nature machine intelligence

Article

https://doi.org/10.1038/s42256-024-00821-x

Invalid SMILES are beneficial rather than detrimental to chemical language models

Received: 11 September 2023

Accepted: 5 March 2024

Published online: 29 March 2024

Check for updates

Michael A. Skinnider [1,2]

Generative machine learning models have attracted intense interest for their ability to sample novel molecules with desired chemical or biological properties. Among these, language models trained on SMILES (Simplified Molecular-Input Line-Entry System) representations have been subject to the most extensive experimental validation and have been widely adopted. However, these models have what is perceived to be a major limitation: some fraction of the SMILES strings that they generate are invalid, meaning that they cannot be decoded to a chemical structure. This perceived shortcoming has motivated a remarkably broad spectrum of work designed to mitigate the generation of invalid SMILES or correct them post hoc. Here I provide causal evidence that the ability to produce invalid outputs is not harmful but is instead beneficial to chemical language models. I show that the generation of invalid outputs provides a self-corrective mechanism that filters low-likelihood samples from the language model output. Conversely, enforcing valid outputs produces structural biases in the generated molecules, impairing distribution learning and limiting generalization to unseen chemical space. Together, these results refute the prevailing assumption that invalid SMILES are a shortcoming of chemical language models and reframe them as a feature, not a bug.

Over the past century, more than 100 million small molecules have been synthesized in the search for new drugs and materials[1]. These efforts have explored only an infinitesimal subset of chemical space, the size of which is estimated at over 10^{60} molecules[2]. Yet, remarkably and often serendipitously, this limited exploration of chemical space has led to the discovery of numerous molecules that can modulate biological processes. That our extremely limited exploration of chemical space has already yielded so many medically or industrially valuable compounds suggests that more efficient approaches to chemical space exploration could help address many of the most pressing challenges facing humanity.

Chemical space is so large that its exhaustive enumeration is essentially impossible. Instead, searches for bioactive molecules generally focus on particular subsets of chemical space[3,4]. Historically, these subsets were defined primarily by rule-based approaches, in which new molecules were generated by iterative application of predefined chemical transformations to a 'starter' population[5–12]. More recently, generative models based on deep neural networks have emerged as a powerful framework for chemical space exploration[13–16]. Given a set of molecules as input, these models are able to learn the chemistries implicitly embedded within this training set, and then leverage this understanding to sample unseen molecules from the same areas of chemical space.

Initial demonstrations that deep generative models could design novel molecules with desired physicochemical or biological properties[17–24] have triggered the development of myriad approaches to molecule generation. These methods differ not only in the architectures of the underlying neural networks, but also in the conceptual frameworks they use to represent molecules: for instance, as chemical graphs[25,26], as combinations of substructures[27] or as three-dimensional objects[28,29]. Thus far, however, these approaches have not consistently surpassed the empirical state of

[1]Lewis-Sigler Institute for Integrative Genomics, Princeton University, Princeton, NJ, USA. [2]Ludwig Institute for Cancer Research, Princeton University, Princeton, NJ, USA. e-mail: skinnider@princeton.edu

https://doi.org/10.1038/s42256-024-00821-x

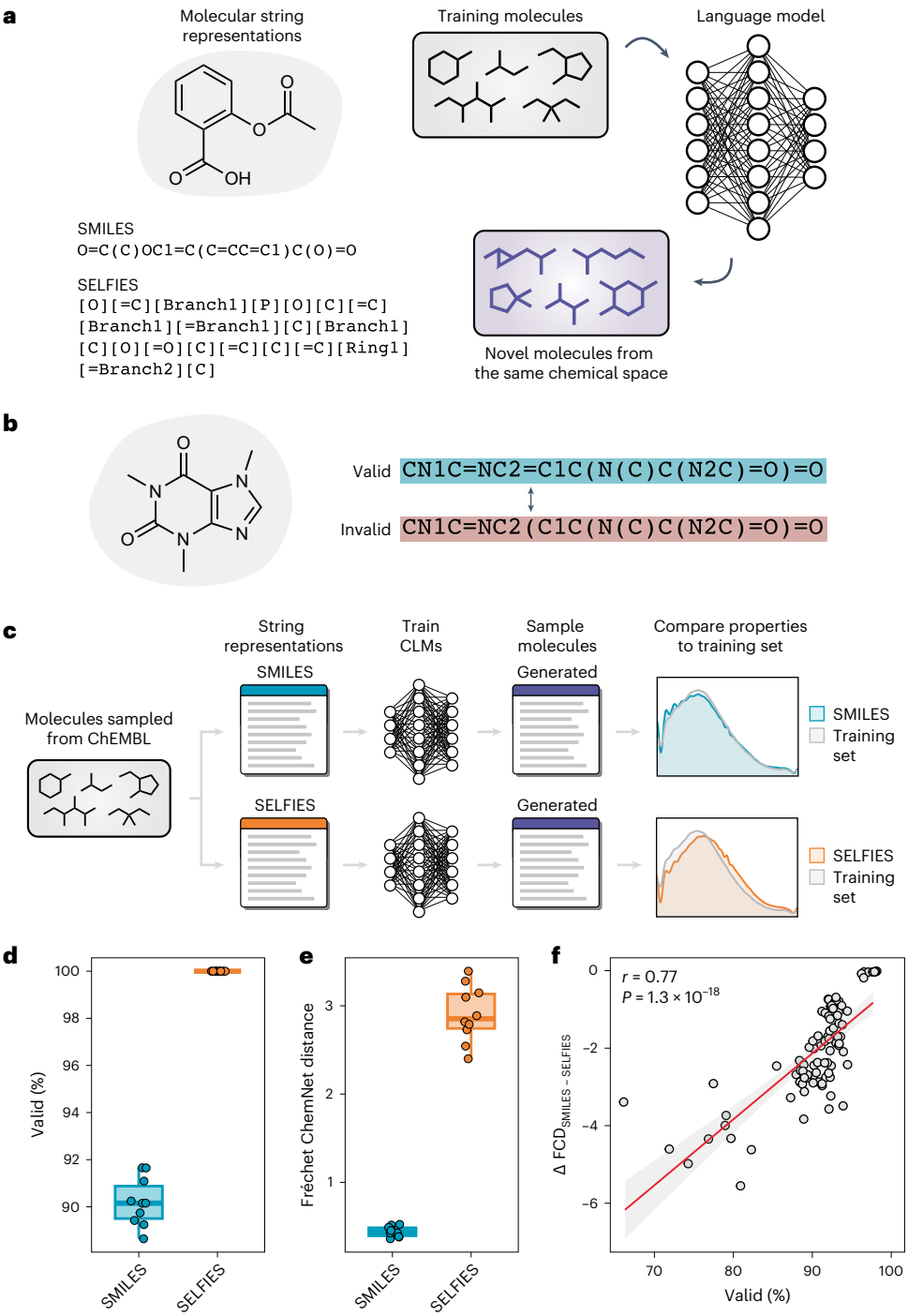


Fig. 1 | Language models that can generate invalid outputs outperform models that cannot. a, Schematic overview of chemical space exploration with chemical language models. Language models are trained on a set of chemical structures represented as strings (for example, in SMILES or SELFIES formats). Sampling new strings from the trained model enables generation of novel molecules from the same chemical space as the training set. **b**, Illustration of invalid SMILES. A single character substitution in the SMILES string for caffeine, top, creates a syntactically invalid SMILES string that does not correspond to any chemical structure, bottom. **c**, Experimental framework to benchmark language models trained on SMILES versus SELFIES. CLM, chemical language model. **d**, Proportion of valid molecules generated by language models trained on SMILES versus SELFIES representations ($n = 10$ each; $P = 2.0 \times 10^{-10}$, paired t-test). **e**, Fréchet ChemNet distance between generated and training molecules for language models trained on SMILES versus SELFIES representations (lower is better; $n = 10$ each; $P = 1.1 \times 10^{-9}$, paired t-test). **f**, Relationship between the proportion of valid SMILES generated by chemical language models, and the difference in Fréchet ChemNet distance (FCD) between each model and an equivalent model trained on SELFIES representations of the same training set. Inset text shows the Pearson correlation coefficient and P value. The line and shaded area show linear regression and 95% confidence interval, respectively.

the art established by the earliest deep neural network approaches based on chemical language models[30,31]. These models represent molecules as strings of text (commonly using the SMILES format[32]; Fig. 1a), and adapt neural network architectures from the field of natural language processing to learn the statistical properties of these strings and generate new ones.

Like a human language, the SMILES syntax imposes strict rules on which strings are syntactically valid. This means that chemical

language models can generate SMILES that do not correspond to any valid chemical structure (Fig. 1b). The generation of invalid SMILES is widely perceived to be an important shortcoming of chemical language models. This perception has motivated an enormous amount of work to address this shortcoming and encourage generation of valid molecules, whether by developing alternative textual representations of molecules[33–35], developing methods that generate valid SMILES by design[36,37], or developing methods to correct invalid SMILES post hoc[38–40]. The generation of invalid SMILES is also frequently cited as a motivation to eschew the language modelling framework and develop models that generate chemical graphs directly[25,27,41–47] and used in benchmark suites to quantify the performance of generative models[48,49].

That the generation of invalid SMILES is so widely perceived to be a limitation of chemical language models might be seen as surprising. Removing invalid SMILES from the output of a chemical language model is a simple post hoc processing step that does not carry substantial computational cost. Moreover, despite the assumption that generating invalid SMILES is a shortcoming, several benchmarks have identified that language models trained on SMILES outperform those trained on SELFIES (SELF-referencIng Embedded Strings)[34], a textual representation that produces 100% valid output by design, as well as models that generate chemical graphs directly[50–52]. These observations raise the possibility that the ability to generate invalid SMILES is actually a desirable property for a generative model: in other words, that generating invalid SMILES is a feature, not a bug.

In this study, I set out to empirically test the possibility that invalid SMILES are beneficial, rather than harmful, to chemical language models. I show that invalid SMILES are sampled with significantly lower likelihoods than valid SMILES, suggesting that filtering invalid SMILES provides an intrinsic mechanism to identify and remove low-quality samples from the model output. I then exploit the design of the SELFIES language by removing the valency constraints that ensure valid molecule generation and obtain causal evidence that generating invalid outputs improves the performance of chemical language models. I elucidate the mechanism by which imposing valency constraints impairs distribution learning, and show that these constraints bias chemical space exploration towards molecules with specific structural properties and impair generalization to unseen chemical space. Finally, I show that language models can correctly elucidate complex chemical structures from minimal analytical data, and that models capable of generating invalid outputs outperform models that cannot on this task.

Results

Models that generate invalid outputs outperform models that do not

Previous benchmarks suggested that chemical language models trained on SMILES could outperform those trained on SELFIES, a format in which every string corresponds to a valid molecule by design. Specifically, these benchmarks showed that language models trained on SMILES strings generated unseen molecules whose physicochemical properties better matched those of the molecules in the training set[50,51]. I initially set out to reproduce this observation. I trained chemical language models on random samples of molecules from the ChEMBL database[53], providing either SMILES or SELFIES representations of the same molecules as input. The trained models were then used to sample new molecules from the same chemical space as the training set, and model performance was evaluated by calculating metrics that captured the similarity between generated molecules and the training set (Fig. 1c).

As expected, models trained on SELFIES strings produced valid molecules at a rate of 100%, compared to an average of 90.2% for models trained on SMILES (Fig. 1d). Nonetheless, models trained on SMILES generated novel molecules that matched the training set significantly better than models trained on SELFIES, as quantified by the Fréchet ChemNet distance (Fig. 1e). This conclusion was unchanged when using other metrics to quantify performance, such as the Murcko scaffold similarity between the training and generated molecules[54], and was recapitulated when integrating multiple metrics into a single measure of model performance using principal component analysis (PCA), as previously described[50] (Extended Data Fig. 1a–d).

The superior performance of models trained on SMILES was robust both to the data used to train the chemical language models, and to the architecture of the models themselves. I reproduced this result when (1) training models on smaller or larger samples of molecules from ChEMBL; (2) training models on molecules from a different chemical database, GDB-13 (ref. 55); (3) training models on more or less chemically diverse training sets; (4) performing data augmentation by SMILES or SELFIES enumeration[56,57]; or (5) using a language model based on the transformer architecture[58] instead of one based on long short-term memory (LSTM) networks (Extended Data Fig. 1e–s).

Language models trained on SELFIES typically generated novel molecules at a higher rate than models trained on SMILES, but models trained on either representation were able to achieve a very high rate of novelty (>99%) except when deliberately constructing training sets with a low degree of chemical diversity (Extended Data Fig. 2).

Together, these results demonstrate that language models trained on SMILES robustly outperformed those trained on SELFIES. Moreover, across all models tested, I found that the magnitude of this difference in performance was strongly and negatively correlated with the proportion of valid SMILES (Fig. 1f): in other words, models trained on SMILES performed proportionately better when generating more invalid outputs.

Invalid SMILES are low-likelihood samples

These findings expose an apparent contradiction. The presence of invalid outputs is widely perceived to be a central shortcoming of generative models based on SMILES strings. However, models that can generate invalid outputs robustly outperformed models that—by design—can only generate valid outputs.

