

PaccMann^{RL}: Designing anticancer drugs from transcriptomic data via reinforcement learning

Jannis Born^{*1[0000-0001-8307-5670]}, Matteo Manica^{*1[0000-0002-8872-0269]}, Ali Oskooei^{*1[0000-0002-8318-687X]}, Joris Cadow^{1[0000-0002-4410-2805]}, and María Rodríguez Martínez^{1[0000-0003-3766-4233]}

IBM Research Zurich, Switzerland
`{jab,tte,osk,dow,mrm}@zurich.ibm.com`

^{*} Equal contributions

Abstract. With the advent of deep generative models in computational chemistry, *in silico* anticancer drug design has undergone an unprecedented transformation. While state-of-the-art deep learning approaches have shown potential in generating compounds with desired chemical properties, they disregard the genetic profile and properties of the target disease. Here, we introduce the first generative model capable of tailoring anticancer compounds for a specific biomolecular profile. Using a RL framework, the transcriptomic profiles of cancer cells are used as a context for the generation of candidate molecules. Our molecule generator combines two separately pretrained variational autoencoders (VAEs) – the first VAE encodes transcriptomic profiles into a smooth, latent space which in turn is used to condition a second VAE to generate novel molecular structures on the given transcriptomic profile. The generative process is optimized through PaccMann, a previously developed drug sensitivity prediction model, to obtain effective anticancer compounds for the given context (i.e., transcriptomic profile).

We demonstrate how the molecule generation can be biased towards compounds with high predicted inhibitory effect against individual cell lines or specific cancer sites. We verify our approach by investigating candidate drugs generated against specific cancer types and find the highest structural similarity to existing compounds with known efficacy against these cancer types. We envision our approach to transform *in silico* anticancer drug design by leveraging the biomolecular characteristics of the disease in order to increase success rates in lead compound discovery.

1 Introduction

The last two decades have seen a decline in the productivity of the drug discovery pipeline while the investment into drug discovery has risen significantly [29]. Indeed, only a minimal portion of drug candidates obtain market approval (less than 0.01%), with an estimated 10-15 years until market release and costs that range between one [29] to three billion dollars per drug [30]. This low efficiency has been attributed to the high dropout rate of candidate molecules in the early

stages of the pipeline, highlighting the need for more accurate in silico and in vitro models that produce more successful candidate drugs. Most recently, deep learning methods have gained popularity within the computational chemistry community [6] and a number of works have demonstrated the feasibility of in silico design of novel candidate compounds with desired chemical properties [26,12,40]. In all of these models, the generative process is controlled via a structurally driven evaluator (or critic) that biases the generation of a chemical to satisfy the required chemical structural properties. While very effective in generating compounds with desired chemical properties, these methods do not integrate information about the cellular environment in which the drug is intended to act. However, the two main causes of the increasing attrition rate in drug design are lacking efficacy against the specific disease of interest and off-target cytotoxicity [34], calling to bridge systems biology closer with drug discovery. In addition to the initial wet-lab validations, the discovery pipeline involves a sequential process that builds upon high-throughput screenings, ADMET-assessments (absorption, distribution, metabolism, excretion and toxicity, i.e., criteria for the pharmacological activity of a compound) and a lengthy phase of clinical trials. The costs of the experimental and clinical phase can be prohibitive and any solution that helps to reduce the number of required experimental assays can provide a competitive advantage and reduce time to market. To this end, we present a deep RL based model for anticancer molecule generation that builds on top of the previous approaches and, for the first time, enables generation of novel anticancer compounds while taking into account the disease context encoded in the form of gene expression profile (GEP) of the tumor cell.

The presented framework consists of a conditional molecule generator (embodied by two separate VAEs) and a critic module that evaluates the efficacy of proposed compounds on the target profile (see Figure 1). The training procedure splits into two stages. In the first stage, the models are trained independently; specifically one VAE is trained on gene expression data from TCGA [36] (see Figure 1A), another VAE is trained on bioactive small molecules from ChEMBL [2] (see Figure 1B) and a multimodal drug sensitivity prediction model is fetched from previous work [20]. In the second stage, the encoder of the profile VAE is combined with the decoder of the molecule VAE (see Figure 1C) and exposed to a joint retraining that is optimized in a policy gradient regime with a reward coming from the critic module (see Figure 1D and E). The goal of the optimization is to tune the generative model such that it generates (novel) compounds that have maximal efficacy against a given biomolecular profile; be it the characteristic for a cancer site, a patient subgroup or even an individual. By *efficacy*, we refer to cellular IC₅₀ (i.e. the micromolar concentration necessary to inhibit 50% of the cells in a sample) as opposed to e.g. enzymatic IC₅₀. It is important to note that this efficacy is a joint property of a drug-cell-pair and empirically it is well-known that the efficacy of a compound heavily varies for different types of cells. In this work, we emphasize profile-specific compound generation and optimize the generator using IC₅₀ as the sole critic.

