Autoregressive Text Generation Beyond Feedback Loops

Autoregressive Text Generation Beyond Feedback Loops

Florian Schmidt
Department of Computer Science
ETH Zürich
florian.schmidt@inf.ethz.ch

Stephan Mandt
Department of Computer Science
University of California, Irvine
mandt@uci.edu

Thomas Hofmann
Department of Computer Science
ETH Zürich
thomas.hofmann@inf.ethz.ch

Abstract

Autoregressive state transitions, where predictions are conditioned on past predictions, are the predominant choice for both deterministic and stochastic sequential models. However, autoregressive feedback exposes the evolution of the hidden state trajectory to potential biases from well-known train-test discrepancies. In this paper, we combine a latent state space model with a CRF observation model. We argue that such autoregressive observation models form an interesting middle ground that expresses local correlations on the word level but keeps the state evolution non-autoregressive. On unconditional sentence generation we show performance improvements compared to RNN and GAN baselines while avoiding some prototypical failure modes of autoregressive models.[1]

1 Introduction

Sequential autoregressive models express predictions of observations based on past predictions. They are the predominant architecture for text generation in a maximum likelihood setup (Graves, 2013; Sutskever et al., 2014) and are used in machine translation (Bahdanau et al., 2015; Vaswani et al., 2017), summarization (Rush et al., 2015), and dialogue systems (Serban et al., 2016).

An immediate consequence of combining autoregressive modeling and maximum likelihood training is that past observations enter the loss functions as ground-truth, not predicted observations (Goodfellow et al., 2016). This discrepancy is often summarized as *teacher-forcing* and the bias it implies is referred to as *exposure-bias* (Ranzato et al., 2016; Goyal et al., 2016).

[1]Code and generated sentences available at https://github.com/schmiflo/crf-generation

The standard methodology to turn a sequential model into an autoregressive one is to introduce a *feedback loop*, where one provides the last predicted token as a feature to the computation of the next state (Graves, 2013). The ground-truth observations become effectively input features for the evolution of the hidden state trajectory at training time. Several attempts have been made to introduce robustness with respect to the model's predictions by leaving the maximum likelihood framework, either implicitly (Bengio et al., 2015; Bowman et al., 2016) or explicitly (Goyal et al., 2016; Leblond et al., 2018). Nevertheless, the same feedback mechanisms have been adopted in latent sequential models where they obfuscate the true stochasticity of transitions during training. Non-autoregressive sequence models have recently regained attention for unconditional (Schmidt and Hofmann, 2018; M. Ziegler and M. Rush, 2019) and conditional (Lee et al., 2018) generation.

We argue that there is an interesting intermediate regime between feedback-driven autoregressive models and completely non-autoregressive models, namely modeling temporal correlations as part of the *observation model*. We propose a neural CRF observation model that leverages word-embeddings to explain local word correlations in a global sequence score. We show how training and generation can be performed efficiently. The result is an autoregressive model that keeps the hidden state evolution less affected by observation noise while generating coherent word sequences.

2 Related Work

Conditional Random Fields (CRF) were originally introduced by Sha and Pereira (2003) to overcome

Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, pages 3400–3406, Hong Kong, China, November 3–7, 2019.

label bias, a shortcoming of locally normalized observation models. They have been applied and integrated into neural-network architectures (Ma and Hovy, 2016; Huang et al., 2017) in various sequence labeling tasks (Goldman and Goldberger, 2017) where the observation space exhibits small cardinality (typically tens to hundreds).

The importance of global normalization for sequence generation has only lately been emphasized, most notably by Wiseman and Rush (2016) for conditional generation in a learning-as-search-optimization framework and by (Andor et al., 2016) for parsing.

Word-embeddings have been reported as excellent dense representations of sparse co-occurrence statistics within several learning frameworks (Mikolov et al., 2013; Pennington et al., 2014). Using embeddings in pairwise potentials has been proposed by Goldman and Goldberger (2017), but they do not compute the true log-likelihood during training as we do. Similar techniques have been applied for various message passing schemata (Kim et al., 2017; Domke, 2013).

Local correlations such as our pairwise potentials have been used by (Noraset et al., 2018), yet as an auxiliary loss and not for model design.