I sought to identify the mechanisms underlying this contradiction. One potential explanation is that invalid SMILES represent low-likelihood samples from the language model. Removing invalid SMILES would, therefore, function as a mechanism to filter low-quality samples from the model output. Notably, this hypothesis is consistent with the observed anticorrelation between invalid SMILES generation and model performance (Fig. 1f): filtering out a larger number of low-quality samples would expectantly result in proportionately better performance.

If this hypothesis were correct, one would expect that invalid SMILES are sampled with larger losses than valid SMILES from the same model. This was, indeed, found to be the case (Fig. 2a,b). Moreover, this difference was not limited to a single subtype of invalid SMILES: all major categories of invalid SMILES[38] were sampled with higher losses than their valid counterparts (Fig. 2c–e). These differences were mediated, in part, by the increased lengths of invalid SMILES, but persisted when comparing the average losses with which individual tokens were sampled within valid versus invalid SMILES (Extended Data Fig. 3a–d). Conversely, SMILES that were sampled with smaller losses were more likely to be valid (Fig. 2f). These findings were robust to varying the size and composition of the training dataset or the architecture of the language model (Extended Data Fig. 3e–m).

Invalid outputs improve performance

These results establish that invalid SMILES are enriched among low-likelihood samples from chemical language models. This finding suggests that removing invalid SMILES has the effect of filtering low-quality samples from the model output, which in turn would be expected to improve performance on distribution-learning metrics such as the Fréchet ChemNet distance. However, these data provide correlative rather than causal evidence for the notion that the

https://doi.org/10.1038/s42256-024-00821-x

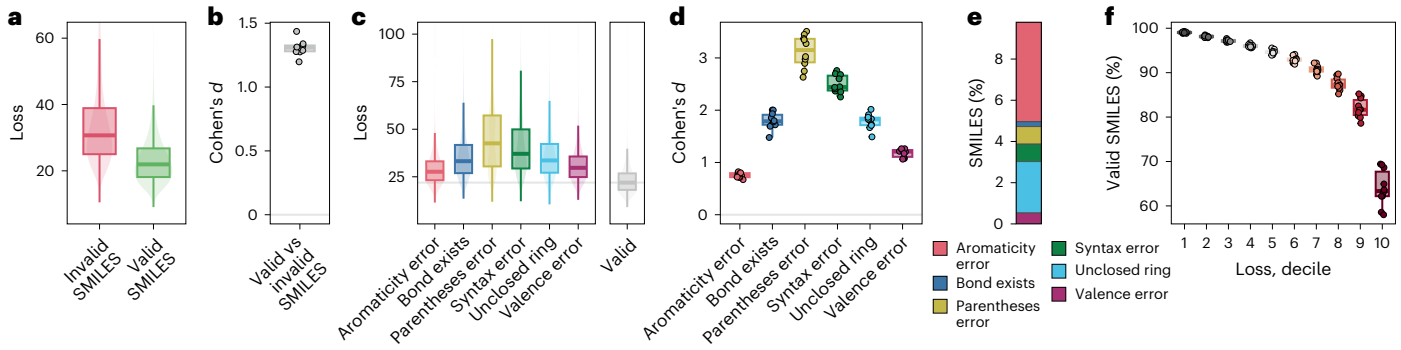


Fig. 2 | Invalid SMILES are low-likelihood samples from chemical language models. a, Losses of valid versus invalid SMILES sampled from a representative chemical language model ($n = 10^7$ SMILES; $P < 10^{-15}$, two-sided t-test). **b**, Effect sizes (Cohen's d) comparing the losses of valid versus invalid SMILES sampled from $n = 10$ chemical language models, demonstrating consistent effects ($P = 1.5 \times 10^{-13}$, one-sample t-test). **c**, Losses of valid SMILES versus invalid SMILES sampled from a representative chemical language model, classified into six different categories based on RDKit error messages[38] ($n = 10^7$ SMILES; all $P < 10^{-15}$, two-sided t-test). **d**, Effect sizes (Cohen's d) comparing the losses of valid SMILES versus six different categories of invalid SMILES across $n = 10$ chemical language models, demonstrating consistent effects (all $P \leq 1.4 \times 10^{-10}$, one-sample t-test). **e**, Frequencies of each invalid SMILES error type, shown as the mean proportion of all generated SMILES across ten chemical language models. **f**, Proportion of valid SMILES within each decile of loss in samples of 500,000 strings from ten chemical models ($P < 10^{-15}$, two-sided Jonckheere–Terpstra test).

ability to generate (and then discard) invalid outputs improves model performance.

To obtain such evidence, I took advantage of the design of SELFIES themselves. Within the SELFIES library, the generation of chemically valid graphs is enforced by a set of constraints on the valence of each atom: for example, the specification that a carbon atom cannot participate in more than four covalent bonds[34,59]. These valency constraints provide a natural mechanism to test the relationship between output validity and model performance. I modified the default valency constraints within the SELFIES library to allow pentavalent carbons, a modification I refer to as 'Texas SELFIES'[60]. Under these modified constraints, language models trained on SELFIES can generate chemically invalid outputs. Remarkably, however, these chemically invalid constraints significantly improved performance: decoding Texas SELFIES yielded samples of novel molecules that were more similar to the training set than those decoded with the default and chemically valid constraints (Fig. 3a and Extended Data Fig. 4a–d).

Next, I tested the effect of removing valency constraints entirely ('unconstrained SELFIES'), and found that this further improved performance (Fig. 3a and Extended Data Fig. 4e–h). Invalid SELFIES were sampled with larger losses than their valid counterparts (Fig. 3b,c), corroborating the trends observed for invalid SMILES, and supporting the notion that removing valency constraints provided a mechanism to filter low-quality samples from the model output.

The superior performance of unconstrained SELFIES was robust to variations in the training dataset or model architecture (Extended Data Fig. 4j,l,m,p,r). Moreover, I identified a significant correlation between the proportion of invalid SELFIES generated and the improvement in performance after removing valency constraints (Fig. 3d): in other words, models performed proportionately better when generating more invalid SELFIES.

Whereas removing valency constraints improved the performance of language models trained on SELFIES, these were generally still outperformed by models trained on SMILES, pointing to residual differences in performance as a function of molecular representation.

These results provide causal evidence that allowing chemical language models to produce invalid outputs improves their performance.

Enforcing valid outputs biases chemical space exploration

I sought to clarify the mechanisms by which the ability to produce invalid outputs improved the performance of chemical language models. I hypothesized that these differences in performance reflected differences in the chemical space explored by models trained on SMILES versus SELFIES. To address this possibility, I computed a series of properties for each generated molecule, and compared the resulting property distributions to those of the training set. By far the largest difference between models trained on SMILES versus SELFIES in this analysis involved their propensity to generate cyclic molecules. Molecules generated as SELFIES were markedly depleted for aromatic rings (Fig. 4a,b) and enriched for aliphatic rings (Fig. 4c,d), relative both to the training set and to molecules generated as SMILES. Smaller but statistically significant differences were observed for a range of other structural properties, reflecting pervasive differences in the chemical space explored by generative models trained on SMILES versus SELFIES (Fig. 4e and Extended Data Fig. 5a–t).

Together, these experiments identified significant differences in the chemical space explored by language models trained on SMILES versus SELFIES. To establish whether a causal relationship existed, I compared the SELFIES that could be successfully parsed without chemical valency constraints to those that required the imposition of these constraints in order to produce a valid chemical graph. This comparison allowed me to directly assess how removing invalid SELFIES influenced the distributions of structural properties among the generated molecules. Remarkably, I observed that the most profound differences between valid and invalid SELFIES again involved their propensity to contain aromatic and aliphatic rings. Invalid SELFIES were significantly depleted for aromatic rings, and enriched for aliphatic rings, relative to both the training set and to valid SELFIES (Fig. 4f,g). Conversely, disabling the valency constraints, and allowing the model to generate invalid SELFIES, reversed the structural differences between molecules generated as SMILES versus SELFIES.

This reversal led me to ask whether other structural differences between molecules generated as SMILES versus SELFIES were also reversed when disabling valency constraints. Indeed, I observed that the differences in structural properties between SMILES and SELFIES were strongly and significantly correlated to those between valid and invalid SELFIES (Fig. 4h and Extended Data Fig. 6a–w). Thus, the structural differences between molecules generated as SMILES versus SELFIES can be attributed at least in part to the correction of invalid outputs.

Together, these experiments expose the mechanism underlying differences in performance between generative models trained on SMILES versus SELFIES. The imposition of valency constraints in SELFIES prevents the generation of invalid outputs, but results in an overrepresentation of aliphatic rings and an underrepresentation of

https://doi.org/10.1038/s42256-024-00821-x

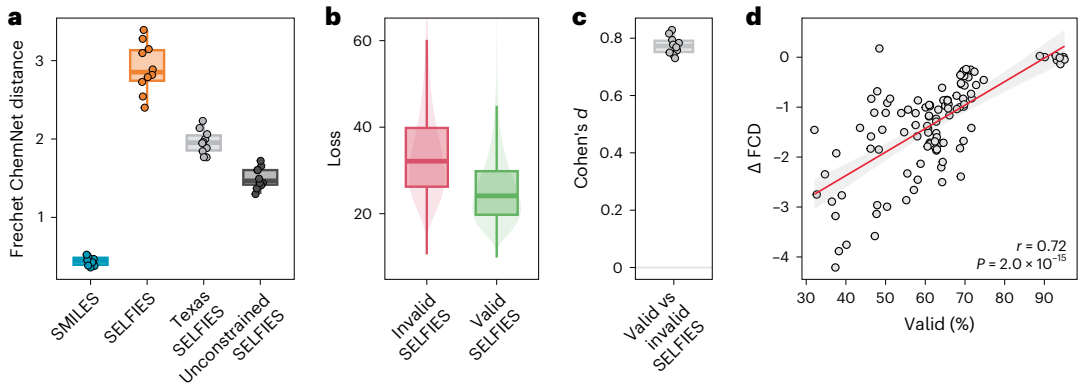


Fig. 3 | Invalid outputs improve the performance of chemical language models. a, Fréchet ChemNet distance between training and generated molecules for language models trained on SMILES or SELFIES representations, with SELFIES valency constraints modified to allow pentavalent carbons ('Texas SELFIES') or removed entirely ('unconstrained SELFIES'; $n = 10$ each; both $P \leq 3.0 \times 10^{-5}$ compared with default valency constraints, paired t-test). **b**, Losses of valid versus invalid SELFIES sampled from a representative chemical language model with valency constraints disabled when parsing generated SELFIES ($n = 10^7$ SELFIES; $P < 10^{-15}$, t-test). **c**, Effect sizes (Cohen's d) comparing the losses of valid versus invalid SMILES sampled from $n = 10$ chemical language models, demonstrating consistent effects ($P = 5.6 \times 10^{-14}$, two-sided one-sample t-test). **d**, Relationship between the proportion of valid SELFIES generated with valency constraints disabled, and the difference in Fréchet ChemNet distance when parsing generated SELFIES with or without valency constraints. Inset text shows the Pearson correlation coefficient and P value. The line and shaded area show linear regression and 95% confidence interval, respectively.

aromatic rings in the resulting molecules. These systematic biases in the chemical composition of the generated molecules are reflected in poor performance on distribution-learning metrics, such as the Fréchet ChemNet distance. Removing valency constraints, and allowing the model to generate invalid outputs, corrects these biases and improves performance.

Structural biases limit generalization

An ideal generative model would sample evenly from the chemical space surrounding the molecules in the training set. The observation of structural biases in the outputs of language models trained on SELFIES is at odds with this goal. I therefore sought to test whether, in addition to introducing biases in the chemical space explored by generative models, the choice of representation would also constrain their capacity for generalization.

To test this hypothesis, I made use of an exhaustively explored chemical space: that of the GDB-13 database, which enumerates all ~975 million drug-like molecules containing up to 13 heavy atoms. Following an experimental design proposed previously, I trained chemical language models on small samples from GDB-13, using either SMILES or SELFIES to represent these molecules[61]. I then drew samples of 100 million strings from each language model, and calculated the total proportion of GDB-13 that was correctly reproduced within the language model output.

Language models trained on SELFIES generated significantly more valid molecules than those trained on SMILES, as expected (Fig. 5a). However, a substantial fraction of the molecules generated as SELFIES were outside the chemical space defined by the GDB-13 database (Fig. 5b). Consequently, despite generating fewer molecules overall, models trained on SMILES explored a significantly larger proportion of the GDB-13 chemical space than models trained on SELFIES (Fig. 5c,d). That models trained on SELFIES showed a greater propensity to explore outside the chemical space of GDB-13, but a lower coverage of GDB-13 itself, can be rationalized on the basis that models trained on SELFIES show a diminished capacity to generalize from the chemical space of the training set.

Invalid outputs improve structure elucidation

To explore the implications of these findings further, I applied chemical language models to a task in which efficient navigation of unknown chemical space is of central importance: namely, structure elucidation of complex natural products. Recent work has shown that chemical language models can generate novel molecules that match experimentally measured properties. One particularly exciting observation is that language models can not only generate plausible chemical structures, but even prioritize the most likely ones on the basis of as little experimental data as an accurate mass measurement[62]. However, thus far this possibility has only been demonstrated for a subset of drug-like molecules, and it remains unclear whether the same approach could be applied to structure elucidation of more complex molecules. In Supplementary Note 1 and Extended Data Figs. 7–9, I show that language models can contribute to the structure elucidation of a range of complex small molecules including natural products, environmental pollutants, and food-derived compounds, and that the ability to generate invalid outputs improves performance on these tasks.

