Supporting Information

Bidirectional Molecule Generation with Recurrent Neural Networks

Francesca Grisoni*, Michael Moret, Robin Lingwood, Gisbert Schneider*

ETH Zurich, Department of Chemistry and Applied Biosciences,
RETHINK, Vladimir-Prelog-Weg 4, 8093, Zurich, Switzerland

*francesca.grisoni@pharma.ethz.ch; gisbert.schneider@pharma.ethz.ch

Table of Contents

This document is the Supporting Information of an unedited Author's version of a Submitted Work that was subsequently accepted for publication in the Journal of *Chemical Information and Modeling*, copyright © American Chemical Society after peer review. To access the final edited and published work see https://pubs.acs.org/doi/10.1021/acs.jcim.9b00943.

Model Architecture and Training

Architecture details

Table S1. Details on the architecture of the Forward RNN, NADE and FB-RNN models.

Name	NADE and Forward RNN		FB-RNN	
	No. Units	*No. Parameters*	*No. Units*	*No. Parameters*
BatchNormalization 1	55	110	110	220
LSTM 1	256 or 512	320512	256 or 512	376832
LSTM 2	256 or 512	526336	256 or 512	526336
BatchNormalization 2	256	512	256	512
Linear Layer	55	14080	55	28160

Table S2. Details on the architecture of the BIMODAL networks.

Name	BIMODAL	
	No. Units	*No. Parameters*
BatchNormalization 1	55	110
LSTM 1 Forward	128 or 256	94720
LSTM 1 Backward	128 or 256	94720
LSTM 2	128 or 256	132096
LSTM 2 Backward	128 or 256	132096
BatchNormalization 2	256	512
Linear Layer	55	14080

Cross-entropy Loss function

Models were trained using cross-entropy loss (*L*) for performance optimization, calculated as follows:

$$L = -\sum_{t=1}^{T} \log P(x_{t+1} = \hat{x}_{t+1} | x_1, \ldots, x_t) \qquad \text{(S1)}$$

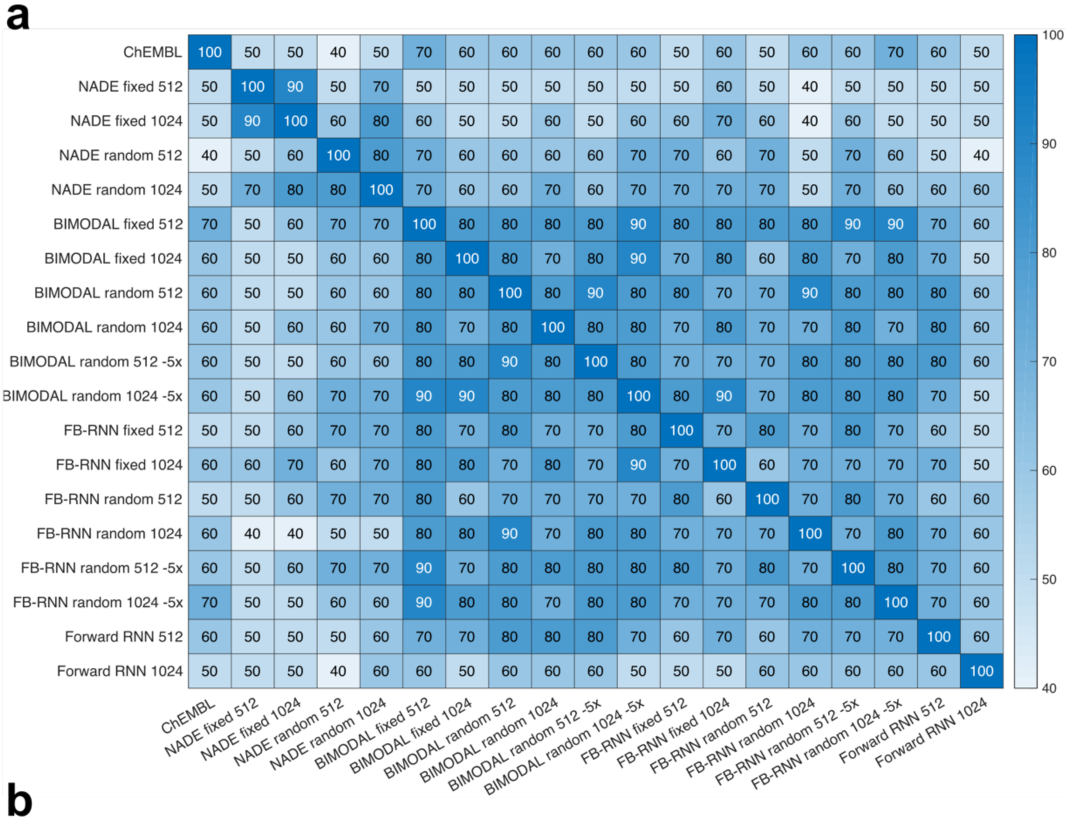
where $\hat{x}_{t+1}$ corresponds to the true token at step t+1. L ranges from 0 to 1 and it increases as the predicted probability diverges from the actual token. Cross-entropy loss was computed in both five-fold cross-validation (random partitioning protocol) and fitting. Models were trained up to 10 epochs, where L converged for all the cases.

