5: Histogram of input gate value on CR dataset. Our proposed model bBeta-LSTM(5G+p) shows the more flexible gate value than that of other models. CR dataset is used for the sentiment classification, and only a few words are important instead of whole words. As a result, the input gate in all models has a relatively higher portion of 0 than the portion of value 1. The bBeta-LSTM(5G+p) is more likely to have such a tendency, and it leads to better performance of bBeta-LSTM(5G+p) on CR dataset.

LSTM (BN-LSTM) (Cooijmans et al. 2017), and h-detach (Kanuparthi et al. 2019). First, we evaluate the performance of the bBeta-LSTM variants to measure the improvements from our structured gate modeling with the text classifications quantitatively and qualitatively on benchmark datasets. Second, we compare the models on polyphonic music modeling to check the performance of multi-label prediction tasks. Third, we evaluate our models on a pixel-by-pixel MNIST dataset to confirm that our model can alleviate the gradient vanishing problems, empirically. Finally, we perform the image caption generation task to check the performance on the multi-modal dataset.

## Text Classification

We compare our models on six benchmark datasets, customer reviews (CR), sentence subjectivity (SUBJ), movie reviews (MR), question type (TREC), opinion polarity (MPQA), and Stanford Sentiment Treebank (SST). For LSTM models, we use a two-layer structure with 128 hidden dimensions for each layer, following (Lei et al. 2018). We set the hidden dimensions of models to have the same number of parameters across the compared models Table 1 shows the test accuracies for the model and dataset combinations. It should be noted that bBeta-LSTM(5G+p) performs better than other models on all datasets. In particular, bBeta-LSTM(5G) and bBeta-LSTM(5G+p), which provides an inductive bias of either positive or negative correlation, shows a significant improvement for CR dataset, the sparsest dataset in the benchmarks. The performance difference between bBeta-LSTM(5G) and bBeta-LSTM(5G+p) shows the importance of the prior modeling to regularize the input and the forget gates. To check the compatibility with LSTM cell variants, we compare the performance between SRU and SRU+bBeta-LSTM(5G+p). For SRU+bBeta-LSTM(5G+p), we replace the gate structure in the SRU cell with our model, and it performs better than the original SRU for all datasets. SRU cell, which has two gate structures, is closer to GRU than LSTM, and this result demonstrates that our proposed gate structure can be compatible with other LSTM/GRU cell variants. We further examine the behavior of the bBeta-LSTM(5G+p) gate values from two perspectives of the input gate value ranges, and the input/forget correlations. Figure 4 (Right) shows the correlation of bBeta-LSTM(5G+p), and the correlation between gates can exhibit both negative and positive values in bBeta-LSTM(5G+p). Figure 5 visualizes the input gate and the forget gate values, and we observed that the input and the forget gate outputs fully utilize the range of [0,1] in bBeta-LSTM(5G+p).

To further understand the model structure and its assumptions, we performed qualitative analysis on a sentence, which has a negative sentiment in the MR dataset. Figure 6 shows the heatmap of the input gate and the forget gate for each model; and the correlation from our proposed model, bBeta-LSTM(5G+p). bBeta-LSTM(5G+p)