Discarding invalid SMILES is fast and easy

One potential criticism of generating (and then discarding) invalid outputs is that the process of parsing every sample from the model to establish its validity necessarily requires additional computational resources[63]. However, filtering invalid SMILES is a lightweight post-processing step that does not substantially increase the computational requirements of a chemical language model. Parsing 1 million SMILES can be achieved with the RDKit in an average of 7.5 minutes on a single CPU, and determining the validity of a SMILES string requires just a single line of code (Extended Data Fig. 10).

Discussion

That chemical language models trained on SMILES strings can produce invalid outputs is widely (if not universally) perceived to be an important deficiency of these models. This perception has motivated a remarkably broad spectrum of work in the field of chemical artificial intelligence, including the development of alternative molecular representations, mechanisms that encourage generation of valid outputs, approaches to correct invalid outputs post hoc, and models that generate chemical graphs directly. Here I provide direct and causal evidence that the ability to produce invalid outputs is not harmful but is instead beneficial to chemical language models, and elucidate the mechanisms underlying this effect. I show that language models trained on SMILES, a representation that can lead to both syntactically

https://doi.org/10.1038/s42256-024-00821-x

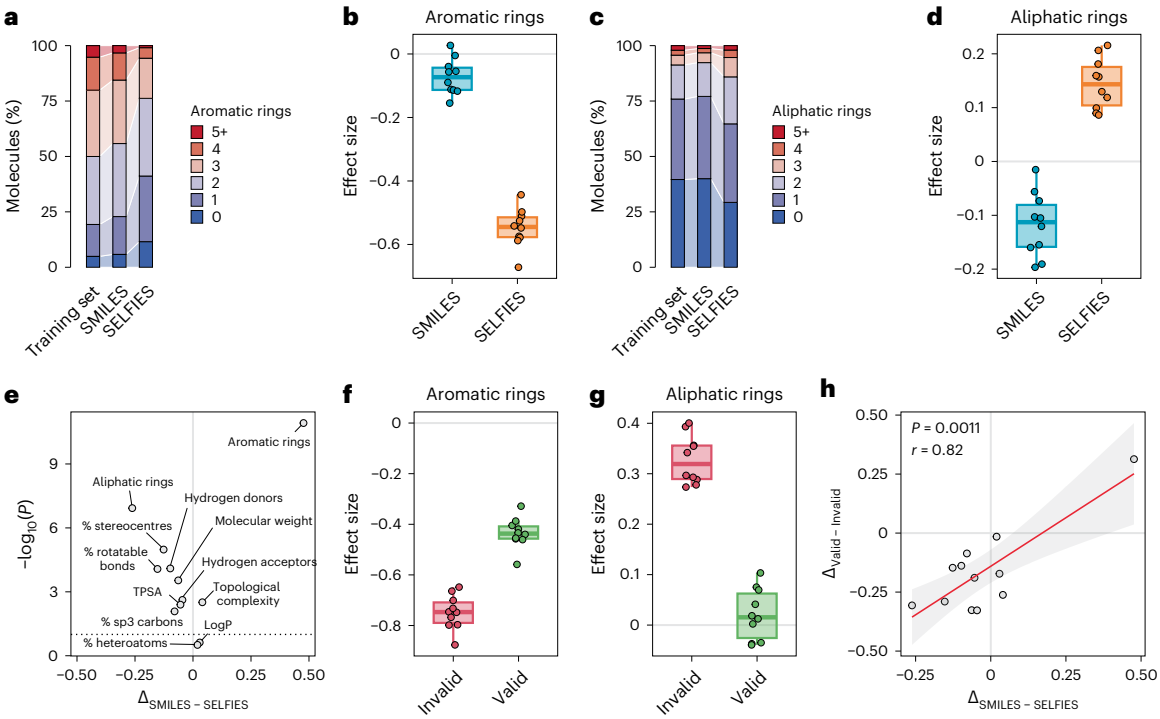


Fig. 4 | Enforcing valid outputs biases chemical space exploration. a, Number of aromatic rings in generated molecules sampled from representative chemical language models trained on the same molecules in SMILES versus SELFIES format, and in the training set molecules themelves. **b**, Effect sizes (Cohen's *d*) comparing the number of aromatic rings in generated molecules from chemical language models trained on SMILES versus SELFIES to the molecules in the training set ($n = 10$ each; $P = 1.2 \times 10^{-11}$, paired *t*-test). **c**, As in **a**, but showing aliphatic rings. **d**, As in **b**, but showing aliphatic rings ($P = 1.2 \times 10^{-7}$, paired *t*-test). **e**, Volcano plot showing differences in structural properties between molecules generated as SMILES versus SELFIES (statistical significance versus mean difference in effect size, paired *t*-test). Dotted line shows $P = 0.05$. **f**, Effect sizes (Cohen's *d*) comparing the number of aromatic rings in generated molecules from chemical language models to the molecules in the training set, shown separately for valid versus invalid SELFIES when parsing generated SELFIES without valency constraints ($n = 10$ each; $P = 7.6 \times 10^{-11}$, paired *t*-test). **g**, As in **f**, but showing aliphatic rings ($P = 2.6 \times 10^{-12}$, paired *t*-test). **h**, Differences in structural properties (mean effect sizes) are correlated between molecules generated as SMILES versus SELFIES (*x*-axis) and valid versus invalid SELFIES when parsing generated SELFIES without valency constraints (*y*-axis). Inset text shows the Pearson correlation coefficient and *P* value. The line and shaded area show linear regression and 95% confidence interval, respectively.

and semantically invalid outputs, outperform models trained on SELFIES, a representation that enforces the generation of valid outputs by design (Fig. 1). Invalid SMILES are sampled with significantly lower likelihoods than valid SMILES, implying that filtering invalid SMILES preferentially removes low-quality samples from the model output (Fig. 2). I leverage the design of the SELFIES representation by removing the valency constraints that enforce valid molecule generation, allowing me to show causally that generating (and then removing) invalid outputs improves language model performance (Fig. 3). I further show that the imposition of valency constraints results in biased exploration of chemical space, reflected in an overrepresentation of aliphatic rings and an underrepresentation of aromatic rings in the generated molecules (Fig. 4), and that these biases in turn impair generalization to unseen chemical space (Fig. 5). Finally, I apply chemical language models to structure elucidation of natural products, and show that (1) language models can develop remarkably accurate hypotheses about unknown chemical structures from minimal analytical data, and (2) models capable of generating invalid outputs significantly outperform models that cannot on this task (Extended Data Fig. 7).

Collectively, these results challenge the often-voiced assumption that invalid SMILES are a problem that must be addressed by developing new computational approaches. They suggest that further efforts to enforce the generation of valid molecules are unlikely to improve model performance. Instead, these results advocate for a more widespread recognition that removing invalid outputs is a simple and computationally efficient post-processing step that does not necessarily reflect a fundamental flaw in the underlying model. More broadly, these results support a redirection of efforts towards improving the performance of generative models of molecules through directions other than maximizing output validity. Indeed, several recent studies have highlighted opportunities to improve molecule generation despite the generation of invalid SMILES[64–67].

That language models trained on SMILES outperformed those trained on SELFIES on distribution-learning metrics does not imply the latter should never be preferred. A number of recent works have presented computational approaches in which the robustness of the SELFIES representation has been a central consideration, including applications to model interpretability[68,69] and inverse design[70,71]. In other words, while I demonstrate that the ability to generate invalid outputs is beneficial to chemical language models in general, there are specific scenarios in which validity is a more important consideration.

I found that removing the valency constraints that enforce valid molecule generation in SELFIES greatly reduced the difference in performance between models trained on SMILES versus SELFIES, but did not abolish it entirely (Fig. 3a). This observation suggests that there are residual differences between the two representations that go beyond the presence of invalid outputs and reflect deeper aspects of how they represent chemical structures. Elucidating the mechanisms underlying these differences will be an important direction for future work.

The application of chemical language models to structure elucidation of natural products indicates that these models can develop remarkably accurate hypotheses about complex and unseen

https://doi.org/10.1038/s42256-024-00821-x

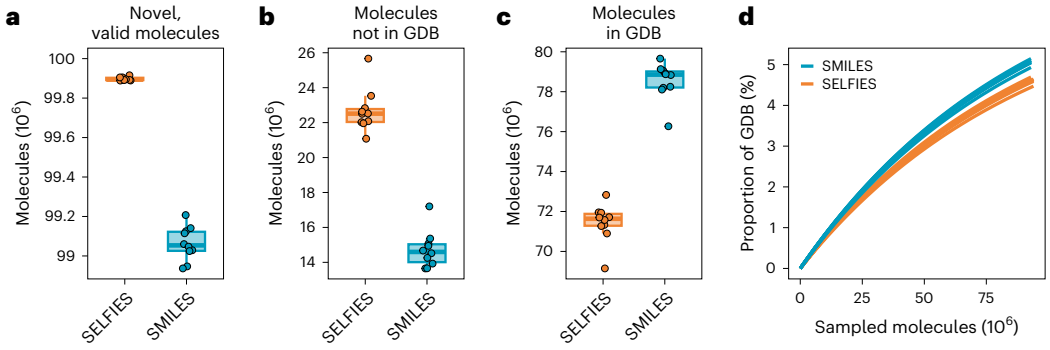


Fig. 5 | Structural biases limit generalization to unseen chemical space. **a**, Numbers of valid, novel molecules within samples of 100 million strings from chemical language models trained on a subset of 1 million molecules from the GDB-13 database, represented as SMILES versus SELFIES ($n = 10$ each; $P = 1.9 \times 10^{-10}$, paired t-test). **b**, As in **a**, but showing the number of sampled molecules outside the chemical space of the GDB-13 database within samples of 100 million strings ($n = 10$ each; $P = 8.5 \times 10^{-7}$). **c**, As in **a**, but showing the number of molecules from the full GDB-13 database reproduced within samples of 100 million strings ($n = 10$ each; $P = 1.1 \times 10^{-7}$). **d**, Saturation curve showing the proportion of the full GDB-13 database reproduced after sampling a given number of valid molecules from chemical language models trained on SMILES versus SELFIES.

chemical structures from as little data as an accurate mass measurement. Structures proposed by the chemical language model were more similar to the true molecule than those obtained by searching in PubChem, or even by searching in the natural product-like chemical space of the training set itself. This latter observation emphasizes the degree to which the model has learned to extrapolate beyond the training set and into unseen chemical space. Of course, the evaluation scenario here is by design unrealistic, and I do not mean to suggest that it is possible to perform complete structure elucidation of complex natural products from an accurate mass alone (not least because it is theoretically impossible to discriminate between different structures with the same molecular formula using only MS1 information). Instead, my intention in this experiment is to highlight that given minimal analytical data, language models can develop remarkably good hypotheses—sometimes even better than those based on much richer analytical data. This is exciting because, to my knowledge, these structural hypotheses represent a new source of information that is not currently used by any methods for structure elucidation of unknown molecules[72]. Integrating the novel chemical structures prioritized by chemical language models with computational approaches that leverage MS/MS or retention time information could provide a powerful mechanism to accelerate structure elucidation of unknown molecules in biological systems.

Methods

Datasets

My experiments initially focused on training chemical language models on random samples of molecules from the ChEMBL database[53]. ChEBML (version 28) was obtained from ftp.ebi.ac.uk/pub/databases/chembl/ChEMBLdb/latest/chembl_28_chemreps.txt.gz. Duplicate SMILES and SMILES that could not be parsed by the RDKit were removed. Salts and solvents were removed by splitting molecules into fragments and retaining only the heaviest fragment containing at least three heavy atoms, using code adapted from the Mol2vec package[73]. Charged molecules were neutralized using code adapted from the RDKit Cookbook. Molecules with atoms other than Br, C, Cl, F, H, I, N, O, P or S were removed, and molecules were converted to their canonical SMILES representations.

Random samples of between 30,000 and 300,000 molecules were then drawn from the preprocessed SMILES. In most experiments, molecules were sampled randomly to achieve uniform coverage of ChEMBL chemical space. Separately, the effect of the chemical diversity of the training set on model performance was assessed by sampling training sets of molecules with decreasing chemical diversity[50]. This was achieved by selecting a 'seed' molecule at random from ChEMBL and then computing the Tanimoto coefficient (Tc) between the seed molecule and the remainder of the database. The database was then filtered to retain only molecules with a Tc greater than some target minimum value. A minimum Tc of zero reflects random selection of molecules across the entire database, whereas increasing the minimum Tc selects molecules that are increasingly similar to the seed molecule (that is, decreasing diversity). The Tanimoto coefficient was calculated on Morgan fingerprints[74] with a radius of 3, folded to 1,024 bits.