Statistical tests details

For each set of values, normality was assessed with the Kolmogorov-Smirnov test. In the case of normality (α = 0.05), after confirming the homoscedasticity with Bartlett analysis of variance (α = 0.05), statistical differences were checked with Analysis of Variance (ANOVA). If any statistically significant difference was identified by ANOVA (α = 0.05), significant differences between methods were tested with Tukey HSD post-hoc analysis. No dataset showed heteroscedasticity. In the case of non-normality (at α = 0.05), the presence of significant differences was checked by a Kruskal-Wallis test, followed by Dunn-Sidak post-hoc analysis (α = 0.05). This statistical procedure was applied to FCD values. All the FCD values resulted normally distributed and homoscedastic, with the exception of NADE-based methods.

Frequent Scaffolds

a



b

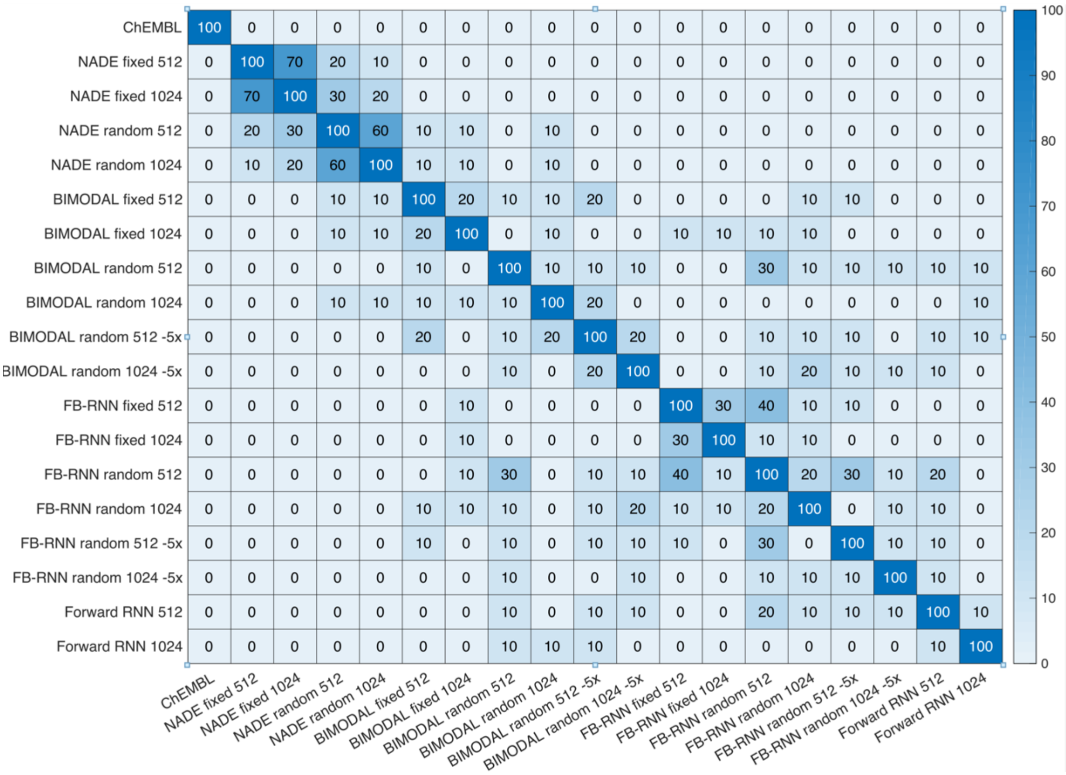


Figure S1. Overlap (% values) between the top 10 most frequently generated scaffolds by each method. (a) Top 10 scaffolds were considered. (b) Top 10 scaffolds not present in ChEMBL were considered.