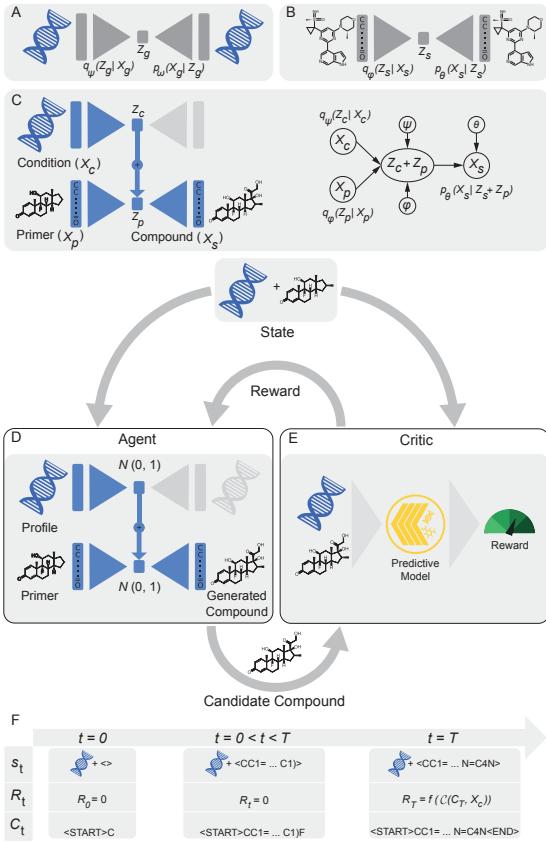


Fig. 1: The proposed framework for anticancer compound design against specific cancer profiles. A biomolecular profile VAE (PVAE, shown in A) and a sequential compound generator VAE (SVAE, B) are combined to obtain a conditional molecule generator (C). Each of the PVAE, SVAE and the predictive critic are pretrained independently. Thereafter, the conditional generation process starts with a biomolecular profile of interest e.g., transcriptomic profile from an individual patient. The given profile is encoded into the latent space of gene expression profiles and is then decoded through the molecular decoder to produce a candidate compound (D). This generative process can optionally be “primed” through encoding a known, effective compound or a functional group with the molecular encoder. Molecules are generated directly as SMILES sequences which are assembled in a sequential process, one atom at a time. A full cycle of this process, including the state (s_t), the reward (R_t) and the candidate compound (C_t) are shown in (F). The proposed compound is then evaluated through the critic, a multimodal drug sensitivity prediction model that ingests the compound and the target profile of interest (E). The RL based optimization is conducted by maximizing the reward given by the critic. Over the course of training, the generator will thus learn to produce candidate compounds with higher and higher efficacy.

2 Methods

[Figure 1](#) shows the two main components of our proposed end-to-end architecture: the conditional generator ([Figure 1D](#)) and the critic ([Figure 1E](#)).

Conditional generator (\mathcal{G}). This is a molecule generator that produces a candidate drug structure using its SMILES string representation [35]. SMILES sequences are preferable over (functional) fingerprint-based representations of molecules (e.g., ECFP [27]) since they have shown to be superior in both predictive [15,20] and generative models for molecules [4]. In our use case, the generative process needs to be conditioned on a target biomolecular profile, e.g., from a patient or a disease. Inspired by [12], we concluded that VAEs are the ideal model for our task since by design they bring about a structurally ordered latent space that simplifies the combination of different information sources. Our conditional generator combines two VAEs that are trained independently prior to being fused together: 1) a denoising VAE for cancer profile encoding/generation (called PVAE) and 2) a sequential VAE (SVAE) for SMILES sequence generation. The detailed derivation and equations for the VAE can be found in [18,31]. PVAE is pretrained on gene expression profiles (GEP) to learn a consistent latent representation for biomolecular signatures. SVAE is pretrained on bioactive drug-like molecules to learn the syntax of valid SMILES and general molecular semantics. The fact that models that process SMILES sequences must have the ability to *count* the ring opening and closing symbols in a molecule necessitates the use of stack memory [14], in our case implemented through stack-augmented GRUs as proposed by [16] (the details of the SVAE architecture can be found in the [appendix](#)). Thereafter, the encoder of the PVAE is fused with the decoder of the SVAE via their latent space. The combination of the two models enables to learn a latent space that links biomolecular profiles and chemical structures providing an effective way to sample novel compounds given a specific GEP. In the final training phase, the weights of the fused model are fine-tuned using a reward from the critic in a RL framework.

Critic (\mathcal{C}). The critic is a multimodal drug sensitivity prediction model which evaluates the efficacy of any given candidate compound against a biomolecular profile of interest (e.g., gene expression of a cancer cell line). It outputs a non-negative reward, a function of the predicted IC₅₀ of the candidate compound for the target profile, which is used in a RL framework to update the conditional generator. Following the most recent advances for multimodal drug sensitivity prediction we herein utilize PaccMann as a critic; specifically convolutional attention encoders as proposed in [20].

The RL framework. The conditional generator is retrained in combination with the critic in a RL-based optimization process to tailor molecules towards the given GEP. First, the GEP is encoded into its latent space, Z_c (see [Figure 1C](#)). This embedding is then added to the latent encoding of a primer compound or substructure (Z_p). The advantage of using a primer is that it enables injection

of prior knowledge into the model by starting the generative process from an existing and proven effective compound or functional group – instead of designing a compound from scratch. Formally, the molecule generation is conditioned on a context \mathcal{Z} , where in this work $\mathcal{Z} = \{Z_c, Z_p\}$. Since Z_c and Z_p reflect embeddings learned from semantically different data sources (gene expression and molecules) it is non-trivial to combine them meaningfully. We use a summation because it is a permutation invariant operation and has been proposed in the deep sets architecture [41] to combine a variable set of unstructured latent encodings. Our additive latent representation is similar in concept to the conditional VAE with additive Gaussian encoding space [33]. Intuitively, this fusion presumably warps the latent space from encoding structural similarity (of molecules or GEP) into functional similarity [12] so as to aggregate molecules with similar predicted efficacy for a given cell line. Note that using a primer compound or substructure is optional and if no priming compound is used, simply the latent space representation of the <START> token is added to the latent encoding of the target GEP.

Next, the conditional generator decodes the latent encoding, $Z_c + Z_p$, and generates a molecular structure that, in combination with the GEP, is fed to the critic to produce a certain reward for the generated compound, as illustrated in Figure 1D and E. Following the notation of [26], the conditional generator, \mathcal{G} , acts as the *agent* and the multimodal IC50 prediction model, \mathcal{C} , represents the *critic*. The weights of \mathcal{C} are fixed. We aim to optimize Θ , the parameters of \mathcal{G} , to produce candidate compounds, C_T , that target a specific GEP, X_c . In contrast to [26], we define the set of states S as all possible SMILES strings (with length $\leq T$) paired with the target GEP. The set of possible actions a that \mathcal{G} can take is a set \mathcal{A} , that is a vocabulary of all characters and symbols of the canonical SMILES language. As depicted in Figure 1F, molecules are generated by \mathcal{G} by sampling an action a_t at each step ($0 < t < T$) from $p(a_t|s_{t-1})$, where $s_{t-1} = (C_{t-1}, X_c)$. Terminal states $S^* \subset S$ are reached when either $t = T$ or when the terminal action $a_T = \langle\text{END}\rangle$ has been sampled. \mathcal{G} is trained to learn a policy, $\Pi(\Theta)$, by maximizing:

$$\Pi(\Theta) = \sum_{s_T \in S^*} p_\Theta(s_T) R(s_T) \quad (1)$$

where the state $s_T = (C_T, X_c)$ is a tuple of the candidate compound C_T and the cell profile X_c and the reward $R(s_T) = f(\mathcal{C}(C_T, X_c))$ is the output of the critic \mathcal{C} scaled by a reward function f . In our experiments, all intermediate rewards $R(s_t) = 0$ where $t < T$, the sum is approximated using policy gradients, specifically the REINFORCE algorithm [37] and the reward function f for determining the reward from the IC50 prediction, $\mathcal{C}(C_T, X_c)$, is computed by $f(\text{IC50}) = \exp\left(\frac{-\text{IC50}}{\alpha}\right)$ where $\alpha = 5$ in this work (see details in the [appendix](#)).