Other approaches to tackle teacher-forcing have been proposed in an adversarial setting (Goyal et al., 2016), in search based optimization (Leblond et al., 2018) and in a reinforcement learning setting (Rennie et al., 2016).

3 Model

Latent sequential models for text generation typically consist of two parts: A mechanism for generating a latent hidden state trajectory $\mathbf{h} = \mathbf{h}_{1:T}$, and an observation model. The latter predicts the data $w = w_{1:T}$ given the latent states. The most simple dependency structure for such a model is that of an Hidden Markov Model, which breaks into transitions $p(\mathbf{h}_t|\mathbf{h}_{t-1})$ and observations $p(w_t|\mathbf{h}_t)$. In contrast, models with *autoregressive transitions* factorize as

$$p(w, \mathbf{h}) = \prod_{t=1}^{T} p(w_t|\mathbf{h}_t)p(\mathbf{h}_t|\mathbf{h}_{t-1}, w_{t-1}) \,. \quad (1)$$

The result is a next-state distribution with dependencies identical to deterministic RNN transitions $\mathbf{h}_t = F(\mathbf{h}_{t-1}, w_{t-1})$ and indeed similar neural networks can be used to parametrize a simple, e.g., Gaussian distribution (Fraccaro et al., 2016).

As a negative consequence, we inherit teacher-forcing. This comes with aforementioned biases and also conflicts with our notion of uncertainty in $p(\mathbf{h}_t|\mathbf{h}_{t-1}, w_{t-1})$ which during training solely depends on the continuous parameters (i.e. a mean and a variance), but is greatly affected by the discrete sampling noise in w_{t-1} at test time.

Autoregressive observation model We consider an alternative to autoregressive feedback mechanisms such as (1), where predictions are directly injected into states. We write

$$p(w, \mathbf{h}) = p(w|\mathbf{h}) \prod_{t=1}^{T} p(\mathbf{h}_t|\mathbf{h}_{t-1}) \quad (2)$$

assuming only Markovian transitions and focus on finding a powerful observation model instead. Crucially, since the state space model is not affected by previous outputs, word coherence may be lost when simply factorizing as in $p(w|\mathbf{h}) = \prod_t p(w_t|\mathbf{h}_t)$, i.e. with independent soft-max factors $p(w_t|\mathbf{h}_t) \propto \exp \psi(w_t, \mathbf{h}_t)$ where $\psi(w_t, \mathbf{h}_t) = \mathbf{x}(w_t)^\top \mathbf{h}_t$. However, a natural extension can be found by reformulating local normalization as a form of global normalization without correlations across time

$$p(w|\mathbf{h}) = \prod_{t=1}^{T} \frac{\exp \psi(w_t, \mathbf{h}_t)}{\sum_{w'_t} \exp \psi(w'_t, \mathbf{h}_t)} \quad (3)$$

$$= \frac{\exp S(w, \mathbf{h})}{\sum_{w'} \exp S(w', \mathbf{h})} \quad (4)$$

where $S = \sum_{t=1}^{T} \psi(w_t, \mathbf{h}_t)$ contains no dependencies between w_t and $w_{t'}$ for $t \neq t'$. As soon as we add word-correlations to S, we obtain a truly global observation model that cannot be expressed in the form of (3).

3.1 CRF Observation Model

Equation (4) describes a conditional random field (CRF) with an energy function S (Sha and Pereira, 2003). We consider up to pairwise interactions between consecutive words

$$S(w; \mathbf{h}) = \sum_{t=1}^{T} \psi(w_t; \mathbf{h}_t) + \psi(w_{t-1}, w_t; \mathbf{h}_{t-1:t}) \quad (5)$$

The potentials ψ reflect the independence assumptions among w and determine the complexity of the normalizer $Z = \sum_{w'} \exp S(w')$. Fortunately, for chain-like interactions such as (5), efficient dynamic programming routines are available.