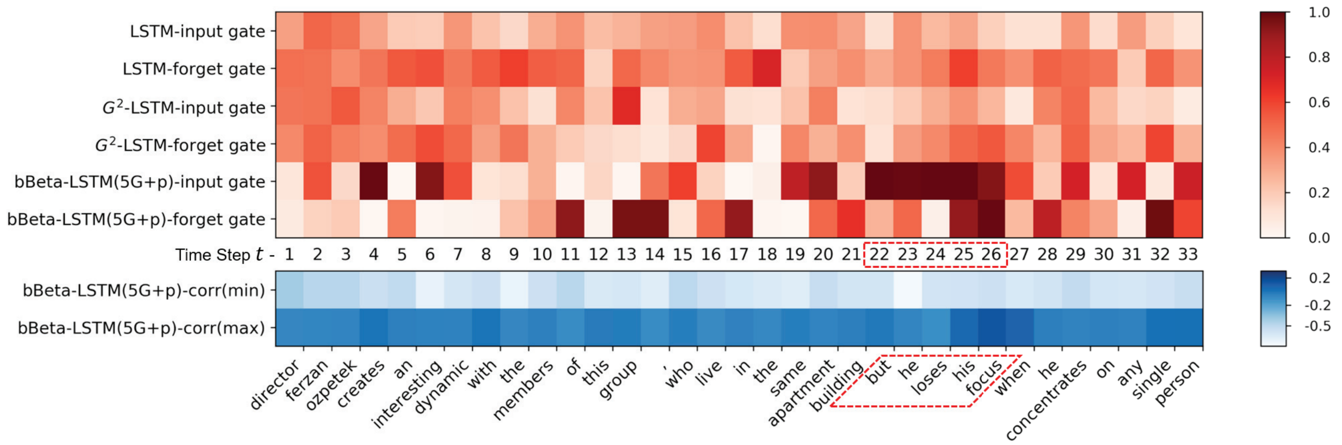


Figure 6: Visualization of input and forget gates for each model and the correlation for bBeta-LSTM(5G+p). The sentence with the negative label is designed for the sentiment classification task, and the ”but he losses his focus”$t = 22 \sim 26$ is an important part. At time step 22 (”but”), the change of context occurs, and bBeta-LSTM(5G+p) has a large input gate and relatively small forget gate to handle the context change. At time step 25 (”his”), both input gate and forget gate have high values to propagate the information ”losses his” efficiently. This is the result of a relatively large correlation value at time step 25, and this correlation helps to propagate the information through the model.

| Models | JSB | Muse | Nottingham | Piano |
|---|---|---|---|---|
| LSTM | 8.68±0.10 | 7.17±0.06 | 3.32±0.11 | 9.23±1.13 |
| CIFG-LSTM | 8.69±0.06 | 7.18±0.01 | 3.28±0.10 | 8.99±1.57 |
| $G^2$-LSTM | 8.70±0.04 | 7.14±0.01 | 3.23±0.04 | 9.00±0.84 |
| Beta-LSTM | 8.60±0.07 | 7.13±0.03 | 3.30±0.06 | 8.24±0.26 |
| bBeta-LSTM(5G) | 8.63±0.12 | 7.11±0.04 | 3.30±0.04 | 8.43±0.64 |
| bBeta-LSTM(5G+p) | **8.30**±0.01 | **7.02**±0.02 | **3.14**±0.02 | **7.65**±0.08 |
| R-Transformer | 8.26±0.03 | 7.00±0.03 | 2.24±0.01 | 7.44±0.03 |
| + bBeta-LSTM(5G+p) | **8.24**±0.01 | **6.19**±0.02 | **2.13**±0.08 | **7.32**±0.03 |

Table 2: Negative log-likelihood on polyphonic music

| Models | sMNIST | pMNIST |
|---|---|---|
| LSTM | 5.08±0.01 | 10.76±1.34 |
| CIFG-LSTM | 1.23±0.13 | 8.42±0.58 |
| $G^2$-LSTM | 3.53±1.32 | 9.47±0.03 |
| Beta-LSTM | 3.14±0.88 | 8.65±0.49 |
| bBeta-LSTM(5G) | 1.75±0.50 | 8.37±0.46 |
| bBeta-LSTM(5G+p) | **1.22**±0.25 | **7.66**±0.16 |
| BN-LSTM | 1.05±0.06 | 4.26±0.50 |
| + bBeta-LSTM(5G+p) | **0.76**±0.05 | **3.90**±0.25 |

Table 3: Test error rates on MNIST

model has a large input gate value on ”but he loses his focus” ($t = 22 \sim 26$) and a large forget gate value on 25 and 26 timestep to propagate the ”losses” information well. Because of the structured gate modeling, bBeta-LSTM(5G+p) compose the meaning of ”but he loses his focus” well. This effect originates from the structured gate modeling, which handles the correlation while other models do not model. There is a relatively large correlation in the ”his focus” ($t = 25, 26$), and as a result, both input and forget gates have large values to propagate the important information efficiently. The sentiment label for the sentence is negative, and only bBeta-LSTM(5G+p) classifies it correctly.