Past studies reported that data augmentation by non-canonical SMILES enumeration[56] could significantly improve the performance of chemical language models[62,75,76]. I therefore also tested the effect of SMILES enumeration, which was performed using the SmilesEnumerator class available from http://github.com/EBjerrum/SMILES-enumeration, with augmentation factors of 10× or 30×. SELFIES enumeration was performed by first enumerating non-canonical SMILES and then converting these to SELFIES, which I verified produced multiple SELFIES representing the same molecules. Finally, conversion from SMILES to SELFIES was performed using the SELFIES package (version 2.1.1).

To verify that the results were not specific to the chemical space populated by molecules from ChEMBL, I also trained chemical language models on the GDB-13 database[55]. The GDB-13 database was obtained from Zenodo (https://doi.org/10.5281/zenodo.5172018) and underwent preprocessing identical to that described above for ChEMBL.

For each set of parameters tested (for example, training dataset size, degree of data augmentation, or input database), ten independent training datasets were created with random seeds in order to evaluate variability in model performance and enable statistical comparisons. In total, I trained 180 models (90 on SMILES and 90 on SELFIES) that differed according to the random seed, the size of the training dataset (30,000, 100,000 or 300,000 molecules), the chemical diversity (Tc ≥ 0.0, 0.05, 0.10, or 0.15), the degree of SMILES/SELFIES augmentation (canonical, 10×, 30×) and the input database (ChEMBL versus GDB-13). Models shown in the main text were trained on 100,000 molecules from ChEMBL, without data augmentation or chemical diversity filters.

Chemical language models

For most experiments, I trained chemical language models based on LSTMs, which have been widely adopted in the field and have been subject to extensive experimental validation. LSTMs were implemented in PyTorch, adapting code from the REINVENT package[77]. Briefly, each SMILES was converted into a sequence of tokens by splitting the SMILES string into its constituent characters, except for atomic symbols composed of two characters (Br, Cl) and environments within square

brackets, such as [nH]. SELFIES were tokenized using the split_selfies function from the selfies package. The vocabulary of the RNN then consisted of all unique tokens detected in the training data, as well as start-of-string and end-of-string characters and a padding token. The architecture of the language models consisted of a three-layer LSTM with a hidden layer of 1,024 dimensions, an embedding layer of 128 dimensions, and a linear decoder layer. Models were trained to minimize the cross-entropy loss of next-token prediction using the Adam optimizer with $\beta_1 = 0.9$ and $\beta_2 = 0.999$, with a batch size of 64 and a learning rate of 0.001. Ten percent of the molecules in the training set were reserved as a validation set and used to perform early stopping with a patience of 50,000 minibatches. A total of 500,000 SMILES strings were sampled from each trained model after completion of model training.

To confirm that these results were robust to the specific architecture of the chemical language models, I also trained a series of models based on the generative pretrained transformer (GPT) architecture[78]. Models were implemented in PyTorch, adapting code and hyperparameters from MolGPT[79]. The architecture consisted of an embedding layer of 256 dimensions, which was concatenated with a learned positional encoding and passed through eight transformer blocks. Each block comprised eight masked self-attention heads and a feed-forward network with a hidden layer of 1,024 dimensions using GELU activation, both preceded by layer normalization. Finally, the outputs of the transformer blocks were passed through a single linear decoder layer with layer normalization. The transformer models were trained as described above for language models based on LSTMs, except that the learning rate was set to 0.0005.

Evaluation of model performance

I evaluated the performance of chemical language models trained on SMILES versus SELFIES by drawing samples of 500,000 SMILES or SELFIES strings from the trained models, and then quantifying the similarity between the generated molecules and the molecules in the training set. I used multiple complementary metrics to evaluate similarity. As the primary measure of model performance, I measured the chemical similarity between the generated molecules and the training set as quantified by the Fréchet ChemNet distance[80]. This metric was previously found to be among the most reliable for evaluating chemical language models[50] and is included in multiple benchmark suites[48,49]. As secondary measures of model performance, I also calculated a series of other metrics that were found to be similarly reliable, including the Jensen–Shannon distances between the Murcko scaffold compositions[54], natural product-likeness scores[81], and fraction of atoms in each molecule that are stereocenters in both the generated and training molecules. Molecules from the training set were filtered before calculating any of the above metrics to ensure that models were not being rewarded for simply reproducing the training molecules. I additionally integrated these metrics into a single measure of model performance using PCA, as previously demonstrated in a study in which chemical language models were trained on datasets of between 1,000 and 500,000 molecules[50]. PCA was shown to recover a first principal component that correlated strongly with the size of the training dataset, and thus also the quality of the learned model, while accounting for the covariance between the underlying evaluation metrics. PCA was performed using the reference dataset collected in the prior study using the 'CLMeval' R package (https://github.com/skinnider/CLMeval), and evaluation metrics for models analysed in the present study were projected onto this PC space. Finally, I confirmed that models trained on SELFIES generated 100% valid molecules whereas models trained on SMILES did not. Valid molecules were defined as those that could be parsed by RDKit, using the Chem.MolFromSmiles function.

Analysis of invalid SMILES

To investigate the properties of invalid SMILES, I drew samples of 10 million SMILES strings from trained chemical language models, and identified invalid SMILES as those that could not be parsed by RDKit. I then compared the losses with which valid versus invalid SMILES were sampled from the chemical language model by calculating Cohen's *d*, as implemented in the R package effsize. Separately, I divided SMILES into ten bins according to their losses and calculated the proportion of valid SMILES in each bin, testing for the presence of a trend with the Jonckheere–Terpstra test as implemented in the R package clinfun. Last, for each invalid SMILES, the parsing errors returned by RDKit were classified into six different error types using code developed as part of the UnCorrupt SMILES approach[38] (available from GitHub at https://github.com/LindeSchoenmaker/SMILES-corrector), and the losses for invalid SMILES from each error type were compared to those of valid SMILES with Cohen's *d*.

Removal of SELFIES valency constraints

To causally test the relationship between the generation of invalid output and model performance, I leveraged the design of the SELFIES library by modifying the valency constraints that ensure semantically correct output. I initially modified the default constraints (encoded in the _DEFAULT_CONSTRAINTS dictionary) by allowing carbon atoms to participate in five bonds ('Texas SELFIES'). I then parsed generated SELFIES under this modified set of constraints, removed outputs that could not be parsed by the RDKit, and re-calculated the Fréchet ChemNet distance and other distribution-learning metrics. I also tested the effect of removing valency constraints entirely by setting all values in the _DEFAULT_CONSTRAINTS dictionary to 999 ('unconstrained SELFIES'), following a previous suggestion[59], after which I removed invalid SELFIES and re-calculated the distribution-learning metrics.

Properties of generated molecules

To understand why the generation of invalid outputs improved performance on distribution-learning metrics, I calculated a series of structural properties for molecules generated as SMILES versus SELFIES using the RDKit. I calculated the same properties for molecules generated as SELFIES that could be parsed without valency constraints, versus molecules parsed under the default constraints. The list of properties examined was as follows:

1. The molecular weight of each molecule.
2. The computed octanol–water partition coefficient[82] of each molecule.
3. The topological complexity[83] of each molecule.
4. The topological polar surface area[84] of each molecule.
5. The proportion of carbon atoms in each molecule that were sp^3 hybridized.
6. The proportion of rotatable bonds in each molecule.
7. The proportion of atoms in each molecule that were stereocentres.
8. The fraction of heteroatoms in each molecule.
9. The number of aliphatic rings in each molecule.
10. The number of aromatic rings in each molecule.
11. The total number of rings in each molecule.
12. The number of hydrogen donors in each molecule.
13. The number of hydrogen acceptors in each molecule.

For each of these properties, I compared the distributions of values for generated molecules versus the training set using Cohen's *d*. I then tested for statistically significant differences using a paired *t*-test. Separately, I computed the Pearson correlation between the mean difference in effect sizes in comparisons of (1) molecules generated as SELFIES versus SMILES and (2) molecules generated as valid versus invalid SELFIES.

Generalization to unseen chemical space

To evaluate the ability of chemical language models to generalize to unseen chemical space surrounding the training set, I adapted an

experimental design proposed previously[61]. I trained chemical language models on samples of 100,000 molecules from the GDB-13 database, represent either as SMILES or SELFIES, and then drew samples of 100 million strings from each trained model. I then intersected these samples with the remainder of the GDB-13 database by comparison of canonical SMILES. For each language model, I computed (1) the number of novel and unique molecules; (2) the number of generated molecules in GDB-13 (excluding the training set); and (3) the number of generated molecules not in either GDB-13 or the training set. The experiment was repeated ten times with different training sets to gauge variability and enable statistical comparison.

Structure elucidation with chemical language models

Previous work had reported that chemical language models could contribute to structure elucidation of unknown small molecules using mass spectrometry, given minimal analytical data as input[62]. Specifically, it was found that when drawing a very large sample of molecules from a trained language model, the frequency with which any given unique molecule appeared in this output provided a model-intrinsic measure that could be used to develop structural hypotheses from an accurate mass measurement. These hypotheses could then be further refined by integrating the sampling frequency with MS/MS data, using existing tools for MS/MS interpretation[85]. Here, I tested (1) whether this principle would apply to other classes of small molecules, including complex natural products and (2) whether the choice of representation would influence the accuracy of structure elucidation using this approach.

I evaluated the performance of chemical language models on this task using four databases representing three different categories of small molecules. These included natural products from the LOTUS[86] and COCONUT[87] databases, food-derived compounds from the FooDB database, and environmental compounds from the NORMAN suspect list[88]. All four databases were preprocessed as described above for ChEMBL, and then split at random (that is, without scaffold splitting) into ten folds. For each test fold, a language model was trained on the 90% of molecules comprising the training set, after which a total of 100 million strings were sampled from the trained model. The sampled molecules were then parsed with the RDKit, invalid outputs were discarded, and the frequency with which each canonical SMILES appeared in the model output was tabulated. Then, for each molecule in the held-out test set, the model output was searched with the exact mass of this query molecule (plus or minus a 10 part per million window) and the generated molecules were sorted in descending order by their sampling frequencies to provide a ranked list of structural hypotheses. A window of 10 ppm was used to allow for differences between the measured (experimental) and theoretical mass; the accuracy of the experimental measurements is usually expressed as a mass error in parts per million as

$$(\mathrm{mass}_{\mathrm{measured}} - \mathrm{mass}_{\mathrm{theoretical}})/\mathrm{mass}_{\mathrm{theoretical}} \times 10^6$$

The top-k accuracy was then calculated as the proportion of held-out molecules for which the correct chemical structure appeared within the k top-ranked outputs from the language model when ordered by sampling frequency (in the case of ties, molecules were ordered at random). A similar evaluation was carried out at the level of molecular formulas, whereby the top-k accuracy was calculated as the proportion of held-out molecules for which the correct formula appeared within the k top-ranked model outputs. In addition, I calculated the Tanimoto coefficient between each held-out molecule and the top-ranked chemical structure proposed by the language model, using Morgan fingerprints with a radius of 3 as above. The entire process was repeated in ten-fold cross-validation.

As a baseline, I compared this language model-directed approach to searching by exact mass in PubChem, mimicking one approach to assigning chemical structures or molecular formulas based on MS1 measurements. I also compared the language model approach to searching by exact mass in the training set itself, recognizing that this would by definition lead to a top-k accuracy of 0 for any value of k, but with the goal of comparing the Tanimoto coefficients between the true molecule and structures prioritized by the language model versus molecules from the training set as a means to assess generalization beyond the training set. In both baselines, the same 10 ppm error window was used, within which molecules were ordered at random.

Finally, I repeated the procedures above with language models trained on SELFIES representations of the same databases, using identical folds. In addition, I parsed molecules generated as SELFIES without valency constraints as described above, discarded invalid outputs, and then re-calculated the sampling frequency of each generated molecule after canonicalization.

CASMI 2022

To further place the performance of chemical language models in context, I benchmarked the language model trained on the LOTUS database against 19 submissions to the CASMI 2022 competition (two submissions, Nikolic_KUCA and Nikolic_POSO, were excluded because these represented manual rather than computational approaches, per the organizers of the competition). In this competition, 500 commercially available compounds were profiled by mass spectrometry; four compounds (identifiers 81, 282, 432, 476) were subsequently excluded from the competition. Entrants were provided with the accurate m/z and retention times for each compound, and an accompanying mzML file containing MS/MS data. I tested the performance of the chemical language model given only the accurate m/z value as input. To simulate de novo elucidation of unknown molecules, I averaged sampling frequencies across all 10 cross-validation folds, removing training set molecules from the generative model output for each fold. This procedure ensured that sampling frequencies could be generated for known natural products without data leakage. For each accurate m/z value, I considered multiple potential adducts ($[M + H]^+$, $[M + NH_4]^+$ and $[M + Na]^+$ in the positive mode and $[M - H]^-$, $[M + Cl]^-$ and $[M + FA - H]^-$ in the negative mode) and retrieved sampled molecules with the corresponding exact masses, plus or minus a 10 ppm error window as above. Generated molecules were then sorted in descending order by sampling frequency across all potential adduct types to produce a ranked list of hypotheses for each target m/z value.

Visualization

Throughout the manuscript, the box plots show the median (horizontal line), interquartile range (hinges) and smallest and largest values no more than 1.5 times the interquartile range (whiskers), and the error bars show the standard deviation.