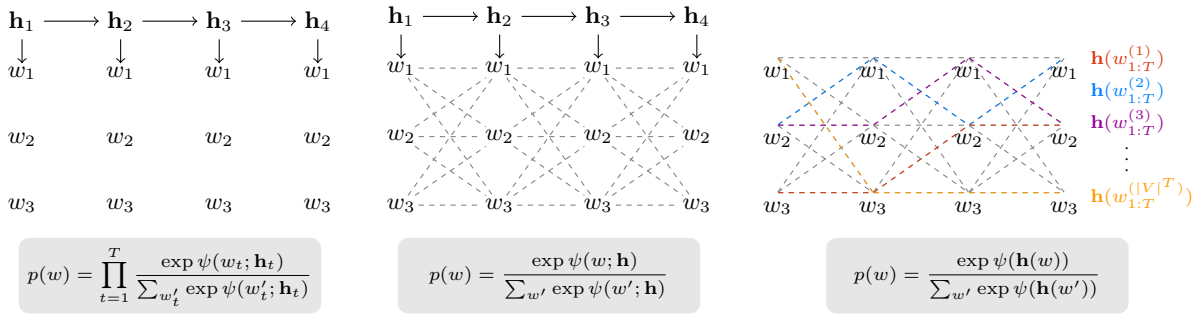

Figure 1: Schematic comparison of differently normalized architectures. We sketch trellis diagrams for $V = \{w_1, w_2, w_3\}$ and $T = 4$. Dashed lines indicate autoregressive dependencies in the log-likelihood computation. **Left**: Standard RNN with soft-max observations. Since the model is locally normalized, the trellis diagram does not unfold across time-steps. **Middle**: Our proposed CRF model. The potentials only span across pairs, but the normalization is global and can be computed exactly and efficiently. **Right**: An intractable globally normalized model in which fully-connected potentials $\psi(\mathbf{h}(w))$ are obtained from an RNN. Computing a single $p(w)$ would require running the RNN $|V|^T$ times. We highlight four runs for illustration.

Two properties set our model apart from feedback-driven autoregressive models. First, although ψ captures only pairwise interactions, a state $\mathbf{h}_t$ will not only affect future observations but also all *past* observations through the global coupling. Second, our model implicitly considers *all* possible sequences w also at training time due to the global normalizer Z.

3.2 Sampling

Given a trained model, we can perform ancestral sampling via $\mathbf{h} \sim p(\mathbf{h})$ and $w \sim p(w|\mathbf{h})$. However, CRFs are undirected graphical models not designed with generation in mind and therefore we first need to derive ancestral sampling for $p(w|\mathbf{h})$. We can always write $p(w|\mathbf{h}) = \prod_t p(w_t|w_{1:t-1}, \mathbf{h})$ and find the factors

$$p(w_t|w_{1:t-1}, \mathbf{h}) = e^{\psi(w_{t-1}, w_t) + \psi(w_t)} \frac{\beta_{t+1}(w_t)}{\beta_t(w_{t-1})} \quad (6)$$

where

$$\beta_t(w_{t-1}) = \sum_{w_t} e^{\psi(w_{t-1}, w_t) + \psi(w_t)} \beta_{t+1}(w_t) \quad (7)$$

with special cases $\beta_1(w_0) = 1$ and $\beta_{T+1}(w_T) = Z$ are the backwards probabilities we anyway need to compute for (4). Not surprisingly, multiplying (6) for $t = 1:T$ lets all β terms cancel except for $1/Z$ and we recover (4). However, this form is more amendable to sampling[2] and reveals an interesting property of globally normalized models: While the chain rule always allows to write such models autoregressively, we must expect a factor – here $\beta_{t+1}(w_t)$ – that implicitly marginalizes out future observations to assess compatibility with a specific next word w_t. Tractability of this factor is key to obtain a tractable model and is traded for expressiveness. While locally normalized models are on one end of the spectrum, a globally normalized with fully-connected potentials $\psi(\mathbf{h}(w))$ is on the other end. Such models employ an RNN in *each* potential to obtain an un-normalized score ψ from states $\mathbf{h}$ and have been investigated in conditional generation where argmax-decoding rather than sampling is requried (Wiseman and Rush, 2016). Figure 1 shows the dependencies of the two extremes with our model in the middle.