## Polyphonic Music

We use four polyphonic music modeling benchmark datasets: JSB Chorales, Muse, Nottingham, and Piano. Table 2 shows the test negative log-likelihood (NLL) on four music datasets. Our proposed model, bBeta-LSTM(5G+p), performs better than all other models. To compare with the pre-existing state-of-the-art model, we included the performance with R-Transformer (Wang et al. 2019) as well as R-Transformer with our gating mechanism. We replace the recurrent structure of R-transformer with our models, and our model shows better performance on all datasets. The results show high compatibility between our models and the Transformer model.

| Models | B-1 | B-2 | B-3 | B-4 | METEOR | CIDEr | ROUGE-L | SPICE |
|---|---|---|---|---|---|---|---|---|
| DeepVS (Karpathy and Li 2015) | 62.5 | 45.0 | 32.1 | 23.0 | 19.5 | 66.0 | — | — |
| ATT-FCN (You et al. 2016) | 70.9 | 53.7 | 40.2 | 30.4 | 24.3 | — | — | — |
| Show & Tell (Vinyals et al. 2015) | — | — | — | 27.7 | 23.7 | 85.5 | — | — |
| Soft Attention (Xu et al. 2015) | 70.7 | 49.2 | 34.4 | 24.3 | 23.9 | — | — | — |
| Hard Attention (Xu et al. 2015) | 71.8 | 50.4 | 35.7 | 25.0 | 23.0 | — | — | — |
| MSM (Yao et al. 2017) | 73.0 | 56.5 | 42.9 | 32.5 | 25.1 | 98.6 | — | — |
| *Show&Tell with Resnet152 (Our implementaion)* | | | | | | | — | — |
| LSTM | 72.0 | 54.6 | 39.8 | 28.8 | 24.8 | 94.7 | 52.5 | 17.9 |
| CIFG-LSTM | 71.2 | 53.9 | 39.3 | 28.5 | 24.4 | 93.0 | 51.9 | 17.7 |
| $G^2$-LSTM | 71.7 | 54.3 | 39.7 | 28.8 | 24.6 | 93.0 | 52.3 | 17.5 |
| bBeta-LSTM(5G+p) | 72.2 | 55.0 | 40.1 | 29.0 | 24.7 | 94.2 | 52.6 | 18.0 |
| h-detach(0.4) (Kanuparthi et al. 2019) | 72.2 | 55.0 | 40.9 | 30.3 | **25.2** | 97.1 | 53.0 | **18.2** |
| + bBeta-LSTM(5G+p) | **72.3** | **55.5** | **41.5** | **30.8** | **25.2** | **97.3** | **53.2** | 18.1 |
| *Show Attend Tell with Resnet152 (Our implementaion)* | | | | | | | | |
| h-detach(0.4) (Kanuparthi et al. 2019) | 73.3 | 56.7 | 42.6 | 31.8 | 25.8 | 101.2 | 54.0 | 19.3 |
| + bBeta-LSTM(5G+p) | **74.1** | **57.3** | **43.1** | **32.1** | **26.1** | **103.2** | **54.3** | **19.4** |

Table 4: Test performance on MS-COCO dataset for BLEU, METEOR, CIDEr, ROUGE-L and SPICE evaluation metric.

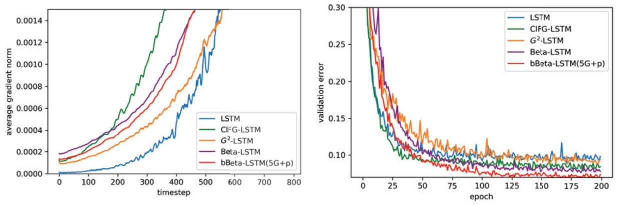


Figure 7: Average gradient norm, $\|\frac{\partial L_{ELBO}}{\partial c_t}\|$ for loss $L_{ELBO}$ over each time step. Beta-LSTM and bBeta-LSTM(5G+p) considers long-term dependency relatively well because they have larger gradients for initial timesteps, (left). bBeta-LSTM(5G+p), which incorporates the prior distribution, shows a relatively stable validation error curve, and shows the lowest validation error (right).

## Pixel by Pixel MNIST

The pixel-by-pixel MNIST task is predicting a category for a given 784 pixels. sMNIST task handles each pixel with a sequential order, and pMNIST task models each pixel in a randomly permutated order. For the LSTM baseline, we use a single-layer model with 128 hidden dimensions with Adam optimizer. Table 3 shows the test error rates for sMNIST and pMNIST, and bBeta-LSTM(5G+p) shows the best performance. Besides, we compared the performance with the BN-LSTM, which performs well on sMNIST and pMNIST dataset. When we replace the recurrent part in BN-LSTM with our model, we improve the test error rates about 27.6% (from 1.05% to 0.76% error rate) in the pMNIST dataset. Batch normalization and its variants are important in various classification tasks, and the results show that our model is well compatible with the batch normalization methodology. Left in Figure 7 shows the gradients flow for each time step and the validation error curve for each epoch on the pMNIST dataset. For the gradient flow, we calculate the Frobenius norm of the gradient $\frac{\partial L_{ELBO}}{\partial c_t}$, and we average the norm over the image instance. We found that our proposed models, Beta-LSTM, and bBeta-LSTM(5G+p), propagate the information to the early timestep, efficiently. Right in Figure 7 shows the validation error curve, and our proposed model bBeta-LSTM(5G+p), which incorporates the prior, shows the relatively stable learning curve.