Table S3. SMILES and frequency of the 10 most frequently occurring novel scaffolds generated by each method.

Scaffold ID	**Scaffold SMILES**	**Scaffold Frequency (%)**	**Scaffold SMILES**	**Scaffold Frequency (%)**
	NADE fixed 512		*NADE fixed 1024*	
Top1	C1=CC(c2ccccc2)=C1	0.14	C1=CC(c2ccccc2)=C1	0.18
Top2	C1=CCCC=C1	0.08	C1=CCCC=C1	0.10
Top3	C1=CC=C1	0.07	C1=CC=CC(c2ccccc2)=CC=C1	0.08
Top4	C1=CC=CC(OCc2ccccc2)=CC=C1	0.07	C1=CC=CC=CC=C1	0.07
Top5	C1=CC=CC(c2ccccc2)=CC=C1	0.06	C1CCC(N2CCNCC2)CC1	0.06
Top6	C1CCCNCC1	0.06	C1=CC=C1	0.05
Top7	C(#Cc1ccccc1)Cc1ccccc1	0.06	C1=CC=CC(OCc2ccccc2)=CC=C1	0.05
Top8	C(=Cc1ccccc1)Cc1ccccc1	0.05	C(=Cc1ccccc1)Cc1ccccc1	0.05
Top9	c1ccc(CCN2CCCC2)cc1	0.05	C1=CC(Oc2ccccc2)=C1	0.05
Top10	c1ccc(OCCN2CCCC2)cc1	0.05	C(#Cc1ccccc1)Cc1ccccc1	0.04
Scaffold ID	*NADE random 512*		*NADE random 1024*	
Top1	C1=CC=CC(c2ccccc2)=CC=C1	0.19	C1=CC=CC(c2ccccc2)=CC=C1	0.14
Top2	C1=CC=CC=CC=C1	0.15	C1=CC=CC=CC=C1	0.11
Top3	C1=CC=CC(Oc2ccccc2)=CC=C1	0.05	c1ccc(Nc2ccc(-c3ccccc3)cc2)cc1	0.05
Top4	C1=CC=Cc2ccccc2C=C1	0.05	C1=CC=CC(Nc2ccccc2)=CC=C1	0.04
Top5	c1ccc(Nc2ccc(-c3ccccc3)cc2)cc1	0.04	C1=CCC=C1	0.04
Top6	c1ccccc(-c2ccccc2)cccc1	0.04	c1ccccccccc1	0.04
Top7	c1ccccccccc1	0.04	C1=CC=Cc2ccccc2C=C1	0.03
Top8	C1=CC(c2ccccc2)=C1	0.03	C1=CC=c2ccccc2=CC=C1	0.03
Top9	c1ccc(Cc2ccccc2-c2ccccc2)cc1	0.03	C1CCC1	0.03
Top10	C1=C(c2ccccc2)c2ccccc21	0.03	c1ccccc(-c2ccccc2)cccc1	0.03
Scaffold ID	*BIMODAL fixed 512*		*BIMODAL fixed 1024*	
Top1	c1ccc(Nc2ccc(-c3ccccc3)cc2)cc1	0.08	O=C(NC1CCCCC1)c1ccc(-c2ncncc2C#Cc2cccnc2)cc1	0.04
Top2	O=C(Nc1ccccc1)c1c[nH]c2ccccc12	0.06	c1ccc(Nc2ccnc(Nc3ccc4c(c3)OCO4)n2)cc1	0.04
Top3	c1ccc(Nc2ccc(Oc3ccccc3)cc2)cc1	0.04	O=C(c1ccc(-c2ncncc2C#Cc2cccnc2)cc1)N1CCN(C2CCCCC2)CC1	0.03
Top4	O=C(NC1CCCCC1)c1ccc(-c2ncncc2C#Cc2cccnc2)cc1	0.04	c1ccc(Nc2ccc(-c3ccccc3)cc2)cc1	0.03
Top5	O=C(NCCc1ccccc1)C1CCNCC1	0.04	c1ccc(Nc2nc(-c3ccccc3)c3ccccc3n2)cc1	0.03
Top6	O=C(NCc1ccccc1)c1c[nH]c2ccccc12	0.04	c1ccc(Nc2nc(-c3ccccc3)cc(-c3ccccc3)n2)cc1	0.03
Top7	O=S(=O)(N=Cc1ccccc1)c1ccccc1	0.04	c1ccc(OCc2conc2-c2ccccc2)cc1	0.03
Top8	c1ccc(OC(c2ccccc2)c2ccccc2)cc1	0.04	O=C(Cc1ccccc1)N1CCN(c2ccccc2)CC1	0.03
Top9	c1cnc(Nc2ccc(N3CCNCC3)cc2)nc1	0.04	O=C(Nc1ccccc1)Nc1cccc(-c2ccccc2)n1	0.03
Top10	O=C(NCc1ccccc1)c1nc(-c2ccccc2)cs1	0.03	c1ccc(CNc2cccc(-c3ccccc3)n2)cc1	0.03
Scaffold ID	*BIMODAL random 512*		*BIMODAL random 1024*	
Top1	O=C(NCCCN1CCC(c2ccccc2)CC1)c1ccccc1	0.05	O=C(Nc1ccccc1)c1ccc(CN2CCCC2)cc1	0.04
Top2	O=C(Nc1ccccc1)Nc1ccccn1	0.04	O=C1CC(C(=O)Nc2ccc(-c3ccccc3)cc2)CN1	0.04
Top3	O=C(NCCCN1CCN(Cc2ccccc2)CC1)c1ccccc1	0.04	c1ccc(Nc2ccc(-c3ccccc3)cc2)cc1	0.04
Top4	O=C(NS(=O)(=O)c1ccccc1)c1ccccc1	0.04	O=C(C=Cc1ccccc1)NS(=O)(=O)c1ccccc1	0.04
Top5	O=C(NC1CCN(c2ccccc2)CC1)c1ccccc1	0.04	O=C(NS(=O)(=O)c1ccccc1)c1ccccc1	0.04
Top6	O=C(NCc1ccccc1)c1ccc(NC(=O)c2ccccc2)cc1	0.04	c1ccc(Cn2cc(-c3ccccc3)c(-c3ccccc3)n2)cc1	0.04
Top7	c1ccc(CNc2ccc(-c3ccccc3)cc2)cc1	0.04	O=C(NCc1ccccc1)c1cccc2ccccc12	0.03