3.3 Embedding-based Local Correlations

Often pairwise potentials can be parametrized directly, i.e. as $\psi(w_i, w_j) = \mathbf{A}_{ij}$ for some parameter matrix $\mathbf{A} \in \mathbb{R}^{V \times V}$. However, in our setting this is problematic for two reasons. First, $|V|^2$ parameters are impractical in terms of model size for most vocabularies. Second, computations involving $\mathbf{A}$ are central to the complexity of computing log-likelihood during training. Namely, to compute the normalizer Z, we need to compute all β quantities in (7). Identifying $\beta_t(w_{t-1})$ as a $|V|$-dimensional vector $\boldsymbol{\beta}_t$, we can write the summation in (7) as a matrix-vector product

$$\boldsymbol{\beta}_t = \mathbf{T}(\mathbf{o}_t \odot \boldsymbol{\beta}_{t+1}) \quad (8)$$

where $\odot$ is an element-wise product, $\mathbf{o}_t$ are the unary potentials $\psi(w_t)$ written as a vector and $\mathbf{T} = \exp \mathbf{A}$ element-wise. We observe, computing Z naively requires $\mathcal{O}(|V|^2 T)$ operations.

[2] In fact, one can train on (6) instead of (4). However, in our experiments we found the latter global normalization to be much more stable numerically.

To overcome the above shortcomings, we propose to factorize $\mathbf{T}$ as

$$\mathbf{T} = \mathbf{X}^\top \mathbf{S}(\mathbf{h}_{t-1}, \mathbf{h}_t)\mathbf{Y} \tag{9}$$

into context-independent d-dimensional embeddings $\mathbf{X}, \mathbf{Y} \in \mathbb{R}^{d \times |V|}$ and a context-dependent $d \times d$ interaction matrix computed by a neural network $\mathbf{S} : \mathbb{R}^{d'} \times \mathbb{R}^{d'} \to \mathbb{R}^{d \times d}$. This reduces the memory requirement to $\mathcal{O}(d|V|)$ and compute time to $\mathcal{O}(d|V|T)$, which is comparable to computing standard soft-max logits. As an additional benefit we can initialize $\mathbf{X}$ and $\mathbf{Y}$ with pre-trained word-embeddings, a technique often reported to improve convergence. Sine $\mathbf{A}$ does not have more structure than being strictly positive element-wise, it is sufficient to use strictly positive activation functions around the layers in (8) to obtain a valid factorization.

3.4 Training

As is standard for latent sequential models, we use variational inference for training (Blei et al., 2017; Zhang et al., 2018). We introduce a parametrized approximate inference model $q(\mathbf{h}|w)$ to maximize the evidence lower bound (ELBO) for a sampled trajectory instead of maximizing the marginal across all trajectories:

$$\log p(w) = \int p(w, \mathbf{h}) d\mathbf{h} \tag{10}$$

$$\geq \mathbb{E}_q \left[\log p(w|\mathbf{h}) + \log \frac{p(\mathbf{h})}{q(\mathbf{h}|w)} \right] \tag{11}$$

The first term of (11) measures reconstruction while the second measures the discrepancy between the trajectories implied by the inference model q and the generative model p. The exact form of $p(\mathbf{h})$ depends on its factorization and if it is autoregressive but for us simply $p(\mathbf{h}) = \prod_t p(\mathbf{h}_t|\mathbf{h}_{t-1})$, which casts us as an autoregressive extension of Schmidt and Hofmann (2018).

Inference model Like (Fraccaro et al., 2016), we choose q to factorize as the true posterior

$$q(\mathbf{h}|w) = \prod_{t=1}^{T} q(\mathbf{h}_t|\mathbf{h}_{t-1}, w_{t:T}) \tag{12}$$

where $w_{1:T}$ is encoded using an RNN running backwards in time to parameterize mean and variance of a Gaussian for $q(\mathbf{h}_t|\mathbf{h}_{t-1}, w_{t:T})$. For optimization we follow existing work (Fraccaro et al., 2016; Goyal et al., 2017) and use the re-parametrization trick (Rezende et al., 2014; Kingma et al., 2016) to perform a stochastic gradient step on (11) with Adam (Kingma and Ba, 2014) using a single trajectory.

4 Experiments

Exposure-bias can be summarized as over-confident conditioning on "pseudo" predictions during training. The strength of the bias depends on the informativeness of such predictions, which in turn depends on the remaining context provided.