Data. For the PVAE, we employed a training dataset of 11,592 (normalized) RNA-Seq GEPs from healthy and cancerous human tissue from the TCGA database and validated it on 1,289 samples from the same database [36]. The

number of genes was reduced to the same 2,128 genes as used in [20,21], following the network propagation procedure described in [22]. The SVAE was pretrained on the SMILES representation of 1,576,904 compounds (10% were held out for performance validation) from the ChEMBL database [10]. For RL optimization of \mathcal{G} , we used GEPs publicly available from GDSC [39] and CCLE [1] databases. Since the RNA-Seq of these cancer cell line databases were passed through the PVAE (pretrained on human samples from TCGA [36]), we compared the standardized gene expression distributions for the selected genes across these databases and found them to be comparable (see [appendix](#)). Others also reported good agreement between transcriptomics data in CCLE and TCGA [11]. Drug sensitivity data (i.e., IC₅₀) from GDSC and CCLE were used to train the IC₅₀ prediction model, the critic (\mathcal{C}). The hyperparameter and details on the utilized hardware and software can be found in the [appendix](#).

3 Results

3.1 PVAE generation of gene expression profiles

The pretraining results of the PVAE are shown in [Figure 2A](#), B and C. As shown in [Figure 2B](#), the reconstructed samples (blue plot) as well as the generated GEPs (green plot) correctly mimic the distribution of the original GEPs (red plot). Furthermore, the sampled GEPs follow the same lognormal distribution as the original data. [Figure 2C](#) shows that the generated GEPs exhibit a higher similarity to the testing than to the training sample. Overall, these results suggest that the pretrained PVAE is able to generate new realistic GEPs of human cells.

3.2 SVAE generation of molecular structures

[Figure 2D](#), E and F give a quantitative analysis of the SVAE results following pretraining for 10 epochs with \sim 1.4 million structures from ChEMBL. For a qualitative analysis, we provide a sample of molecules in the [appendix](#). Since the model had a tendency for posterior collapse (relying predominantly on the decoder [23]), we explored various levels of token dropout for the decoder of SVAE [5]. To investigate the novelty and diversity of the generated molecules, we sampled 10,000 molecules by decoding random points from the latent space and utilized a well-established chemical structure similarity measure, the Tanimoto similarity [32] to compare the ECFP [27] of a subset of 1000 generated molecules with the training and test data from ChEMBL. [Figure 2F](#) presents the distributions of the highest Tanimoto similarity between each generated compound and all compounds in training and test dataset respectively. Only a negligible fraction of the generated molecules existed in either of the datasets, whereas the vast majority had a Tanimoto similarity (τ) between 0.2 and 0.6 suggesting that our model learned to propose novel molecular structures from the chemical space of about 10^{30} to 10^{60} molecules [24]. Overall, 96.2% of the 10,000 generated molecules were valid molecular structures (validity assessed via RDKit) surpassing the state of the art results reported by [26] who used the same stack-augmented GRUs trained on

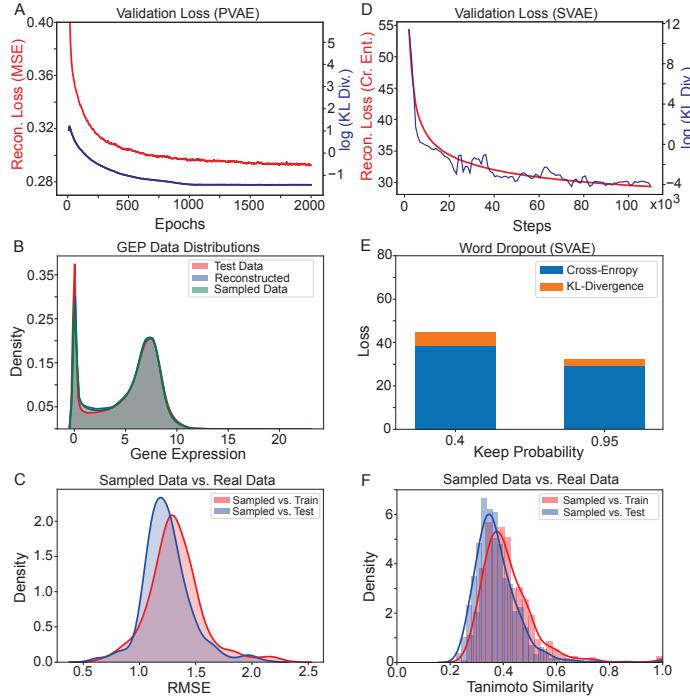


Fig. 2: Results of pretrained PVAE and SVAE models. PVAE: (A) Development of validation error over the course of training. Reconstruction loss (MSE) and KL divergence are shown separately for comparison. (B) Distribution of gene expression values in real, reconstructed and generated samples. (C) Sampled (i.e., generated) data from the latent space of PVAE compared against training and test datasets from TCGA. SVAE: (D) Development of validation error over the course of training. Cross-entropy between target and generated SMILES is shown separately from the KL divergence (log scale for visual clarity). One epoch corresponds to $\sim 11\,000$ training steps. (E) The two losses of our baseline model (keep probability 95%) is compared with a model using higher a word dropout of 60% (KL loss is scaled by a factor 200 for visual clarity). Higher word dropout in the decoder increases the encoder loss, i.e., results in storing more information in the latent space. (F) The Tanimoto similarity between the Morgan fingerprints (ECFP) of the generated molecules and the structures from ChEMBL train and test datasets is used to verify that the generated compounds are sufficiently different from the training data.

the ChEMBL database (95% SMILES validity). In addition, 99.8% of the valid generated molecules were unique across the 10,000 generations.