Top8	O=C(NCCCCN1CCC(c2ccccc2)CC1)c1ccccc1	0.03	O=C(Nc1ccccc1)Nc1ccc(C(=O)Nc2ccccc2)cc1	0.03
Top9	O=C(NCCCOc1ccccc1)c1ccccc1	0.03	O=S(=O)(Nc1ccccc1)c1ccccc1-c1ccccc1	0.03
Top10	O=C(Nc1ccccc1)c1c[nH]c2ccccc12	0.03	O=S(=O)(Nc1ncns1)c1ccc(Oc2ccccc2)cc1	0.03
Scaffold ID	*BIMODAL random 512 - 5x*		*BIMODAL random 1024 - 5x*	
Top1	O=C(CCc1ccccc1)N1CCC(c2ccccc2)CC1	0.04	O=C(c1ccccc1)N1CCC(Oc2ccccc2)CC1	0.06
Top2	c1ccc(-c2nc3ccccn3c2-c2ccccc2)cc1	0.04	c1ccc(-c2ncc3ncn(C4CCCO4)c3n2)cc1	0.05
Top3	O=C1CC(C(=O)Nc2ccc(-c3ccccc3)cc2)CN1	0.04	O=C(NCCCN1CCC(c2ccccc2)CC1)c1ccccc1	0.04
Top4	c1ccc(OC(c2ccccc2)c2ccccc2)cc1	0.04	O=C(NC1CCN(Cc2ccccc2)CC1)c1ccc(-c2ccccc2)cc1	0.04
Top5	O=C(NCCCCc1ccccc1)c1ccccc1	0.03	c1ccc(-c2nc3ccccn3c2-c2ccccc2)cc1	0.04
Top6	O=C(NS(=O)(=O)c1ccccc1)c1ccccc1	0.03	O=C(Cc1ccccc1)Nc1ccc(CCCCc2nnc(NC(=O)c3ccccc3)s2)nn1	0.03
Top7	O=S(=O)(Nc1nccs1)c1ccc(Oc2ccccc2)cc1	0.03	O=C(NC1CCNCC1)c1ccc(-c2ccccc2)cc1	0.03
Top8	c1ccc(-c2ccc(OCCN3CCCCC3)cc2)cc1	0.03	O=C(c1ccc(-c2ccccc2)cc1)N1CCNCC1	0.03
Top9	c1ccc(Nc2ccc(Oc3ccccc3)cc2)cc1	0.03	O=S(=O)(Nc1nccs1)c1ccc(Oc2ccccc2)cc1	0.03
Top10	c1ccc(OCCN2CCN(c3ccccc3)CC2)cc1	0.03	c1ccc(-c2ncncn2)cc1	0.03
Scaffold ID	*FB-RNN fixed 512*		*FB-RNN fixed 1024*	
Top1	c1ccc(OC2CCN(c3ccccc3)CC2)cc1	0.10	c1ccc(OC2CCN(c3ccccc3)CC2)cc1	0.07
Top2	c1ccc(OCCCCN2CCN(c3ccccc3)CC2)cc1	0.06	c1ccc(OC2CN(c3ccc4ccccc4c3)C2)cc1	0.04
Top3	C1=CC=CCC=C1	0.05	O=C(Cc1ccccc1)N1CCN(c2ccccc2)CC1	0.03
Top4	O=C(Cc1ccccc1)N1CCN(c2ccccc2)CC1	0.03	O=C(CNC(=O)OCc1ccccc1)NC(CCc1ccccc1)Cc1ccccc1	0.03
Top5	O=C1c2ccccc2CCN1c1ccccc1	0.03	c1ccc(CC2CCc3ccccc32)cc1	0.03
Top6	c1ccc(N2CCN(CCC3CCNCC3)CC2)cc1	0.03	c1ccc(OC2CCN(c3ccccn3)CC2)cc1	0.03