We test our proposed method on *unconditional* generation which does not provide context such as a source sentence to narrow down possible outputs a priori. Hence, potential biases are more pronounced and generation is isolated from effects induced by i.e. a translation or summarization task.

Setup Unconditional generation is still considered a challenging task for both, GANs and latent stochastic models, (Fedus et al., 2018) and standard RNNs form a very competitive baseline (Semeniuta et al., 2018). To obtain a homogeneous text dataset of low complexity we extract the plain text (text and hypothesis) from the Standard SNLI dataset (Bowman et al., 2015) (For details and samples see Appendix A).

Baselines We compare against a GRU (an LSTM performed on par) standard RNN of matching state size denoted DRNN. We also include SeqGAN[3] (Yu et al., 2017), a popular GAN architecture for unconditional generation. Further, we restrict our model to unary potentials to obtain a non-autoregressive state space model similar to that of Schmidt and Hofmann (2018), denoted SSM. Finally, 2-GRAM is a bi-gram language model and ORACLE is held-out data, which represents the gold-standard for unconditional generation.

Parameterization We use 16-dimensional latent states, pre-train 100-dimensional GloVe embeddings and use word and context vectors for $\mathbf{Y}$ and $\mathbf{X}$. For $\mathbf{S}$ we found a diagonal matrix to perform best. In this case, the symmetry of $\mathbf{T}$ is broken by larger unary potentials. While we find larger word embedding dimensionality to improve performance, the model does not benefit from more latent dimensions as an RNN does from

[3] We use the hyper-parameters recommended by the authors even though the state size is larger than ours.

hidden dimensions, a known issue of deep latent variable models (Schmidt and Hofmann, 2018; M. Ziegler and M. Rush, 2019).

4.1 Qualitative Results

Table 1 shows selected output generated by our model (See Appendix B for more output). While

a dog runs .
the children are alone .
the man is being beaten .
the man is inside working onstage .
the dog is outside with his girlfriend .
two dogs going swimming in an open-air festival .
a young lady wearing a pink shirt is studying .

Table 1: Output of our model of different length.

many of our sentences are grammatical and mimic those of the dataset we note that the corpus is not large enough to learn common sense and all models including the baselines sometimes generate output such as *two men are burning snow*.

4.2 Quantitative Results

Perplexity under external language models is the standard metric to evaluate unconditional output (Fedus et al., 2018) and we use Kneser-Ney-smoothed models up to[4] $n = 3$ estimated on the training data using SRILM (Stolcke, 2002).

In addition, we propose to estimate some important aggregate statistics easily verifiable against the real data. We choose length l and percentage of unique sentences ρ_{UNI} to assess diversity and percentage of token repetitions ρ_{REP} to adress a failure mode often found in generative models (Tu et al., 2016). Table 2 shows the results.

	PPL_2	PPL_3	ρ_{REP}	l	ρ_{UNI}
SSM+CRF	40.1	41.9	0.35	8.4	98
SSM	158.5	172.2	9.20	8.9	100
DRNN	47.1	43.5	0.78	8.7	99
SEQGAN-20E	22.4	23.1	0.63	5.7	58
SEQGAN-200E	53.0	57.0	6.88	7.1	80
2-GRAM	34.4	46.3	0.27	8.0	82
ORACLE	26.7	17.7	0.17	8.9	99

Table 2: Our model **SSM+CRF** evaluated against the baselines on 100K generated sentences each: Perplexity of output under external language model PPL_n, percentage of repeated tokens per sentence ρ_{REP}, length l, and percentage of unique sentences ρ_{UNI}. All statistics should be compared to ORACLE, a held-out data split.

[4]We find that the data is too sparse to train 4-gram language models as measured on a test-set.

5 Discussion and Future Work

In terms of perplexity our model clearly improves over SSM, outperforms DRNN as measured by bigram statistics, and is on par with it in terms of trigram statistics. Of course, 2-GRAM excels in terms of bigram statistics, yet falls behind on longer statistics. This confirms that our model can learn beyond pairwise interactions through the latent chain. In addition, through our explicit model of pairwise interaction we obtain repetitions ρ_{REP} significantly closer to the real data distribution.