3.3 Disease-specific compound generation

Herein, we present the results of our generator conditioned on gene expression profiles of cancer subtypes. Four different models were trained, one for each of the cancer sites: breast, lung, prostate and neuroblastoma.

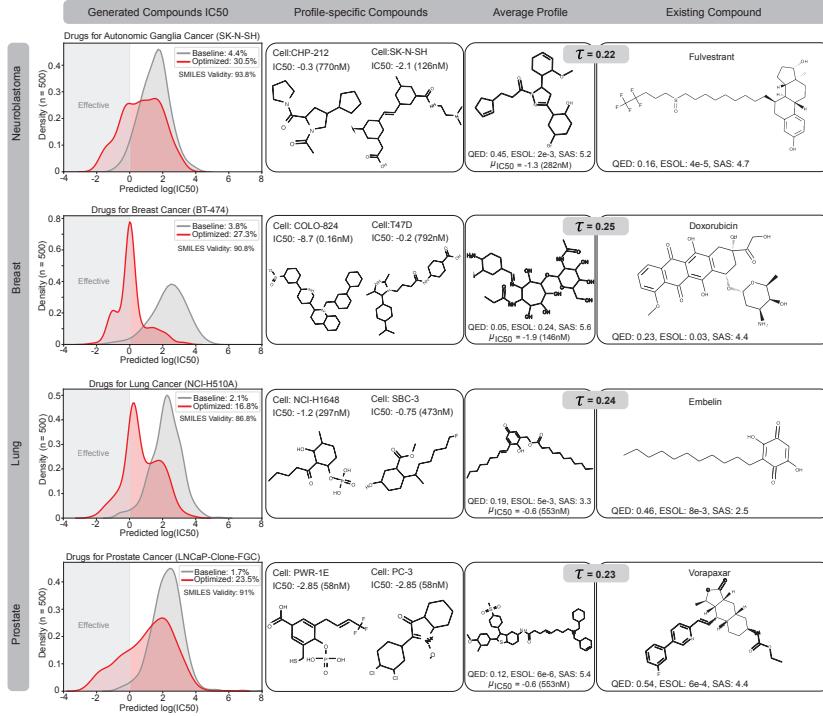


Fig. 3: Sample results for profile-driven model optimization and anticancer compound generation. Each row illustrates the results of training the RL pipeline on cell lines from a specific site: neuroblastoma, breast, lung and prostate cancer. The first column compares the distributions of IC50 predictions given by the critic model for a set of $n=500$ drug candidates generated with RL optimization and without RL optimization. The second column presents candidate compounds with a high predicted efficacy (low IC50) against a particular cell line that was not seen during training. The third column showcases generated compounds that were optimized to be effective against the average cell-line profiles of the given cancer type in each row. In the fourth column, we present an *existing* anticancer compound (approved against at least one type of cancer), that was in the top-3 neighborhood of the generated compound in the third column. The existing and generated compounds are compared in terms of Tanimoto structural similarity as well as three chemical scores crucial in drug design namely, druglikeness (QED, 0 worst, 1 best), synthesizability (SAS, 1 best, 10 worst) and solubility (ESOL, given in M/L).

For the evaluation of the four models, all generated compounds with a predicted IC50 value below $1\mu\text{M}$ were considered as *effective*. Moreover, within each cancer site 80% of the cell lines were considered as training cell lines and used to optimize the parameters Θ of the conditional generator whereas the remaining 20% were set aside for testing. Our model learned to produce compounds with lower IC50 values, for unseen cell lines from the given cancer site. The first

column of [Figure 3](#) shows that the IC50 distribution of candidate compounds proposed by the generative model were successfully shifted towards higher efficacy. The baseline model corresponds to the pretrained SVAE from which $n = 500$ molecules were randomly sampled. In all four cases, a significant portion (between 17% and 30%) of molecules generated from the optimized model were assigned a IC50 value below $1\mu\text{M}$, whereas only 1-4% of the candidates generated by the baseline model (i.e., the SVAE, a generative model of drug-like molecules that was not yet optimized via RL retraining) were classified as effective. The second column of [Figure 3](#) shows generated molecules that are predicted as being effective against an unseen cell line from the respective cancer site. In the third column of [Figure 3](#), we showcase novel molecules that were designed specifically for each cancer site, as opposed to a specific cancer cell profile. All compounds exhibited high predicted efficacy against the average cellular profile of the target site while maintaining efficacy against the majority of individual cell lines for that site. In the last column of [Figure 3](#), we compare the four site-specific candidate compounds with one of their top-3 neighbors (Tanimoto similarity, τ) from several hundreds of existing anticancer compounds. The third closest neighbor of the generated compound against neuroblastoma ([Figure 3](#) first row, third column) is Fulvestrant, an antagonist/modulator of ER α which has recently been proposed as a novel anticancer agent for neuroblastoma [13]. The candidate compound proposed against breast cancer ([Figure 3](#) second row, third column), has Doxorubicin as one of the top-3 nearest neighbors. Doxorubicin is a commonly used chemotherapeutic against breast cancer [19]. The generated compound against lung cancer ([Figure 3](#), third row, third column) results close to Embelin, an existing anticancer compound from the GDSC database.

Embelin is known to be a promising anticancer compound as it is the only known non-peptide inhibitor of the XIAP protein [25], a protein that plays an important role in lung cancer development [7]. Lastly, the closest neighbor of the prostate-specific generated compound ([Figure 3](#) fourth row, third column) is Vorapaxar. Its efficacy is highest against a prostate cancer cell line (DU-145) according to GDSC/CCLE. Vorapaxar is an antagonist of a protease-activated receptor (PAR-1) that is known to be overexpressed in various types of cancer, including prostate [42].