Data availability

Datasets used to train chemical language models, unprocessed samples of 10 million molecules from each model trained on ChEMBL or GDB-13, and samples of 100 million molecules from each cross-validation fold of LOTUS, COCONUT, FooDB and NORMAN, represented as canonical SMILES and sorted by sampling frequency, are available via Zenodo (https://doi.org/10.5281/zenodo.8321735)[89].

Code availability

Source code used to train models, analyse generated molecules and reproduce the figures, along with relevant intermediate data files, is available via Zenodo (https://doi.org/10.5281/zenodo.10680855)[90].

References

1. Reymond, J.-L. The chemical space project. *Acc. Chem. Res.* **48**, 722–730 (2015).
2. Bohacek, R. S., McMartin, C. & Guida, W. C. The art and practice of structure-based drug design: a molecular modeling perspective. *Med. Res. Rev.* **16**, 3–50 (1996).

3. Lipinski, C. & Hopkins, A. Navigating chemical space for biology and medicine. *Nature* **432**, 855–861 (2004).
4. Dobson, C. M. Chemical space and biology. *Nature* **432**, 824–828 (2004).
5. Lameijer, E.-W., Kok, J. N., Bäck, T. & Ijzerman, A. P. The molecule evoluator. An interactive evolutionary algorithm for the design of drug-like molecules. *J. Chem. Inf. Model.* **46**, 545–552 (2006).
6. van Deursen, R. & Reymond, J.-L. Chemical space travel. *ChemMedChem* **2**, 636–640 (2007).
7. Nicolaou, C. A., Apostolakis, J. & Pattichis, C. S. De novo drug design using multiobjective evolutionary graphs. *J. Chem. Inf. Model.* **49**, 295–307 (2009).
8. Virshup, A. M., Contreras-García, J., Wipf, P., Yang, W. & Beratan, D. N. Stochastic voyages into uncharted chemical space produce a representative library of all possible drug-like compounds. *J. Am. Chem. Soc.* **135**, 7296–7303 (2013).
9. Li, L. et al. MyCompoundID: using an evidence-based metabolome library for metabolite identification. *Anal. Chem.* **85**, 3401–3408 (2013).
10. Djoumbou-Feunang, Y. et al. BioTransformer: a comprehensive computational tool for small molecule metabolism prediction and metabolite identification. *J. Cheminform.* **11**, 2 (2019).
11. Jeffryes, J. G. et al. MINEs: open access databases of computationally predicted enzyme promiscuity products for untargeted metabolomics. *J. Cheminform.* **7**, 44 (2015).
12. Jensen, J. H. A graph-based genetic algorithm and generative model/Monte Carlo tree search for the exploration of chemical space. *Chem. Sci.* **10**, 3567–3572 (2019).
13. Anstine, D. M. & Isayev, O. Generative models as an emerging paradigm in the chemical sciences. *J. Am. Chem. Soc.* **145**, 8736–8750 (2023).
14. Bilodeau, C., Jin, W., Jaakkola, T., Barzilay, R. & Jensen, K. F. Generative models for molecular discovery: recent advances and challenges. *WIREs Comput. Mol. Sci.* https://doi.org/10.1002/wcms.1608 (2022).
15. Schwalbe-Koda, D. & Gómez-Bombarelli, R. Generative models for automatic chemical design. In *Machine Learning Meets Quantum Physics* (eds Schütt, K. T. et al.) Vol. 968, 445–467 (Springer, 2020).
16. Vanhaelen, Q., Lin, Y.-C. & Zhavoronkov, A. The advent of generative chemistry. *ACS Med. Chem. Lett.* **11**, 1496–1505 (2020).
17. Merk, D., Friedrich, L., Grisoni, F. & Schneider, G. De novo design of bioactive small molecules by artificial intelligence. *Mol. Inform.* **37**, 1700153 (2018).
18. Zhavoronkov, A. et al. Deep learning enables rapid identification of potent DDR1 kinase inhibitors. *Nat. Biotechnol.* **37**, 1038–1040 (2019).
19. Merk, D., Grisoni, F., Friedrich, L. & Schneider, G. Tuning artificial intelligence on the de novo design of natural-product-inspired retinoid X receptor modulators. *Commun. Chem.* **1**, 68 (2018).
20. Moret, M., Helmstädter, M., Grisoni, F., Schneider, G. & Merk, D. Beam search for automated design and scoring of novel ROR ligands with machine intelligence. *Angew. Chem. Int. Ed.* **60**, 19477–19482 (2021).
21. Grisoni, F. et al. Combining generative artificial intelligence and on-chip synthesis for de novo drug design. *Sci. Adv.* **7**, eabg3338 (2021).
22. Li, Y. et al. Generative deep learning enables the discovery of a potent and selective RIPK1 inhibitor. *Nat. Commun.* **13**, 6891 (2022).
23. Ballarotto, M. et al. *De novo* design of Nurr1 agonists *via* fragment-augmented generative deep learning in low-data regime. *J. Med. Chem.* **66**, 8170–8177 (2023).
24. Moret, M. et al. Leveraging molecular structure and bioactivity with chemical language models for de novo drug design. *Nat. Commun.* **14**, 114 (2023).
25. Li, Y., Zhang, L. & Liu, Z. Multi-objective de novo drug design with conditional graph generative model. *J. Cheminform.* **10**, 33 (2018).
26. Mercado, R. et al. Practical notes on building molecular graph generative models. *Appl. AI Lett.* https://doi.org/10.1002/ail2.18 (2020).
27. Jin, W., Barzilay, R. & Jaakkola, T. Junction tree variational autoencoder for molecular graph generation. In *Proc. 35th International Conference on Machine Learning* (eds Dy, J. & Krause, A.) Vol. 80, 2323–2332 (PMLR, 2018).
28. Li, Y., Pei, J. & Lai, L. Structure-based de novo drug design using 3D deep generative models. *Chem. Sci.* **12**, 13664–13675 (2021).
29. Xie, W., Wang, F., Li, Y., Lai, L. & Pei, J. Advances and challenges in de novo drug design using three-dimensional deep generative models. *J. Chem. Inf. Model.* **62**, 2269–2279 (2022).
30. Segler, M. H. S., Kogej, T., Tyrchan, C. & Waller, M. P. Generating focused molecule libraries for drug discovery with recurrent neural networks. *ACS Cent. Sci.* **4**, 120–131 (2018).
31. Gómez-Bombarelli, R. et al. Automatic chemical design using a data-driven continuous representation of molecules. *ACS Cent. Sci.* **4**, 268–276 (2018).
32. Weininger, D. SMILES, a chemical language and information system. 1. Introduction to methodology and encoding rules. *J. Chem. Inf. Model.* **28**, 31–36 (1988).
33. O'Boyle, N. & Dalke, A. DeepSMILES: an adaptation of SMILES for use in machine-learning of chemical structures. https://doi.org/10.26434/chemrxiv.7097960 (2018).
34. Krenn, M., Häse, F., Nigam, A., Friederich, P. & Aspuru-Guzik, A. Self-referencing embedded strings (SELFIES): a 100% robust molecular string representation. *Mach. Learn. Sci. Technol.* **1**, 045024 (2020).
35. Cheng, A. H. et al. Group SELFIES: a robust fragment-based molecular string representation. *Digital Discov.* **2**, 748–758 (2023).
36. Dai, H., Tian, Y., Dai, B., Skiena, S. & Song, L. Syntax-directed variational autoencoder for structured data. Preprint at https://doi.org/10.48550/arXiv.1802.08786 (2018).
37. Kusner, M. J., Paige, B. & Hernández-Lobato, J. M. Grammar variational autoencoder. Preprint at https://doi.org/10.48550/arxiv.1703.01925 (2017).
38. Schoenmaker, L., Béquignon, O. J. M., Jespers, W. & van Westen, G. J. P. UnCorrupt SMILES: a novel approach to de novo design. *J. Cheminform.* **15**, 22 (2023).
39. Zheng, S., Rao, J., Zhang, Z., Xu, J. & Yang, Y. Predicting retrosynthetic reactions using self-corrected transformer neural networks. *J. Chem. Inf. Model.* **60**, 47–55 (2020).
40. Bilsland, A. E., McAulay, K., West, R., Pugliese, A. & Bower, J. Automated generation of novel fragments using screening data, a dual SMILES autoencoder, transfer learning and syntax correction. *J. Chem. Inf. Model.* **61**, 2547–2559 (2021).
41. Walters, W. P. & Barzilay, R. Applications of deep learning in molecule generation and molecular property prediction. *Acc. Chem. Res.* **54**, 263–270 (2021).
42. Guo, M. et al. Data-efficient graph grammar learning for molecular generation. Preprint at https://doi.org/10.48550/arxiv.2203.08031 (2022).
43. De Cao, N. & Kipf, T. MolGAN: An implicit generative model for small molecular graphs. Preprint at https://doi.org/10.48550/arXiv.1805.11973 (2018).
44. Li, Y., Vinyals, O., Dyer, C., Pascanu, R. & Battaglia, P. Learning deep generative models of graphs. Preprint at https://doi.org/10.48550/arXiv.1803.03324 (2018).
45. Ma, T., Chen, J. & Xiao, C. Constrained generation of semantically valid graphs via regularizing variational autoencoders. Preprint at https://doi.org/10.48550/arXiv.1809.02630 (2018).
46. Liu, Q., Allamanis, M., Brockschmidt, M. & Gaunt, A. L. Constrained graph variational autoencoders for molecule design. Preprint at https://doi.org/10.48550/arxiv.1805.09076 (2018).