## Image Captioning

We evaluate our model on the image captioning task with Microsoft COCO dataset (MS-COCO) (Lin et al. 2014). For the experiment, we split the dataset into 80,000, 5,000, 5,000 for the train, the validation, and the test dataset, respectively (Karpathy and Li 2015). We use 512 hidden dimensions for the conditional caption generation, and we also used $Resnet152$ to retrieve image feature vectors. Table 4 shows the test performance for MS-COCO dataset based on $Show\&Tell$ (Vinyals et al. 2015) and *Show Attend Tell* (Xu et al. 2015) encoder-decoder structure. Besides, to verify compatibility with other models, we re-implemented h-detach (Kanuparthi et al. 2019) and incorporate our models, bBeta-LSTM(5G+p). When we replace the LSTM of h-detach with our models, we identified the improvement in the performance of h-detach.

## Conclusion

We propose a new structured gate modeling which can improve the LSTM structure through the probabilistic modeling on gates. The gate structure in LSTM is a crucial component, and the gate value is the main controller for the information flow. While the current sigmoid gate would satisfy the boundedness, we improve the sigmoid function with the Beta distribution to add flexibility. Moreover, bBeta-LSTM enables the detailed modeling of the covariance structure between gates, and bBeta-LSTM with prior guides the learning of the covariance structure. Also, our propositions state the improved characteristics of our probabilistic gate compared to the sigmoid function. From the application perspective, imposing the correlation between the input gate and the forget gate is necessary to handle the semantic information efficiently. This work envisions how to incorporate the neural network models with probabilistic components to improve its flexibility and stability. We demonstrated the necessity and effectiveness of flexible and prior modeling of gate structure on extensive experiments.

**Acknowledgments.** This work was conducted at High-Speed Vehicle Research Center of KAIST with the support of the Defense Acquisition Program Administration and the Agency for Defense Development under Contract UD170018CD.

## References

Arnold, B. C., and Ng, H. K. T. 2011. Flexible bivariate beta distributions. *Journal of Multivariate Analysis* 102(8):1194–1202.

Battaglia, P. W.; Hamrick, J. B.; Bapst, V.; Sanchez-Gonzalez, A.; Zambaldi, V.; Malinowski, M.; Tacchetti, A.; Raposo, D.; Santoro, A.; Faulkner, R.; et al. 2018. Relational inductive biases, deep learning, and graph networks. *arXiv preprint arXiv:1806.01261*.

Blei, D. M.; Ng, A. Y.; and Jordan, M. I. 2003. Latent dirichlet allocation. *Journal of Machine Learning Research* 3:993–1022.

Chung, J.; Kastner, K.; Dinh, L.; Goel, K.; Courville, A. C.; and Bengio, Y. 2015. A recurrent latent variable model for sequential data. In *Advances in neural information processing systems*, 2980–2988.

Cooijmans, T.; Ballas, N.; Laurent, C.; Gülçehre, Ç.; and Courville, A. 2017. Recurrent batch normalization. *5th International Conference on Learning Representations, ICLR 2017,*.

Dieng, A. B.; Ranganath, R.; Altosaar, J.; and Blei, D. M. 2018. Noisin: Unbiased regularization for recurrent neural networks. In *Proceedings of the 35th International Conference on Machine Learning, ICML 2018, Stockholmsmässan, Stockholm, Sweden, July 10-15, 2018*, 1251–1260.

Gal, Y., and Ghahramani, Z. 2016. A theoretically grounded application of dropout in recurrent neural networks. In *Advances in neural information processing systems*, 1019–1027.

Greff, K.; Srivastava, R. K.; Koutník, J.; Steunebrink, B. R.; and Schmidhuber, J. 2017. Lstm: A search space odyssey. *IEEE transactions on neural networks and learning systems* 28(10):2222–2232.