4 Discussion

We herein presented the first machine-learning based anti-cancer compound generator that enables us to condition the molecular generation on the biomolecular profile (specifically we explored transcriptomic profiles) of the target cell or cancer site. We demonstrated, using a RL optimization framework, that our proposed generative model could be optimized to produce candidate compounds with high predicted efficacy (IC50) against a given target profile. We showcased in a post-hoc analysis that each of the four site-specific generated compounds had structural similarities to known anticancer compounds commonly used to treat cancer of the same type as the generated compound was optimized for.

Oftentimes however, medical chemists do not start the drug design from scratch, but from an approved drug and with the goal to find a drug with similar effects (e.g. increased efficacy or reduced side effects). Our framework neatly grants the option to incorporate this prior knowledge into the design process already. Another endeavour is to develop a drug that specifically targets a protein (e.g. one that has been implicated in tumor proliferation or treatment response according to a gene-knockout study). Whilst [43] very recently presented a model that proposed potent DDR1 kinase inhibitors, we are working towards a generic framework where the molecule generation can be conditioned on possibly multimodal context information such as a target protein, a primed drug, a tumor profile and notably also a combination thereof by utilising permutation invariant operations in the multimodal latent space [41]. We have, however, not yet fully explored the full potential of the framework, e.g. by conditioning the design on a known drug in conjunction with a cell profile. While we believe our results to be a promising stepping stone for profile-specific anticancer compound generation, we are aware further optimization must be done before it can be used a reliable tool for drug discovery. For instance, there are various other properties of a candidate drug other than its efficacy that determine its potential for becoming a successful anticancer compound, for example water solubility, drug-likeness, synthesizability and cytotoxicity. Our RL optimization framework can easily be extended with further critics that reward or punish the conditional generator according to the mentioned properties. For example, water solubility (ESOL [8]), drug-likeness (QED [3]) and synthesizability (SAS [9]) can all be estimated from the raw SMILES sequences. In the [appendix](#) we compare these properties from approved drugs in GDSC and CCLE databases to our generated compounds and find that we do not yet mimick. To that end, we aim to amend the reward function in the future so as to incorporate rewards not only for efficacy but also based on other drug-relevant chemical properties.

5 Availability of software and materials

The omics data used to pretrain the PVAE, the molecular data for the SVAE and the cell profiles used in the RL regime as well as the pretrained models can be found on <https://ibm.box.com/v/paccmann-pytoda-data>. To assess the critic, please see [20]. All code to reproduce the experiments is publicly available on <https://github.com/PaccMann/>. For a detailed example see https://github.com/PaccMann/paccmann_rl.

Acknowledgements

The project leading to this publication has received funding from the European Union’s Horizon 2020 research and innovation programme under grant agreement No 826121.

References

1. Barretina, J., Caponigro, G., Stransky, N., Venkatesan, K., et al.: The cancer cell line encyclopedia enables predictive modelling of anticancer drug sensitivity. *Nature* **483**(7391), 603 (2012)
2. Bento, A.P., Gaulton, A., Hersey, A., Bellis, L.J., et al.: The ChEMBL bioactivity database: an update. *Nucleic Acids Research* **42**(D1), D1083–D1090 (11 2013). <https://doi.org/10.1093/nar/gkt1031>, <https://doi.org/10.1093/nar/gkt1031>
3. Bickerton, G.R., Paolini, G.V., Besnard, J., Muresan, S., Hopkins, A.L.: Quantifying the chemical beauty of drugs. *Nature chemistry* **4**(2), 90 (2012)
4. Bjerrum, E., Sattarov, B.: Improving chemical autoencoder latent space and molecular de novo generation diversity with heteroencoders. *Biomolecules* **8**(4), 131 (2018)
5. Bowman, S.R., Vilnis, L., Vinyals, O., Dai, A.M., Jozefowicz, R., Bengio, S.: Generating sentences from a continuous space. arXiv preprint arXiv:1511.06349 (2015)
6. Chen, H., Engkvist, O., Wang, Y., Olivecrona, M., Blaschke, T.: The rise of deep learning in drug discovery. *Drug discovery today* (2018)
7. Cheng, Y.J., Jiang, H.S., Hsu, S.L., Lin, L.C., Wu, C.L., Ghanta, V.K., Hsueh, C.M.: Xiap-mediated protection of h460 lung cancer cells against cisplatin. *European journal of pharmacology* **627**(1-3), 75–84 (2010)
8. Delaney, J.S.: Esol: Estimating aqueous solubility directly from molecular structure. *Journal of Chemical Information and Computer Sciences* **44**(3), 1000–1005 (2004). <https://doi.org/10.1021/ci034243x>
9. Ertl, P., Schuffenhauer, A.: Estimation of synthetic accessibility score of drug-like molecules based on molecular complexity and fragment contributions. *Journal of cheminformatics* **1**(1), 8 (2009)
10. Gaulton, A., Hersey, A., Nowotka, M., Bento, A.P., Chambers, J., Mendez, D., Mutowo, P., Atkinson, F., Bellis, L.J., Cibrián-Uhalte, E., et al.: The chembl database in 2017. *Nucleic acids research* **45**(D1), D945–D954 (2016)
11. Ghandi, M., Huang, F.W., Jané-Valbuena, J., Kryukov, G.V., Lo, C.C., McDonald, E.R., Barretina, J., Gelfand, E.T., Bielski, C.M., Li, H., et al.: Next-generation characterization of the cancer cell line encyclopedia. *Nature* **569**(7757), 503 (2019)
12. Gomez-Bombarelli, R., Wei, J.N., Duvenaud, D., Hernandez-Lobato, J.M., et al.: Automatic chemical design using a data-driven continuous representation of molecules. *ACS central science* **4**(2), 268–276 (2018)
13. Gorska, M., Kuban-Jankowska, A., Milczarek, R., Wozniak, M.: Nitro-oxidative stress is involved in anticancer activity of 17 β -estradiol derivative in neuroblastoma cells. *Anticancer research* **36**, 1693–8 (04 2016)
14. Hopcroft, J.E., Ullman, J.D.: Formal Languages and Their Relation to Automata. Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA (1969)
15. Jastrzebski, S., Leśniak, D., Czarnecki, W.M.: Learning to smile (s). arXiv preprint arXiv:1602.06289 (2016)
16. Joulin, A., Mikolov, T.: Inferring algorithmic patterns with stack-augmented recurrent nets. In: Advances in neural information processing systems. pp. 190–198 (2015)
17. Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980 (2014)
18. Kingma, D.P., Welling, M.: Auto-encoding variational bayes. arXiv preprint arXiv:1312.6114 (2013)