47. Zang, C. & Wang, F. MoFlow: an invertible flow model for generating molecular graphs. In *Proc. 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining* https://doi.org/10.1145/3394486.3403104 (ACM, 2020).
48. Polykovskiy, D. et al. Molecular sets (MOSES): a benchmarking platform for molecular generation models. *Front. Pharmacol.* **11**, 565644 (2020).
49. Brown, N., Fiscato, M., Segler, M. H. S. & Vaucher, A. C. GuacaMol: benchmarking models for de novo molecular design. *J. Chem. Inf. Model.* **59**, 1096–1108 (2019).
50. Skinnider, M. A., Stacey, R. G., Wishart, D. S. & Foster, L. J. Chemical language models enable navigation in sparsely populated chemical space. *Nat. Mach. Intell.* **3**, 759–770 (2021).
51. Flam-Shepherd, D., Zhu, K. & Aspuru-Guzik, A. Language models can learn complex molecular distributions. *Nat. Commun.* **13**, 3293 (2022).
52. Mahmood, O., Mansimov, E., Bonneau, R. & Cho, K. Masked graph modeling for molecule generation. *Nat. Commun.* **12**, 3156 (2021).
53. Mendez, D. et al. ChEMBL: towards direct deposition of bioassay data. *Nucleic Acids Res.* **47**, D930–D940 (2019).
54. Bemis, G. W. & Murcko, M. A. The properties of known drugs. 1. Molecular frameworks. *J. Med. Chem.* **39**, 2887–2893 (1996).
55. Blum, L. C. & Reymond, J.-L. 970 million druglike small molecules for virtual screening in the chemical universe database GDB-13. *J. Am. Chem. Soc.* **131**, 8732–8733 (2009).
56. Bjerrum, E. J. SMILES enumeration as data augmentation for neural network modeling of molecules. Preprint at https://doi.org/10.48550/arXiv.1703.07076 (2017).
57. Born, J. et al. Chemical representation learning for toxicity prediction. *Digital Discov.* https://doi.org/10.1039/D2DD00099G (2023).
58. Vaswani, A. et al. Attention is all you need. Preprint at https://doi.org/10.48550/arxiv.1706.03762 (2017).
59. Lo, A. et al. Recent advances in the self-referencing embedded strings (SELFIES) library. *Digital Discov.* **2**, 897–908 (2023).
60. Kiappes, J. L. in *Technology-Enabled Blended Learning Experiences for Chemistry Education and Outreach* (eds Fung, F. M. & Zimmermann, C.) Ch. 3, 43–64 (Elsevier, 2021).
61. Arús-Pous, J. et al. Exploring the GDB-13 chemical space using deep generative models. *J. Cheminform.* **11**, 20 (2019).
62. Skinnider, M. A. et al. A deep generative model enables automated structure elucidation of novel psychoactive substances. *Nat. Mach. Intell.* **3**, 973–984 (2021).
63. Elton, D. C., Boukouvalas, Z., Fuge, M. D. & Chung, P. W. Deep learning for molecular design—a review of the state of the art. *Mol. Syst. Des. Eng.* https://doi.org/10.1039/C9ME00039A (2019).
64. Özçelik, R., de Ruiter, S. & Grisoni, F. Structured state-space sequence models for de novo drug design. Preprint at *ChemRxiv* https://doi.org/10.26434/chemrxiv-2023-jwmf3 (2023).
65. Guo, J. et al. Improving de novo molecular design with curriculum learning. *Nat. Mach. Intell.* **4**, 555–563 (2022).
66. Mokaya, M. et al. Testing the limits of SMILES-based de novo molecular generation with curriculum and deep reinforcement learning. *Nat. Mach. Intell.* https://doi.org/10.1038/s42256-023-00636-2 (2023).
67. Born, J. & Manica, M. Regression transformer enables concurrent sequence regression and generation for molecular language modelling. *Nat. Mach. Intell.* https://doi.org/10.1038/s42256-023-00639-z (2023).
68. Wellawatte, G. P., Seshadri, A. & White, A. D. Model agnostic generation of counterfactual explanations for molecules. *Chem. Sci.* **13**, 3697–3705 (2022).
69. Gandhi, H. A. & White, A. D. Explaining molecular properties with natural language. Preprint at *ChemRxiv* https://doi.org/10.26434/chemrxiv-2022-v5p6m-v3 (2022).
70. Nigam, A., Pollice, R., Krenn, M., Gomes, G. D. P. & Aspuru-Guzik, A. Beyond generative models: superfast traversal, optimization, novelty, exploration and discovery (STONED) algorithm for molecules using SELFIES. *Chem. Sci.* **12**, 7079–7090 (2021).
71. Shen, C., Krenn, M., Eppel, S. & Aspuru-Guzik, A. Deep molecular dreaming: inverse machine learning for *de-novo* molecular design and interpretability with surjective representations. *Mach. Learn. Sci. Technol.* **2**, 03LT02 (2021).
72. Hu, G. & Qiu, M. Machine learning-assisted structure annotation of natural products based on MS and NMR data. *Nat. Prod. Rep.* **40**, 1735–1753 (2023).
73. Jaeger, S., Fulle, S. & Turk, S. Mol2vec: unsupervised machine learning approach with chemical intuition. *J. Chem. Inf. Model.* **58**, 27–35 (2018).
74. Rogers, D. & Hahn, M. Extended-connectivity fingerprints. *J. Chem. Inf. Model.* **50**, 742–754 (2010).
75. Arús-Pous, J. et al. Randomized SMILES strings improve the quality of molecular generative models. *J. Cheminform.* **11**, 71 (2019).
76. Moret, M., Friedrich, L., Grisoni, F., Merk, D. & Schneider, G. Generative molecular design in low data regimes. *Nat. Mach. Intell.* **2**, 171–180 (2020).
77. Blaschke, T. et al. REINVENT 2.0: an AI tool for de novo drug design. *J. Chem. Inf. Model.* **60**, 5918–5922 (2020).
78. Radford, A., Narasimhan, K., Salimans, T. & Sutskever, I. Improving language understanding by generative pre-training. https://openai.com/research/language-unsupervised (2018).
79. Bagal, V., Aggarwal, R., Vinod, P. K. & Priyakumar, U. D. MolGPT: Molecular generation using a transformer-decoder model. *J. Chem. Inf. Model.* **62**, 2064–2076 (2022).
80. Preuer, K., Renz, P., Unterthiner, T., Hochreiter, S. & Klambauer, G. Fréchet chemnet distance: a metric for generative models for molecules in drug discovery. *J. Chem. Inf. Model.* **58**, 1736–1741 (2018).
81. Ertl, P., Roggo, S. & Schuffenhauer, A. Natural product-likeness score and its application for prioritization of compound libraries. *J. Chem. Inf. Model.* **48**, 68–74 (2008).
82. Wildman, S. A. & Crippen, G. M. Prediction of physicochemical parameters by atomic contributions. *J. Chem. Inf. Comput. Sci.* **39**, 868–873 (1999).
83. Bertz, S. H. The first general index of molecular complexity. *J. Am. Chem. Soc.* **103**, 3599–3601 (1981).
84. Ertl, P., Rohde, B. & Selzer, P. Fast calculation of molecular polar surface area as a sum of fragment-based contributions and its application to the prediction of drug transport properties. *J. Med. Chem.* **43**, 3714–3717 (2000).
85. Wang, F. et al. CFM-ID 4.0: More accurate ESI-MS/MS spectral prediction and compound identification. *Anal. Chem.* **93**, 11692–11700 (2021).
86. Rutz, A. et al. The LOTUS initiative for open knowledge management in natural products research. *eLife* **11**, e70780 (2022).
87. Sorokina, M., Merseburger, P., Rajan, K., Yirik, M. A. & Steinbeck, C. COCONUT online: collection of open natural products database. *J. Cheminform.* **13**, 2 (2021).
88. Mohammed Taha, H. et al. The NORMAN Suspect List Exchange (NORMAN-SLE): facilitating European and worldwide collaboration on suspect screening in high resolution mass spectrometry. *Environ. Sci. Eur.* **34**, 104 (2022).
89. Skinnider, M. A. Molecules used to train or generated by chemical language models. *Zenodo* https://doi.org/10.5281/zenodo.8321735 (2023).
90. Skinnider, M. A. Code used to train chemical language models and analyze generated molecules. *Zenodo* https://doi.org/10.5281/zenodo.10680855 (2024).

Acknowledgements

M.A.S. acknowledges support from Ludwig Cancer Research.

Author contributions

M.A.S. designed and performed experiments and wrote the manuscript.

Competing interests

The author declares no competing interests.

Additional information

Extended data is available for this paper at https://doi.org/10.1038/s42256-024-00821-x.

Supplementary information The online version contains supplementary material available at https://doi.org/10.1038/s42256-024-00821-x.

Correspondence and requests for materials should be addressed to Michael A. Skinnider.

Peer review information *Nature Machine Intelligence* thanks Tao Huan, Saer Samanipour, and the other, anonymous, reviewer(s) for their contribution to the peer review of this work.

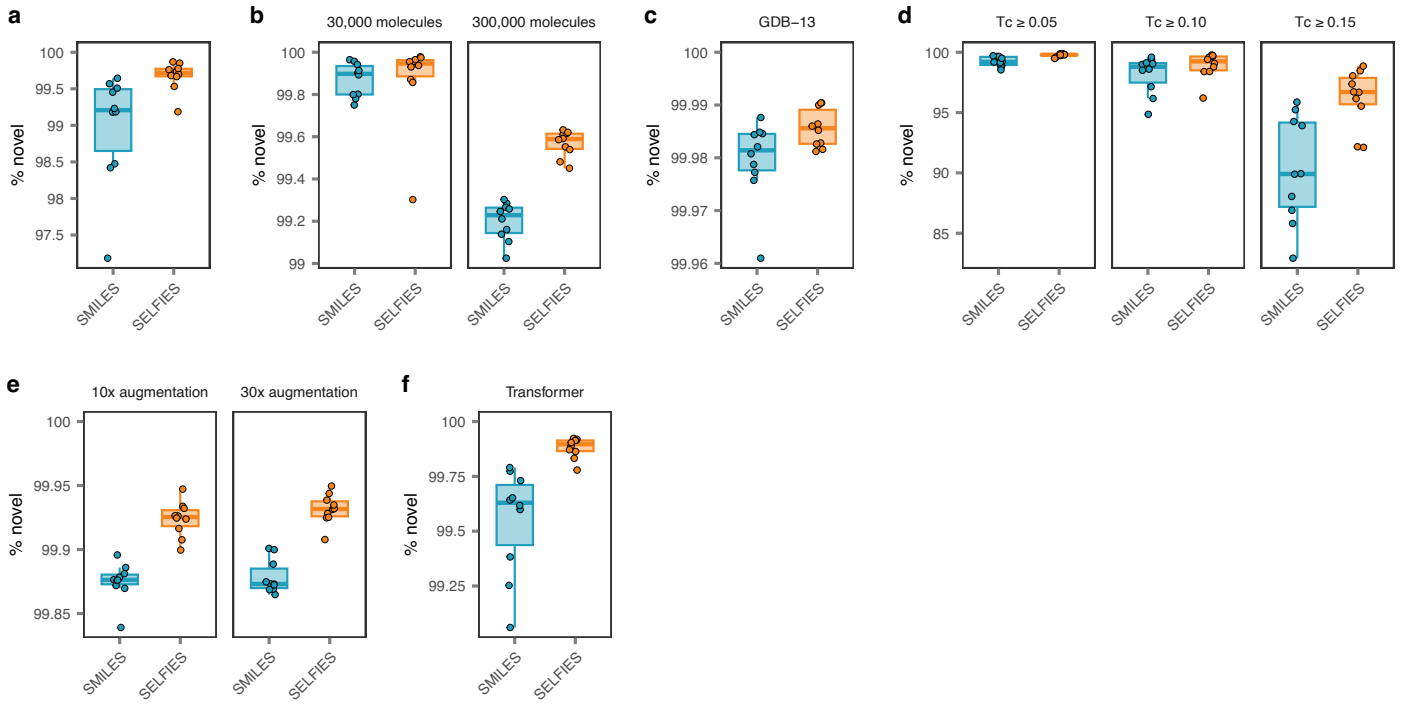
https://doi.org/10.1038/s42256-024-00821-x

Extended Data Fig. 1 | See next page for caption.

https://doi.org/10.1038/s42256-024-00821-x

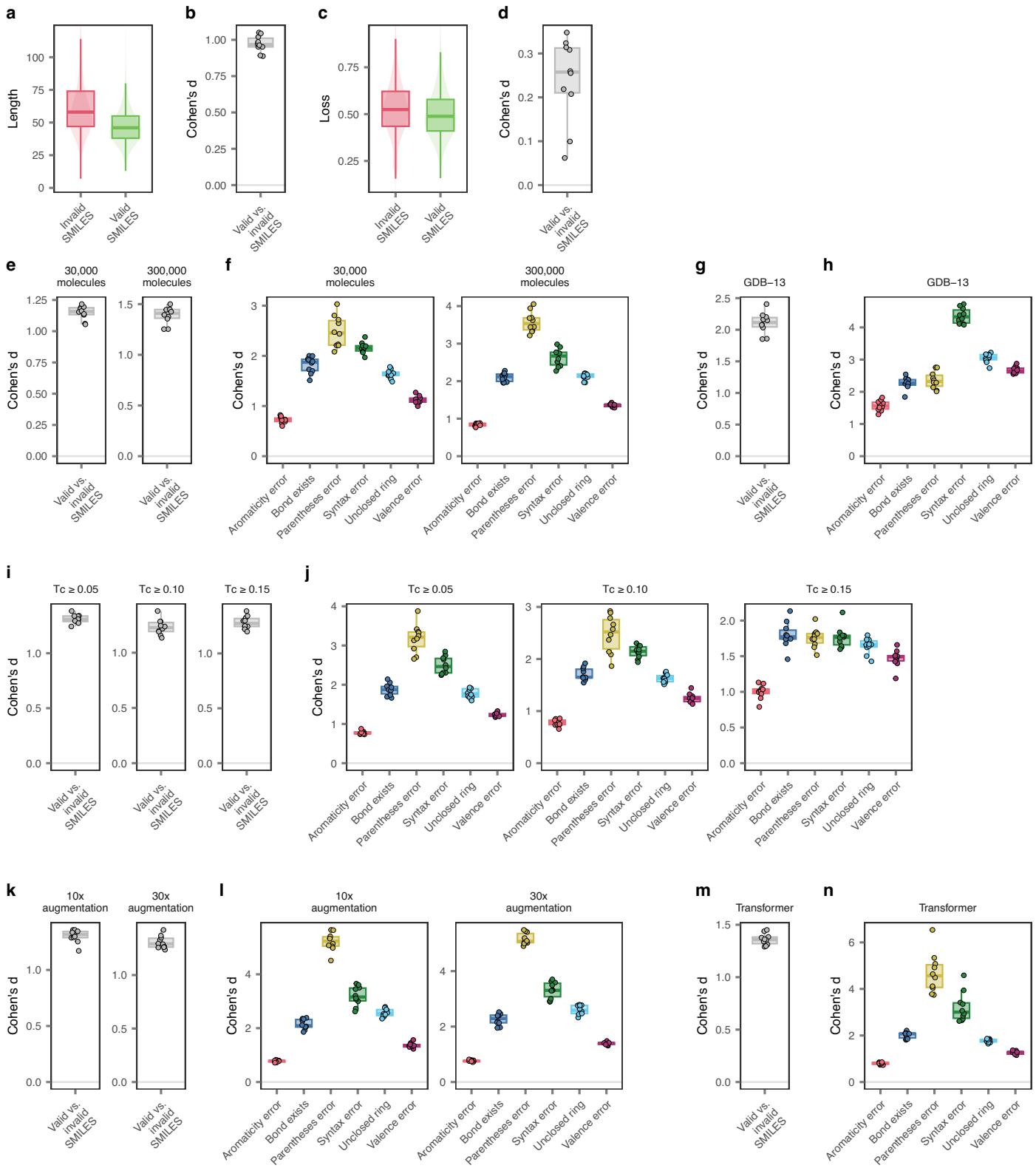
Extended Data Fig. 1 | Chemical language models trained on SMILES robustly outperform models trained on SELFIES. a, Jensen–Shannon distances between the Murcko scaffold compositions of generated and training molecules, shown for language models trained on SMILES versus SELFIES representations ($n = 10$ each; $p = 3.6 \times 10^{-6}$, paired t-test). **b**, Jensen–Shannon distances between the natural product-likeness scores of generated and training molecules, shown for language models trained on SMILES versus SELFIES representations ($n = 10$ each; $p = 1.8 \times 10^{-9}$, paired t-test). **c**, Jensen–Shannon distances between the fraction of atoms in each molecule that are stereocenters in generated and training molecules, shown for language models trained on SMILES versus SELFIES representations ($n = 10$ each; $p = 0.10$, paired t-test). **d**, PC1 scores integrating multiple distribution-learning metrics for language models trained on SMILES versus SELFIES representations (higher is better; $n = 10$ each; $p = 3.5 \times 10^{-8}$, paired t-test). **e**, Proportion of valid molecules generated by language models trained on SMILES versus SELFIES representations, for models trained on 30,000 or 300,000 molecules ($n = 10$ each; all $p \leq 6.7 \times 10^{-8}$, paired t-test). **f**, Fréchet ChemNet distances between generated and training molecules for language models trained on SMILES versus SELFIES representations, for models trained on 30,000 or 300,000 molecules ($n = 10$ each; all $p \leq 3.0 \times 10^{-8}$, paired t-test). **g**, PC1 scores integrating multiple distribution-learning metrics for language models trained on SMILES versus SELFIES representations, for models trained on 30,000 or 300,000 molecules ($n = 10$ each; all $p \leq 6.9 \times 10^{-6}$, paired t-test). **h**, As in **e**, but for models trained on molecules from the GDB-13 database ($p = 2.6 \times 10^{-9}$). **i**, As in **f**, but for models trained on molecules from the GDB-13 database ($p = 0.011$). **j**, As in **g**, but for models trained on molecules from the GDB-13 database ($p = 5.2 \times 10^{-4}$). **k**, As in **e**, but for models trained on sets of molecules with decreasing chemical diversity, as quantified by the minimum Tanimoto coefficient (Tc) from the seed molecule (all $p \leq 2.4 \times 10^{-7}$). **l**, As in **f**, but for models trained on sets of molecules with decreasing chemical diversity (all $p \leq 3.2 \times 10^{-7}$). **m**, As in **g**, but for models trained on sets of molecules with decreasing chemical diversity (all $p \leq 4.0 \times 10^{-7}$). **n**, As in **e**, but for models trained with data augmentation by non-canonical SMILES or SELFIES enumeration (all $p \leq 8.0 \times 10^{-11}$). **o**, As in **f**, but for models trained with data augmentation by non-canonical SMILES or SELFIES enumeration (all $p \leq 3.9 \times 10^{-7}$). **p**, As in **g**, but for models trained with data augmentation by non-canonical SMILES or SELFIES enumeration ($p \leq 8.0 \times 10^{-6}$). **q**, As in **e**, but for models based on the transformer architecture ($p = 3.1 \times 10^{-9}$). **r**, As in **f**, but for models based on the transformer architecture ($p = 1.7 \times 10^{-8}$). **s**, As in **g**, but for models based on the transformer architecture ($p = 7.9 \times 10^{-9}$).