Top7	c1ccc(OC2CCN(c3ccccn3)CC2)cc1	0.03	C1=CC=CC(Cc2ccccc2)C=C1	0.03
Top8	C1=CCc2ccccc2C=C1	0.03	O=C(c1ccccc1)N1CCC(N2CCC(c3ccccc3)CC2)CC1	0.03
Top9	O=C(NC(=O)c1ccccc1)c1ccccc1	0.03	O=c1n(Cc2ccccc2)c2ccccc2n1Cc1ccccc1	0.03
Top10	c1ccc(Cn2cc(-c3ccccc3)cn2)cc1	0.03	c1ccc(CN2CCN(c3ncnc4ccccc34)CC2)cc1	0.03
Scaffold ID	*FB-RNN random 512*		*FB-RNN random 1024*	
Top1	C1=CC=CCC=C1	0.12	O=C(Cc1ccccc1)N1CCN(c2ccccc2)CC1	0.06
Top2	O=C(Cc1ccccc1)N1CCN(c2ccccc2)CC1	0.06	O=C(CNC(=O)OCc1ccccc1)NC(Cc1ccccc1)C(=O)NCCc1ccccc1	0.05
Top3	c1ccc(OCCN2CCN(c3ccccc3)CC2)cc1	0.06	c1ccc(-c2cc3cccccc-3n2)cc1	0.05
Top4	C1=CCc2ccccc2C=C1	0.04	c1ccc(OC(c2ccccc2)c2ccccc2)cc1	0.05
Top5	c1ccc(CNc2ccc(-c3ccccc3)cc2)cc1	0.03	O=C(NCCCN1CCC(c2ccccc2)CC1)c1ccccc1	0.04
Top6	c1ccc(OCCCCN2CCN(c3ccccc3)CC2)cc1	0.03	O=C(Nc1ccccc1)c1ccc(OCCN2CCCCC2)cc1	0.04
Top7	O=C(NC1CCN(C(=O)c2ccccc2)CC1)c1ccccc1	0.03	c1ccc(-c2ccc3ncncc3n2)cc1	0.04
Top8	O=C(NC1CCN(c2ccccc2)CC1)c1ccccc1	0.03	c1ccc(-c2ncc3ncn(C4CCCO4)c3n2)cc1	0.04
Top9	O=C(NCCCN1CCC(c2ccccc2)CC1)c1ccccc1	0.03	c1ccc(C2CC3CCC2N3)cc1	0.04
Top10	O=C(Nc1ccc2ccccc2c1)c1ccccc1	0.03	c1ccc(Cc2cn(Cc3ccccc3)c3ccccc23)cc1	0.04
Scaffold ID	*FB-RNN random 512 - 5x*		*FB-RNN random 1024 - 5x*	
Top1	C1=COc2ccccc2C=C1	0.03	C1=COc2ccccc2C=C1	0.04
Top2	O=C(NC(c1ccccc1)c1ccccc1)c1ccccc1	0.03	O=C(NCCCN1CCC(c2ccccc2)CC1)c1ccccc1	0.04
Top3	O=C(Nc1ccccc1)c1c[nH]c2ccccc12	0.03	O=C(NCCc1ccccc1)C1CCCN1C(=O)Cc1ccccc1	0.04
Top4	O=C(NC1CCN(C(=O)c2ccccc2)CC1)c1ccccc1	0.03	O=C(Nc1ccccn1)c1ccccc1-c1ccccc1	0.04
Top5	O=C(c1ccc(-c2ccccc2)cc1)N1CCNCC1	0.03	O=C(c1ccc(-c2ccccc2)cc1)N1CCC(c2ccccc2)CC1	0.04
Top6	c1ccc(OCCN2CCN(c3ccccc3)CC2)cc1	0.03	c1ccc(-c2ccc(CN3CCCC3)cc2)cc1	0.04