19. Lao, J., Madani, J., Puértolas, T., Álvarez, M., Hernández, A., Pazo-Cid, R., Artal, Á., Antón Torres, A.: Liposomal doxorubicin in the treatment of breast cancer patients: a review. *Journal of drug delivery* **2013** (2013)
20. Manica, M., Oskooei, A., Born, J., Subramanian, V., Saez-Rodriguez, J., Rodriguez Martinez, M.: Toward explainable anticancer compound sensitivity prediction via multimodal attention-based convolutional encoders. *Molecular Pharmaceutics* (2019). <https://doi.org/10.1021/acs.molpharmaceut.9b00520>, pMID: 31618586
21. Oskooei, A., Born, J., Manica, M., Subramanian, V., et al.: Paccmann: Prediction of anticancer compound sensitivity with multi-modal attention-based neural networks. arXiv preprint arXiv:1811.06802 (presented at NeurIPS 2018 workshop: “Workshop for Machine Learning for Molecules and Materials”) (2018)
22. Oskooei, A., Manica, M., Mathis, R., Martínez, M.R.: Network-based biased tree ensembles (NetBiTE) for drug sensitivity prediction and drug sensitivity biomarker identification in cancer. arXiv preprint arXiv:1808.06603 (presented at ICML 2019 workshop: “Workshop on Computational Biology”) (2018)
23. Pagnoni, A., Liu, K., Li, S.: Conditional variational autoencoder for neural machine translation. arXiv preprint arXiv:1812.04405 (2018)
24. Polishchuk, P.G., Madzhidov, T.I., Varnek, A.: Estimation of the size of drug-like chemical space based on gdb-17 data. *Journal of computer-aided molecular design* **27**(8), 675–679 (2013)
25. Poojari, R.: Embelin—a drug of antiquity: shifting the paradigm towards modern medicine. *Expert opinion on investigational drugs* **23**(3), 427–444 (2014)
26. Popova, M., Isayev, O., Tropsha, A.: Deep reinforcement learning for de novo drug design. *Science advances* **4**(7), eaap7885 (2018)
27. Rogers, D., Hahn, M.: Extended-Connectivity Fingerprints. *Journal of Chemical Information and Modeling* **50**(5), 742–754 (May 2010). <https://doi.org/10.1021/ci100050t>, <https://doi.org/10.1021/ci100050t>
28. Savjani, K.T., Gajjar, A.K., Savjani, J.K.: Drug solubility: importance and enhancement techniques. *ISRN pharmaceutics* **2012** (2012)
29. Scannell, J.W., Blanckley, A., Boldon, H., Warrington, B.: Diagnosing the decline in pharmaceutical r&d efficiency. *Nature reviews Drug discovery* **11**(3), 191 (2012)
30. Schneider, G.: Mind and machine in drug design. *Nature Machine Intelligence* p. 1 (2019)
31. Sohn, K., Lee, H., Yan, X.: Learning structured output representation using deep conditional generative models. In: *Advances in neural information processing systems*. pp. 3483–3491 (2015)
32. Tanimoto, T.T.: Elementary mathematical theory of classification and prediction. IBM Internal Report (1958)
33. Wang, L., Schwing, A., Lazebnik, S.: Diverse and accurate image description using a variational auto-encoder with an additive gaussian encoding space. In: *Advances in Neural Information Processing Systems*. pp. 5756–5766 (2017)
34. Wehling, M.: Assessing the translatability of drug projects: What needs to be scored to predict success? *Nature reviews. Drug discovery* **8**, 541–6 (07 2009). <https://doi.org/10.1038/nrd2898>
35. Weininger, D.: Smiles, a chemical language and information system. 1. introduction to methodology and encoding rules. *Journal of chemical information and computer sciences* **28**(1), 31–36 (1988)
36. Weinstein, J.N., Collisson, E.A., Mills, G.B., Shaw, K.R.M., Ozenberger, B.A., Ellrott, K., Shmulevich, I., Sander, C., Stuart, J.M., Network, C.G.A.R., et al.: The cancer genome atlas pan-cancer analysis project. *Nature genetics* **45**(10), 1113 (2013)

37. Williams, R.J.: Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Machine learning* **8**(3-4), 229–256 (1992)
38. Williams, R.J., Zipser, D.: A learning algorithm for continually running fully recurrent neural networks. *Neural computation* **1**(2), 270–280 (1989)
39. Yang, W., Soares, J., Greninger, P., Edelman, et al.: Genomics of drug sensitivity in cancer (gdsc): a resource for therapeutic biomarker discovery in cancer cells. *Nucleic acids research* **41**(D1), D955–D961 (2012)
40. You, J., Liu, B., Ying, Z., Pande, V., Leskovec, J.: Graph convolutional policy network for goal-directed molecular graph generation pp. 6412–6422 (2018)
41. Zaheer, M., Kottur, S., Ravanbakhsh, S., Poczos, B., Salakhutdinov, R.R., Smola, A.J.: Deep sets. In: Advances in neural information processing systems. pp. 3391–3401 (2017)
42. Zhang, X., Wang, W., True, L.D., Vessella, R.L., Takayama, T.K.: Protease-activated receptor-1 is upregulated in reactive stroma of primary prostate cancer and bone metastasis. *The Prostate* **69**(7), 727–736 (2009). <https://doi.org/10.1002/pros.20920>
43. Zhavoronkov, A., Ivanenkov, Y.A., Aliper, A., Veselov, M.S., Aladinskiy, V.A., Aladinskaya, A.V., Terentiev, V.A., Polykovskiy, D.A., Kuznetsov, M.D., Asadulaev, A., et al.: Deep learning enables rapid identification of potent ddr1 kinase inhibitors. *Nature biotechnology* **37**(9), 1038–1040 (2019)