https://doi.org/10.1038/s42256-024-00821-x



Extended Data Fig. 2 | Chemical language models trained on SMILES and SELFIES generate novel molecules at a high rate. a, Proportion of novel molecules generated by language models trained on SMILES versus SELFIES representations ($n = 10$ each; $p = 0.025$, paired t-test). **b**, As in **a**, but for models trained on 30,000 or 300,000 molecules ($p = 0.98$ and 7.8×10^{-6}, respectively). **c**, As in **a**, but for models trained on molecules from the GDB-13 database ($p = 0.078$). **d**, As in **a**, but for models trained on sets of molecules with decreasing chemical diversity ($p \leq 0.032$). **e**, As in **a**, but for models trained with data augmentation by non-canonical SMILES or SELFIES enumeration ($p \leq 2.5 \times 10^{-5}$). **f**, As in **a**, but for models based on the transformer architecture ($p = 0.0028$).

https://doi.org/10.1038/s42256-024-00821-x

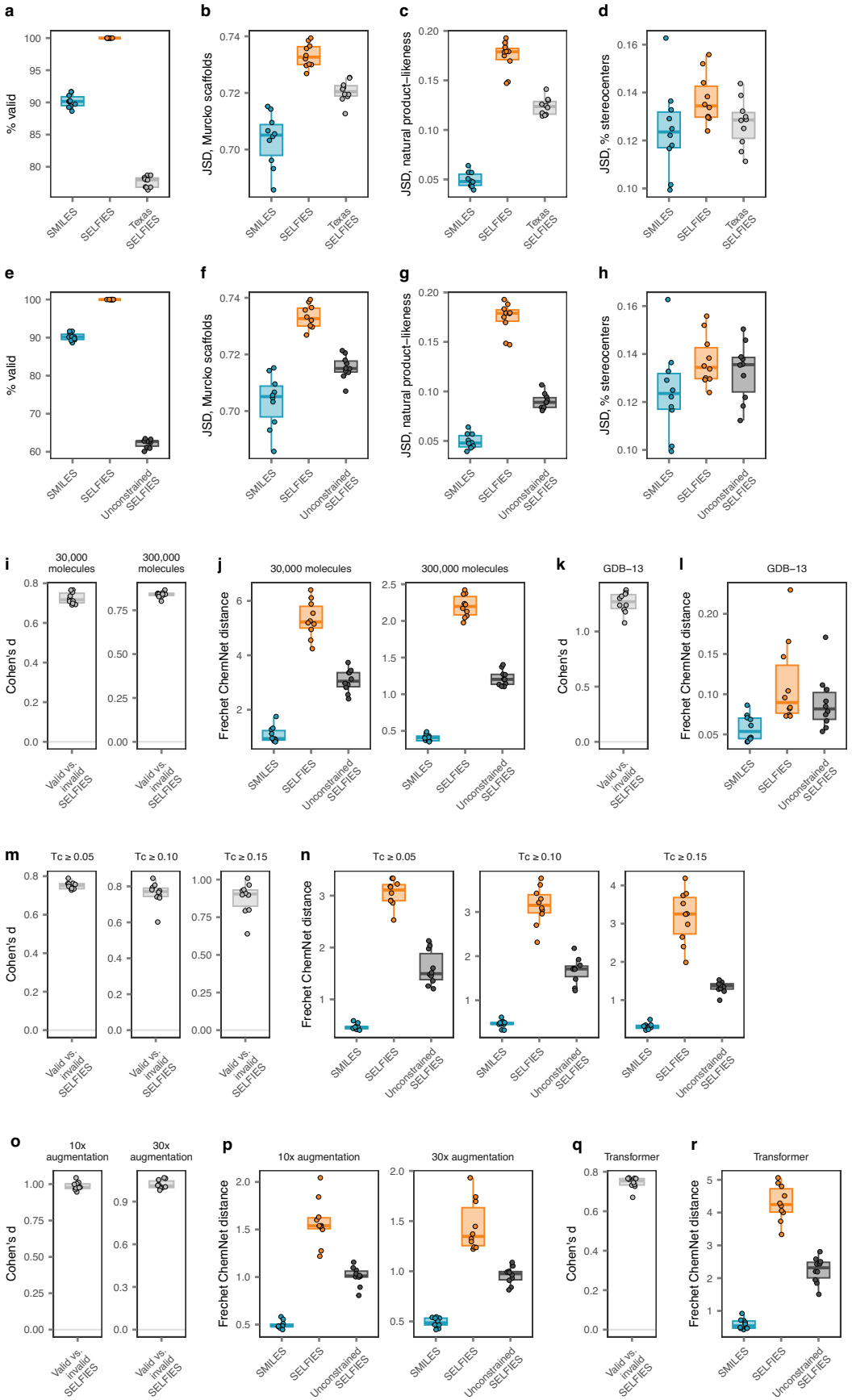


Extended Data Fig. 3 | See next page for caption.

https://doi.org/10.1038/s42256-024-00821-x

Extended Data Fig. 3 | Invalid SMILES are low-likelihood samples from chemical language models across architectures and training datasets. **a**, Lengths of valid versus invalid SMILES sampled from a representative chemical language model ($n = 10^7$ SMILES; $p < 10^{-15}$, two-sided t-test). **b**, Effect sizes (Cohen's *d*) comparing the lengths of valid versus invalid SMILES sampled from $n = 10$ chemical language models, demonstrating consistent effects ($p = 1.5 \times 10^{-13}$, two-sided one-sample t-test). **c**, Per-character losses of valid versus invalid SMILES sampled from a representative chemical language model ($n = 10^7$ SMILES; $p < 10^{-15}$, two-sided t-test). **d**, Effect sizes (Cohen's *d*) comparing per-character losses of valid versus invalid SMILES sampled from $n = 10$ chemical language models, demonstrating consistent effects ($p = 2.4 \times 10^{-5}$, two-sided one-sample t-test). **e**, Effect sizes (Cohen's *d*) comparing the losses of valid versus invalid SMILES sampled from $n = 10$ chemical language models trained on 30,000 or 300,000 molecules (all $p \leq 1.5 \times 10^{-12}$, two-sided t-test). **f**, Losses of valid SMILES versus invalid SMILES sampled from a representative chemical language model trained on 30,000 or 300,000 molecules, classified into six different categories based on RDKit error messages ($n = 10^7$ SMILES; all $p \leq 1.1 \times 10^{-9}$, two-sided one-sample t-test). **g**, As in **e**, but for models trained on molecules from the GDB-13 database ($p = 2.0 \times 10^{-11}$). **h**, As in **f**, but for models trained on molecules from the GDB-13 database (all $p \leq 4.5 \times 10^{-10}$). **i**, As in **e**, but for models trained on sets of molecules with decreasing chemical diversity, as quantified by the minimum Tanimoto coefficient (Tc) from the seed molecule (all $p \leq 1.1 \times 10^{-12}$). **j**, As in **f**, but for models trained on sets of molecules with decreasing chemical diversity (all $p \leq 4.3 \times 10^{-9}$). **k**, As in **e**, but for models trained with data augmentation by non-canonical SMILES or SELFIES enumeration (all $p \leq 1.2 \times 10^{-13}$). **l**, As in **f**, but for models trained with data augmentation by non-canonical SMILES or SELFIES enumeration (all $p \leq 4.3 \times 10^{-10}$). **m**, As in **e**, but for models based on the transformer architecture ($p = 4.3 \times 10^{-14}$). **n**, As in **f**, but for models based on the transformer architecture (all $p \leq 6.5 \times 10^{-8}$).

https://doi.org/10.1038/s42256-024-00821-x

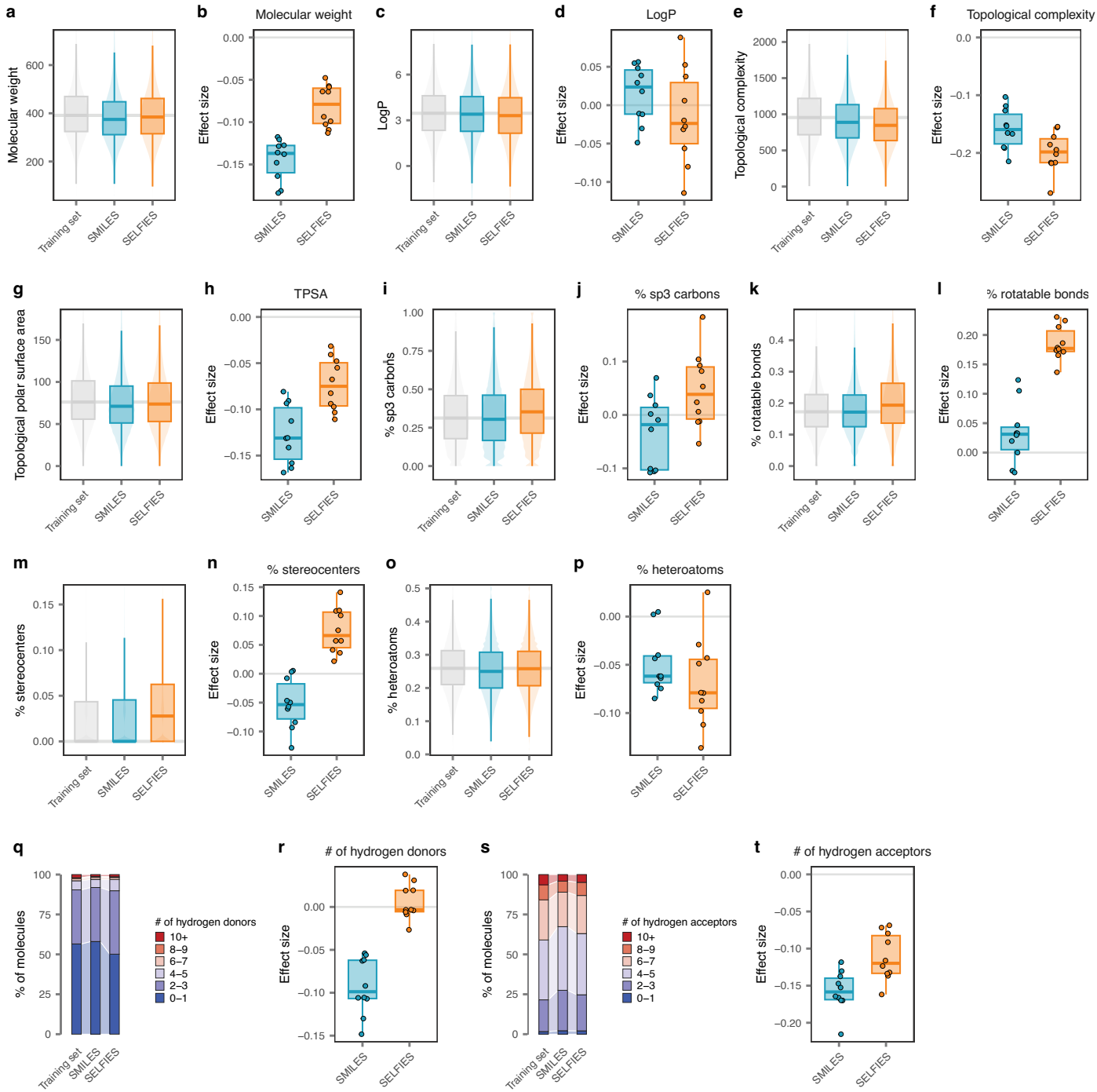


Extended Data Fig. 4 | See next page for caption.