Harabagiu, S. 2004. Incremental topic representations. In *Proceedings of the 20th international conference on Computational Linguistics*, 583. Association for Computational Linguistics.

Hochreiter, S., and Schmidhuber, J. 1997. Long short-term memory. *Neural computation* 9(8):1735–1780.

Jankowiak, M., and Obermeyer, F. 2018. Pathwise derivatives beyond the reparameterization trick. *arXiv preprint arXiv:1806.01851*.

Kampffmeyer, M.; Dong, N.; Liang, X.; Zhang, Y.; and Xing, E. P. 2019. Connnet: A long-range relation-aware pixel-connectivity network for salient segmentation. *IEEE Transactions on Image Processing* 28(5):2518–2529.

Kanuparthi, B.; Arpit, D.; Kerg, G.; Ke, N. R.; Mitliagkas, I.; and Bengio, Y. 2019. h-detach: Modifying the LSTM gradient towards better optimization. In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*.

Karpathy, A., and Li, F. 2015. Deep visual-semantic alignments for generating image descriptions. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2015, Boston, MA, USA, June 7-12, 2015*, 3128–3137.

Kingma, D. P., and Welling, M. 2014. Auto-encoding variational bayes. In *2nd International Conference on Learning Representations, ICLR 2014, Banff, AB, Canada, April 14-16, 2014, Conference Track Proceedings*.

Lei, T.; Zhang, Y.; Wang, S. I.; Dai, H.; and Artzi, Y. 2018. Simple recurrent units for highly parallelizable recurrence. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, 4470–4481. Brussels, Belgium: Association for Computational Linguistics.

Li, Z.; He, D.; Tian, F.; Chen, W.; Qin, T.; Wang, L.; and Liu, T.-Y. 2018. Towards binary-valued gates for robust lstm training. *arXiv preprint arXiv:1806.02988*.

Lin, T.; Maire, M.; Belongie, S. J.; Hays, J.; Perona, P.; Ramanan, D.; Dollár, P.; and Zitnick, C. L. 2014. Microsoft COCO: common objects in context. In *Computer Vision - ECCV 2014 - 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part V*, 740–755.

Olkin, I., and Liu, R. 2003. A bivariate beta distribution. *Statistics & Probability Letters* 62(4):407–412.

Park, S.; Song, K.; Ji, M.; Lee, W.; and Moon, I. 2019. Adversarial dropout for recurrent neural networks. *CoRR* abs/1904.09816.

Serban, I. V.; Sordoni, A.; Lowe, R.; Charlin, L.; Pineau, J.; Courville, A.; and Bengio, Y. 2017. A hierarchical latent variable encoder-decoder model for generating dialogues. In *Thirty-First AAAI Conference on Artificial Intelligence*.

Vinyals, O.; Toshev, A.; Bengio, S.; and Erhan, D. 2015. Show and tell: A neural image caption generator. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2015, Boston, MA, USA, June 7-12, 2015*, 3156–3164.

Wang, Z.; Ma, Y.; Liu, Z.; and Tang, J. 2019. R-transformer: Recurrent neural network enhanced transformer. *arXiv preprint arXiv:1907.05572*.

Wolter, M., and Yao, A. 2018. Complex gated recurrent neural networks. In Bengio, S.; Wallach, H.; Larochelle, H.; Grauman, K.; Cesa-Bianchi, N.; and Garnett, R., eds., *Advances in Neural Information Processing Systems 31*. Curran Associates, Inc. 10536–10546.

Xu, K.; Ba, J.; Kiros, R.; Cho, K.; Courville, A. C.; Salakhutdinov, R.; Zemel, R. S.; and Bengio, Y. 2015. Show, attend and tell: Neural image caption generation with visual attention. In *Proceedings of the 32nd International Conference on Machine Learning, ICML 2015, Lille, France, 6-11 July 2015*, 2048–2057.

Yao, T.; Pan, Y.; Li, Y.; Qiu, Z.; and Mei, T. 2017. Boosting image captioning with attributes. In *IEEE International Conference on Computer Vision, ICCV 2017, Venice, Italy, October 22-29, 2017*, 4904–4912.

You, Q.; Jin, H.; Wang, Z.; Fang, C.; and Luo, J. 2016. Image captioning with semantic attention. In *2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016*, 4651–4659.