PaccMann^{RL}: Designing anticancer drugs from transcriptomic data via reinforcement learning - Appendix

5.1 SVAE architecture with StackGRU

To enable neural networks to count, [16] introduced stack-augmented RNN. Stack-RNNs complement RNNs with a differentiable push-down stack operated through learnable controllers, op_t at step t , that involve three operations: PUSH, POP and NO-OP (see Figure 4).

$$op_t = \mathbf{s}(W_{op} h_t), \quad (2)$$

where h_t is the hidden state, W_{op} is a $3 \times H$ matrix (H being the dimension of hidden state) and \mathbf{s} is the softmax function. At each time step the controller probabilities are determined from Equation 2 and the stack memory is updated using the learned controller via a multiplicative gating mechanism:

$$\begin{cases} S_t[0] &= op_t[\text{PUSH}] \mathbf{s}(W_{so} h_t) + op_t[\text{POP}] S_{t-1}[1] + \\ &\quad op_t[\text{NO-OP}] S_{t-1}[0] \\ h_t &= \mathbf{s}(W_i X_t + W_R h_{t-1} + W_{si} S_{t-1}) \end{cases} \quad (3)$$

where S_t is the stack, W_{so} is a $1 \times H$ matrix and W_{si} is a $H \times N$ matrix (N being the stack height). W_i is the input matrix applied to the sequence and W_R is the recurrent matrix. It should be noted that for the sake of brevity, we only show the update equation for the topmost element of the stack in Equation 3.

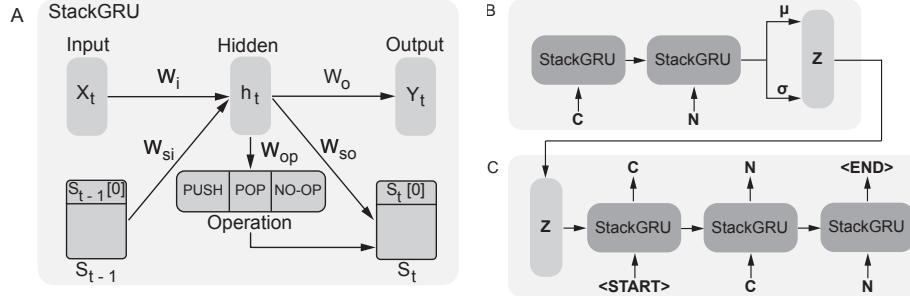
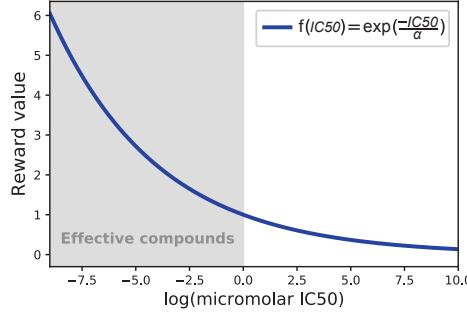


Fig. 4: (A) The StackGRU architecture adopted in the SVAE. The stack-augmented GRU (StackGRU) architecture complements a regular GRU with a stack that allows one out of three possible operations at each time-step: PUSH, POP and NO-OP. The operation vector is determined through a softmax from the hidden state of each time step. (B) and (C) are encoder and decoder of the SVAE architecture. (B) encodes the SMILES sequences into multivariate Gaussians with parameters μ and σ . (C) The decoder StackGRU units reconstruct the SMILES sequence from a latent representation (Z_p) sampled from the multivariate Gaussian.

5.2 Reward function

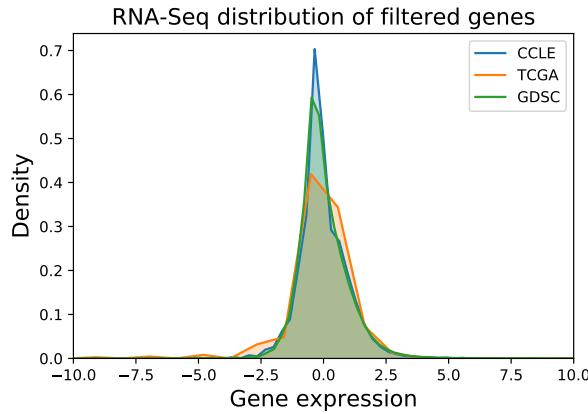
The reward function $f(x) = \exp\left(\frac{-x}{\alpha}\right)$ was used to map the logarithmic micromolar IC₅₀ values predicted by the critic to a reward that was subject to maximization in our adopted RL framework (see [Figure 5](#)).



[Fig. 5](#): Reward function to map the predicted IC₅₀ of the critic (PaccMann) to a reward being fed to the conditional generator. To produce the plot, α was set to 5.

5.3 Gene expression in human samples and cancer cell lines

Comparing the standardized gene expression values of GDSC [39] and CCLE [1] with the one from human samples from TCGA [36] reveals a similarity ([Figure 6](#)). This justifies our choice of utilizing the encoder of the PVAE for cell line data during the RL regime, although it was initially pretrained on human samples from TCGA.



[Fig. 6](#): Distribution of standardized gene expression values across the cancer cell line databases CCLE [1] and GDSC [39] as well as the human sample database TCGA [36].

5.4 Implementation and training details

All models were implemented in PyTorch 1.0 and trained on a cluster equipped with POWER8 processors and a NVIDIA Tesla P100.

PVAE. The model consisted of four dense layers of [1024, 512, 256 and 200] units with ReLU activation function and dropout of $p = 0.2$ in both, the encoder and the decoder. The dimensionality of the latent space (n) was 128. We minimized the variational loss, consisting of the reconstruction loss and KL divergence, using Adam optimizer ($\beta_1 = 0.9$, $\beta_2 = 0.999$, $\varepsilon = 1e-8$) and a decreasing learning rate starting at 0.001 [17]. To further regularize the PVAE, denoising methods were employed by 1) applying a dropout of 0.1 on the input genes and 2) adding noise to gene expression values ($\varepsilon \sim \mathcal{N}(0, 0.1)$). The model was trained with a batch size of 64 for a maximum of 2000 epochs.