Top7	c1ccc(-c2ccc3c(c2)CCCC3)cc1	0.03	c1ccc2c(c1)CCc1ccccc1-2	0.04
Top8	c1ccc(OCCCCN2CCN(c3ccccc3)CC2)cc1	0.03	C(#CC1CC2CCC(C1)N2C(c1ccccc1)c1ccccc1)c1ccccc1	0.03
Top9	O=C(CC(=O)NCCc1ccccc1)NCCc1ccccc1	0.02	C(=Cc1ccc(Cc2ccccc2)cc1)c1ccccc1	0.03
Top10	O=C(NC(Cc1ccccc1)C(=O)NCCc1ccccc1)c1ccccc1	0.02	C1=CC(c2ccccc2)C(c2ccccc2)n2nccc2N1	0.03
Scaffold ID	*Forward RNN 512*		*Forward RNN 1024*	
Top1	O=C(Nc1ccccc1)c1ccnn1-c1ccccc1	0.07	O=C(CNC(=O)c1ccccc1)NC1CCN(Cc2ccccc2)CC1	0.06
Top2	O=C(Nc1ccccc1)Nc1ccc(NC(=O)c2ccccc2)cc1	0.04	O=C(NCc1cccc[nH+]1)OCC1CO1	0.06
Top3	c1ccc(OCCN2CCN(c3ccccc3)CC2)cc1	0.04	O=C(NCCN1CCC2(CC1)C(=O)NCN2Cc1ccccc1)c1ccccc1	0.05
Top4	O=C(NCCCN1CCC(c2ccccc2)CC1)c1ccccc1	0.04	O=C1CNc2[nH]ncc2N1	0.04
Top5	O=C(Nc1ccccc1)C1CCNCC1	0.04	c1ccc(N2CCN(CCCCN3CCN(c4ccccc4)CC3)CC2)cc1	0.04
Top6	c1ccc(Nc2ncnc3cc(-c4ccccc4)ccc23)cc1	0.04	O=C(NS(=O)(=O)c1ccccc1)c1ccccc1	0.03
Top7	C=C1CCC(=O)O1	0.03	O=C(Nc1ccc(N2CCCCC2)cc1)c1ccccc1	0.03
Top8	O=C(CNC(=O)c1ccccc1)NC1CCN(Cc2ccccc2)CC1	0.03	O=C(Nc1ccccc1)C1CCCCN1S(=O)(=O)c1ccccc1	0.03
Top9	O=C(NC(CCNCCCc1ccccc1)Cc1ccccc1)c1ccccc1	0.03	O=C1CCc2ccccc2N1CCCCN1CCN(c2ccccc2)CC1	0.03
Top10	O=C(NCc1ccccc1)c1csc(-c2ccccc2)n1	0.03	O=C1COc2[nH]ncc2N1	0.03

Measured runtime

Table S4. Measured runtime for pre-training and sampling 1000 SMILES (*mean±std.dev*), on a NVIDIA GeForce GTX 1080 Ti - 256 GB (settings reported in Tables S1 and S2, 512 hidden units in total).

Model	**Starting point**	**Time** [*min*] Pre-training	Sampling
Forward	fixed	4.51±0.06	0.79±0.01
BIMODAL	fixed	28.72±0.17	3.94±0.02
FB-RNN	fixed	3.13±0.09	0.50±0.01
NADE	fixed	220±7	46.0±0.4
BIMODAL	random	104.72±0.05	12.4±0.3
FB-RNN	random	6.34±0.02	1.00±0.01
NADE	random	10.4±0.1	48.2±0.3