For SEQGAN we report after 20 epochs (as used by the authors) and 200 epochs. We observe in general shorter output with more repetition (i.e. of words *are*, *is* and *up*) and note that depending on training time the stellar fluency is traded with a significant bias on length l and very poor diversity ρ_{UNI}, a tendency also observed by Xu et al. (2018) and possibly related to the choice of temperature parameter (Caccia et al., 2018). While it is not our goal to provide a deeper analysis of GANs here, the example shows how unconditional generation can reveal tradeoffs not present in a conditional setting.

Future Work We have shown that autoregressive predictions expressed in the observation model instead of hidden states deliver better results on a simple corpus. In particular, mistakes at the bigram-level, such as repetitions, are avoided and we suspect that more densely connected CRFs allow to extend these promising results to more complex patterns found in more complex corpora. In future work we plan to investigate if CRF variants such as (Belanger et al., 2017) or (Krähenbühl and Koltun, 2012) can be adapted to allow efficient sampling and to scale to word vocabulary sizes.

6 Conclusion

We have shown an alternative methodology to autoregressive modeling that avoids exposure-bias in hidden states by design through a globally normalized observation model. We derived a sampling method and an efficient embedding-based parameteriation of CRFs to trade expressiveness with tractability. On an unconditional generation task, we obtain better results than a deterministic RNN in a low-dimensional setting and more consistent results than a GAN baseline. Finally, we have pointed into directions on how to capture more complex correlations.

References

Daniel Andor, Chris Alberti, David Weiss, Aliaksei Severyn, Alessandro Presta, Kuzman Ganchev, Slav Petrov, and Michael Collins. 2016. Globally normalized transition-based neural networks. *CoRR*, abs/1603.06042.

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2015. Neural machine translation by jointly learning to align and translate. In *ICLR*.

David Belanger, Bishan Yang, and Andrew McCallum. 2017. End-to-end learning for structured prediction energy networks. In *ICML*.

Samy Bengio, Oriol Vinyals, Navdeep Jaitly, and Noam Shazeer. 2015. Scheduled sampling for sequence prediction with recurrent neural networks. In *NIPS*.

David M Blei, Alp Kucukelbir, and Jon D McAuliffe. 2017. Variational inference: A review for statisticians. *Journal of the American Statistical Association*.

Samuel R. Bowman, Gabor Angeli, Christopher Potts, and Christopher D. Manning. 2015. A large annotated corpus for learning natural language inference. In *EMNLP*.

Samuel R. Bowman, Luke Vilnis, Oriol Vinyals, Andrew M. Dai, Rafal Józefowicz, and Samy Bengio. 2016. Generating sentences from a continuous space. In *ACL*.

Massimo Caccia, Lucas Caccia, William Fedus, Hugo Larochelle, Joelle Pineau, and Laurent Charlin. 2018. Language gans falling short. *CoRR*, abs/1811.02549.

Justin Domke. 2013. Learning graphical model parameters with approximate marginal inference. In *IEEE Transactions on Pattern Analysis and Machine Intelligence*.

William Fedus, Ian J. Goodfellow, and Andrew M. Dai. 2018. Maskgan: Better text generation via filling in the _____. In *ICLR*.

Marco Fraccaro, Søren Kaae Sø nderby, Ulrich Paquet, and Ole Winther. 2016. Sequential neural models with stochastic layers. In *NIPS*.

Eran Goldman and Jacob Goldberger. 2017. Structured image classification from conditional random field with deep class embedding. *arXiv preprint arXiv:1705.07420*.

Ian Goodfellow, Yoshua Bengio, and Aaron Courville. 2016. *Deep Learning*. MIT Press.

Anirudh Goyal, Alex Lamb, Ying Zhang, Saizheng Zhang, Aaron C. Courville, and Yoshua Bengio. 2016. Professor forcing: A new algorithm for training recurrent networks. In *NIPS*.

Anirudh Goyal, Alessandro Sordoni, Marc-Alexandre Côté, Nan Rosemary Ke, and Yoshua Bengio. 2017. Z-forcing: Training stochastic recurrent networks. In *NIPS*.

Alex Graves. 2013. Generating sequences with recurrent neural networks. *arXiv preprint arXiv:1308.0850*.

Zhiheng Huang, Wei Xu, and Kai Yu. 2017. Bidirectional LSTM-CRF models for sequence tagging. In *First Workshop on Subword and Character Level Models in NLP*.