SVAE. The model was trained on molecules provided in SMILES notation, the longest molecules had 1423 tokens. Both encoder and decoder consisted of two layers of bidirectional GRU (hidden size of 128, dropout of 0.1 at the first layer), each complemented with 50 parallel memory stacks with the depth of 50. The latent space of SVAE had the same dimensionality as the PVAE (128) to enable the addition of encodings. Similar optimization parameters as PVAE were used. This model further utilized teacher forcing [38], i.e., the model’s output is conditioned on the previous ground truth sample as opposed to its generated output. Whilst this significantly simplifies learning, it may drive the generator to predominantly rely on the decoder (thus neglecting the latent encoding). This so called posterior collapse was resolved by applying a token dropout rate of 0.1 during teacher forcing as suggested by [5]. In addition to token dropout, KL cost-annealing [5] was employed during training, The model was trained with a batch size of 128 for a maximum (early stopping) of $\sim 110,000$ steps (i.e., exactly 10 epochs) During training, KL cost-annealing as described in [5] was explored in order to trade-off reconstruction and KL loss.

Critic. The critic was trained using the parameters reported in [20] and replicating the best performing architecture based on multiscale convolutional encoders.

RL training. In order to maximize Equation 1, we employed Adam optimizer ($\beta_1 = 0.9$, $\beta_2 = 0.999$, $\varepsilon = 1e-4$, weight decay $1e-4$) and a decreasing learning rate starting at $1e-5$. The gradients were clipped to 2 to prevent \mathcal{G} from destroying its chemical knowledge about SMILES syntax obtained through pretraining on ChEMBL. The reward function hyperparameter α was set to 5.

5.5 SVAE qualitative analysis

Figure 7A, showcases a panel of 12 generated molecules for qualitative assessment of the molecular structures. The generated molecules generally share drug-like structural features. To inspect the smoothness of the latent space of molecules, we encoded a reference molecule shown at the top of **Figure 7B** into the latent space and decoded four points in the vicinity of the reference molecule leading to the generation of structurally similar yet different compounds.

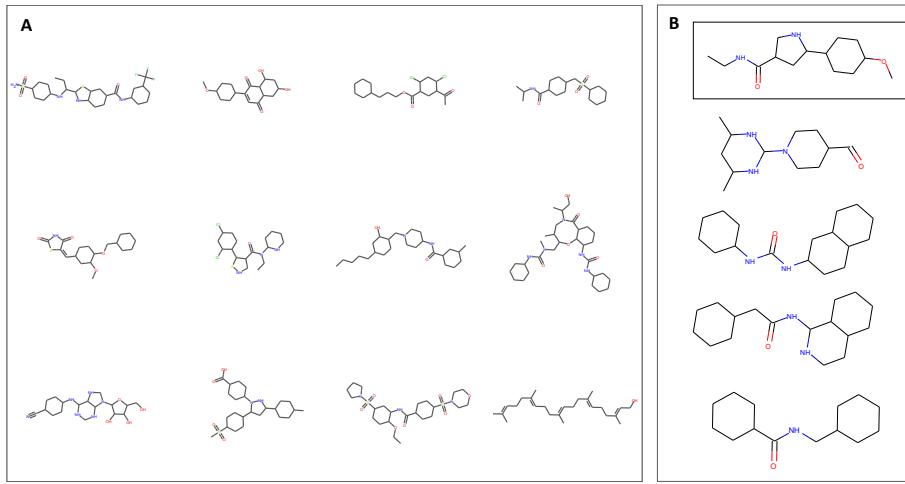


Fig. 7: Qualitative inspection of generated molecules. (A) A sample of 12 molecular structures produced with the SVAE. (B) The molecule depicted at the top was encoded into the latent space. The four molecules below show different decodings from the latent space in the vicinity of the starting molecule.

5.6 Chemical properties of generated molecules

In this work, the conditional generator is trained using PaccMann as sole critic. However, besides inhibitory efficacy, there is a myriad of properties of a candidate drug that crucially influence its potential for becoming an anticancer compound. Some of these can be approximated in-silico, e.g. water solubility (ESOL [8]), drug-likeness (QED [3]) and synthesizability (SAS [9]). Figure 8 gives an overview about the distribution of QED, SAS and ESOL scores for all drugs in GDSC and CCLE as well as a set of candidate drugs with high predicted efficacy. Comparing the distributions reveals that our generated molecules do not yet exhibit the same properties like real anticancer drugs. This is expected, since these scores are not yet incorporated into the reward function of the generator. It is curious that even within CCLE database, there is a significant portion of drugs with an estimated $\text{QED} < 0.2$. Whilst across both databases the average QED is around 0.5, it is only 0.2 for our molecules. As [28] reported, possessing a certain solubility is a key property of drugs, with not much deviation being tolerated (Figure 8 middle panel). In particular regarding the ease of synthesizability (see right panel of Figure 8), it is evident that further improvement is necessary.

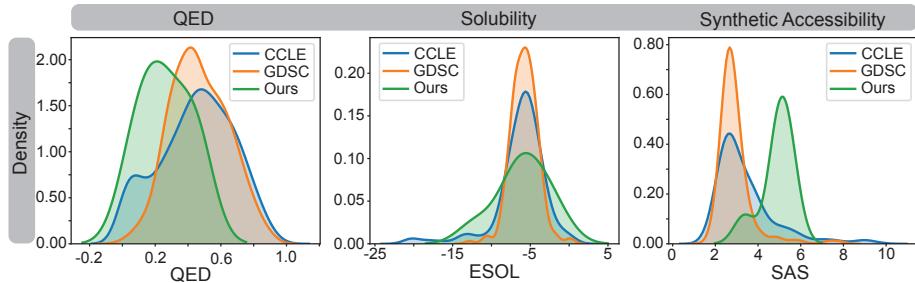


Fig. 8: Comparison of chemical scores for real drugs in GDSC and CCLE database versus our generated compounds. We compared three chemical scores for druglikeness as assessed by QED score (0 worst, 1 best), for solubility as assessed via ESOL, given in $\log(M/L)$ and for synthetic accessibility as assessed by SAS (1 best, 10 worst). These three scores are computed for all drugs from 1) GDSC database, 2) CCLE database and 3) a set of generated candidate compounds with high predicted efficacy.