Yoon Kim, Carl Denton, Luong Hoang, and Alexander M. Rush. 2017. Structured attention networks. In *ICLR*.

Diederik P. Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. In *ICLR*.

Diederik P. Kingma, Tim Salimans, and Max Welling. 2016. Improving variational inference with inverse autoregressive flow. In *NIPS*.

Philipp Krähenbühl and Vladlen Koltun. 2012. Efficient inference in fully connected crfs with gaussian edge potentials. In *NIPS*.

Rémi Leblond, Jean-Baptiste Alayrac, Anton Osokin, and Simon Lacoste-Julien. 2018. SEARNN: training rnns with global-local losses. In *ICLR*.

Jason Lee, Elman Mansimov, and Kyunghyun Cho. 2018. Deterministic non-autoregressive neural sequence modeling by iterative refinement. In *EMNLP*.

Zachary M. Ziegler and Alexander M. Rush. 2019. Latent normalizing flows for discrete sequences. *arXiv preprint arXiv:1901.10548*.

Xuezhe Ma and Eduard H. Hovy. 2016. End-to-end sequence labeling via bi-directional lstm-cnns-crf. In *ACL*.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013. Distributed representations of words and phrases and their compositionality. In *NIPS*.

Thanapon Noraset, David Demeter, and Doug Downey. 2018. Controlling global statistics in recurrent neural network text generation.

Jeffrey Pennington, Richard Socher, and Christopher D. Manning. 2014. Glove: Global vectors for word representation. In *EMNLP*.

Marc'Aurelio Ranzato, Sumit Chopra, Michael Auli, and Wojciech Zaremba. 2016. Sequence level training with recurrent neural networks. In *ICLR*.

Steven J. Rennie, Etienne Marcheret, Youssef Mroueh, Jerret Ross, and Vaibhava Goel. 2016. Self-critical sequence training for image captioning. *CoRR*, abs/1612.00563.

Danilo Jimenez Rezende, Shakir Mohamed, and Daan Wierstra. 2014. Stochastic back-propagation and variational inference in deep latent gaussian models. In *ICML*.

Alexander M. Rush, Sumit Chopra, and Jason Weston. 2015. A neural attention model for abstractive sentence summarization. In *EMNLP*.

Florian Schmidt and Thomas Hofmann. 2018. Deep state space models for unconditional word generation. In *NeurIPS*.

Stanislau Semeniuta, Aliaksei Severyn, and Sylvain Gelly. 2018. On accurate evaluation of gans for language generation. In *ICML workshop on Theoretical Foundations and Applications of Deep Generative Models*.

Iulian Vlad Serban, Alessandro Sordoni, Yoshua Bengio, Aaron C Courville, and Joelle Pineau. 2016. Building end-to-end dialogue systems using generative hierarchical neural network models. In *AAAI*.

Fei Sha and Fernando Pereira. 2003. Shallow parsing with conditional random fields. In *NAACL*.

Andreas Stolcke. 2002. Srilm – an extensible language modeling toolkit. In *ICSLP*.

Ilya Sutskever, Oriol Vinyals, and Quoc V. Le. 2014. Sequence to sequence learning with neural networks. In *NIPS*.

Zhaopeng Tu, Yang Liu, Lifeng Shang, Xiaohua Liu, and Hang Li. 2016. Neural machine translation with reconstruction. *CoRR*, abs/1611.01874.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *NIPS*.

Sam Wiseman and Alexander M. Rush. 2016. Sequence-to-sequence learning as beam-search optimization. In *EMNLP*.

Jingjing Xu, Xu Sun, Xuancheng Ren, Junyang Lin, Bingzhen Wei, and Wei Li. 2018. DP-GAN: diversity-promoting generative adversarial network for generating informative and diversified text. In *EMNLP*.

Lantao Yu, Weinan Zhang, Jun Wang, and Yong Yu. 2017. Seqgan: Sequence generative adversarial nets with policy gradient. In *AAAI*.

Cheng Zhang, Judith Butepage, Hedvig Kjellstrom, and Stephan Mandt. 2018. Advances in variational inference. *IEEE transactions on pattern analysis and machine intelligence*.