https://doi.org/10.1038/s42256-024-00821-x

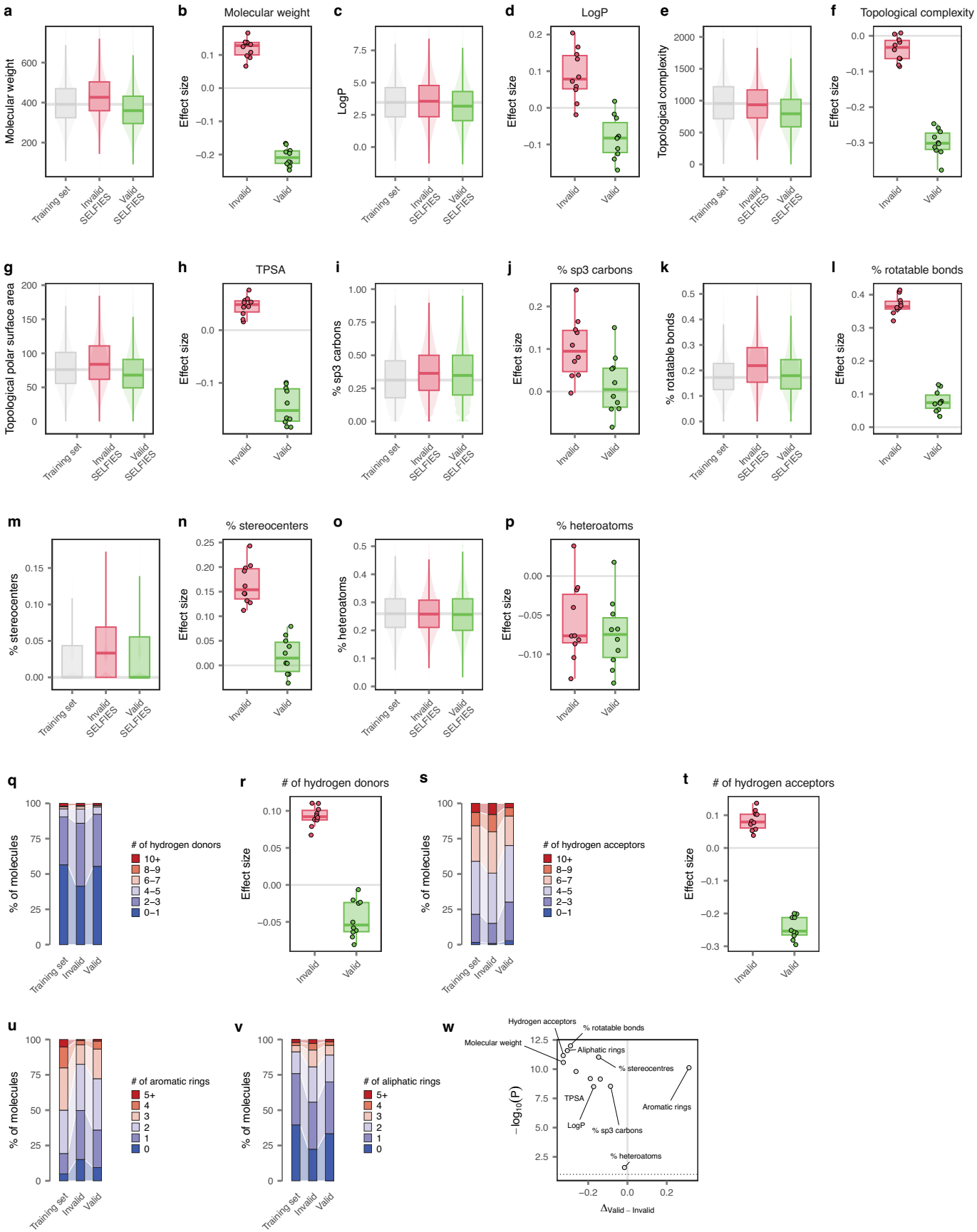
Extended Data Fig. 4 | Generating invalid outputs robustly improves the performance of chemical language models. a, Fréchet ChemNet distance between training and generated molecules for language models trained on SMILES or SELFIES representations, and with SELFIES valency constraints modified to allow pentavalent carbons ('Texas SELFIES'; $n = 10$ each; $p = 2.8 \times 10^{-14}$ compared to default valency constraints, paired t-test). **b**, Jensen–Shannon distances between the Murcko scaffold compositions of training and generated molecules for language models trained on SMILES or SELFIES representations, and with SELFIES valency constraints modified to allow pentavalent carbons ($n = 10$ each; $p = 7.6 \times 10^{-5}$ compared to default valency constraints, paired t-test). **c**, Jensen–Shannon distances between the natural product-likeness scores of training and generated molecules for language models trained on SMILES or SELFIES representations, and with SELFIES valency constraints modified to allow pentavalent carbons ($n = 10$ each; $p = 1.0 \times 10^{-5}$ compared to default valency constraints, paired t-test). **d**, Jensen–Shannon distances between the fraction of atoms that are stereocenters in training and generated molecules for language models trained on SMILES or SELFIES representations, and with SELFIES valency constraints modified to allow pentavalent carbons ($n = 10$ each; $p = 0.058$ compared to default valency constraints, paired t-test). **e-h**, As in **a-d**, but showing the removal of SELFIES valency constraints entirely ('unconstrained SELFIES'; $p = 2.6 \times 10^{-15}$, $p = 5.6 \times 10^{-16}$, $p = 8.5 \times 10^{-8}$, and $p = 0.38$, respectively). **i**, Effect sizes (Cohen's *d*) comparing the losses of valid versus invalid SELFIES sampled from $n = 10$ chemical language models trained on 30,000 or 300,000 molecules (all $p \leq 4.5 \times 10^{-14}$, two-sided one-sample t-test). **j**, Fréchet ChemNet distances between generated and training molecules for language models trained on SMILES versus SELFIES representations, and with SELFIES valency constraints disabled, for models trained on 30,000 or 300,000 molecules ($n = 10$ each; all $p \leq 3.2 \times 10^{-5}$, paired t-test). **k-l**, As in **i-j**, but showing language models trained on molecules from the GDB-13 database (losses, $p = 1.0 \times 10^{-11}$; Fréchet ChemNet distances, $p = 0.17$). **m-n**, As in **i-j**, but showing language models trained on sets of molecules with decreasing chemical diversity (losses, all $p \leq 8.7 \times 10^{-10}$; Fréchet ChemNet distances, all $p \leq 5.4 \times 10^{-6}$). **o-p**, As in **i-j**, but showing language models trained with data augmentation by non-canonical SMILES enumeration (losses, all $p \leq 3.3 \times 10^{-15}$; Fréchet ChemNet distances, all $p \leq 2.1 \times 10^{-4}$). **q-r**, As in **i-j**, but showing language models based on the transformer architecture (losses, $p = 5.3 \times 10^{-14}$; Fréchet ChemNet distances, $p = 8.8 \times 10^{-11}$).

https://doi.org/10.1038/s42256-024-00821-x



Extended Data Fig. 5 | Biased exploration of chemical space by language models trained on SMILES versus SELFIES. a, Molecular weights of molecules generated by chemical language models trained on SMILES versus SELFIES, as compared to the molecules in the training set (for valid molecules parsed from $n = 10^7$ sampled SMILES or SELFIES). Horizontal line shows the median molecular weight of molecules in the training set. **b**, Effect sizes (Cohen's *d*) comparing the molecular weights of generated molecules from chemical language models trained on SMILES versus SELFIES to the molecules in the training set ($n = 10$ each; $p = 2.9 \times 10^{-4}$, paired t-test). **c-d**, As in **a-b**, but showing octanol–water partition coefficients ($p = 0.24$). **e-f**, As in **a-b**, but showing topological complexity ($p = 3.1 \times 10^{-3}$). **g-h**, As in **a-b**, but showing topological polar surface area ($p = 4.0 \times 10^{-3}$). **i-j**, As in **a-b**, but showing the fraction of sp^3 carbons ($p = 8.3 \times 10^{-3}$). **k-l**, As in **a-b**, but showing the fraction of rotatable bonds ($p = 8.5 \times 10^{-5}$). **m-n**, As in **a-b**, but showing the fraction of stereocenters ($p = 1.0 \times 10^{-5}$). **o-p**, As in **a-b**, but showing the fraction of heteroatoms ($p = 0.31$). **q-r**, As in **a-b**, but showing the number of hydrogen donors ($p = 8.0 \times 10^{-5}$). **s-t**, As in **a-b**, but showing the number of hydrogen acceptors ($p = 2.4 \times 10^{-3}$).

https://doi.org/10.1038/s42256-024-00821-x

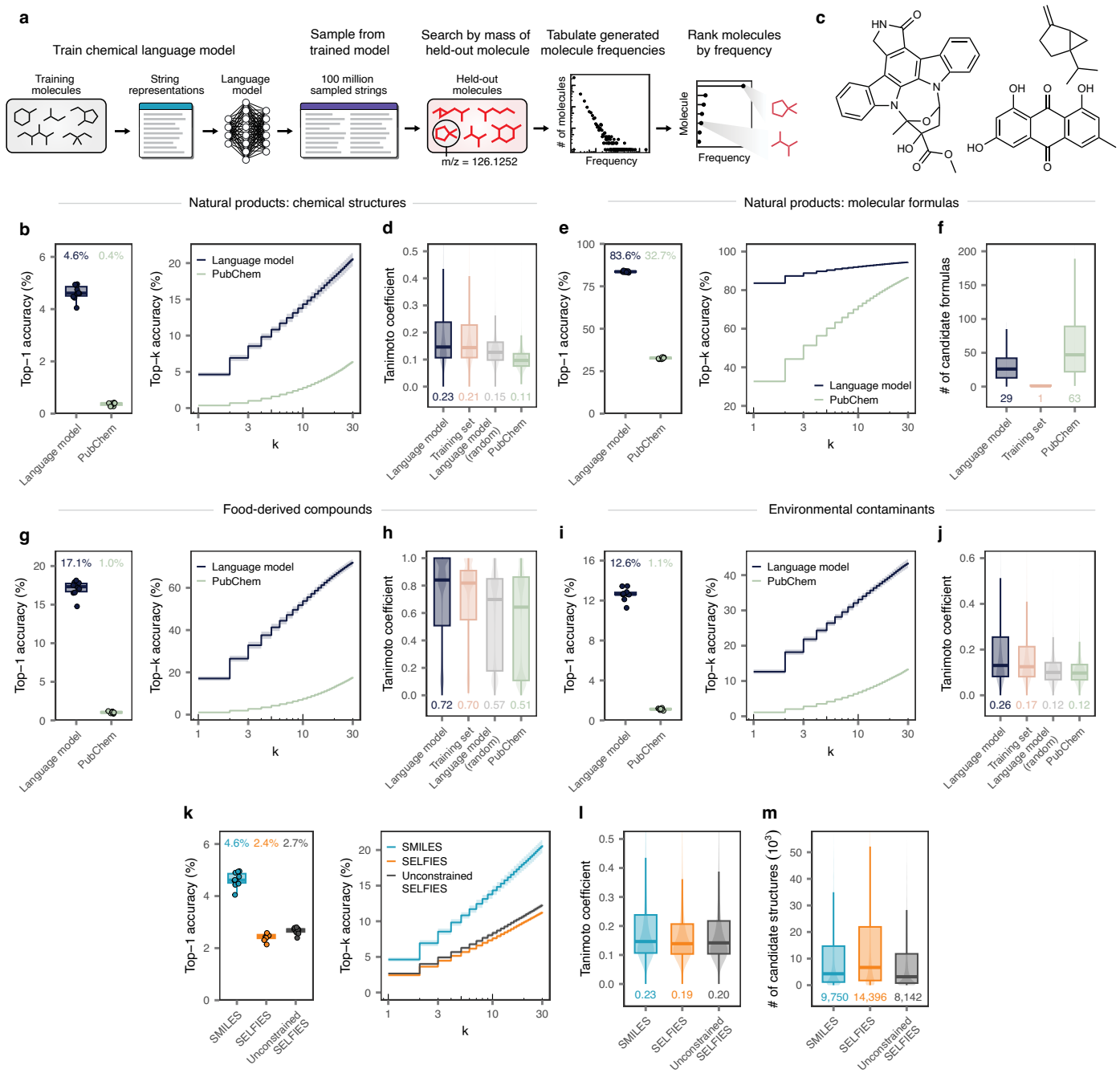


Extended Data Fig. 6 | See next page for caption.

https://doi.org/10.1038/s42256-024-00821-x

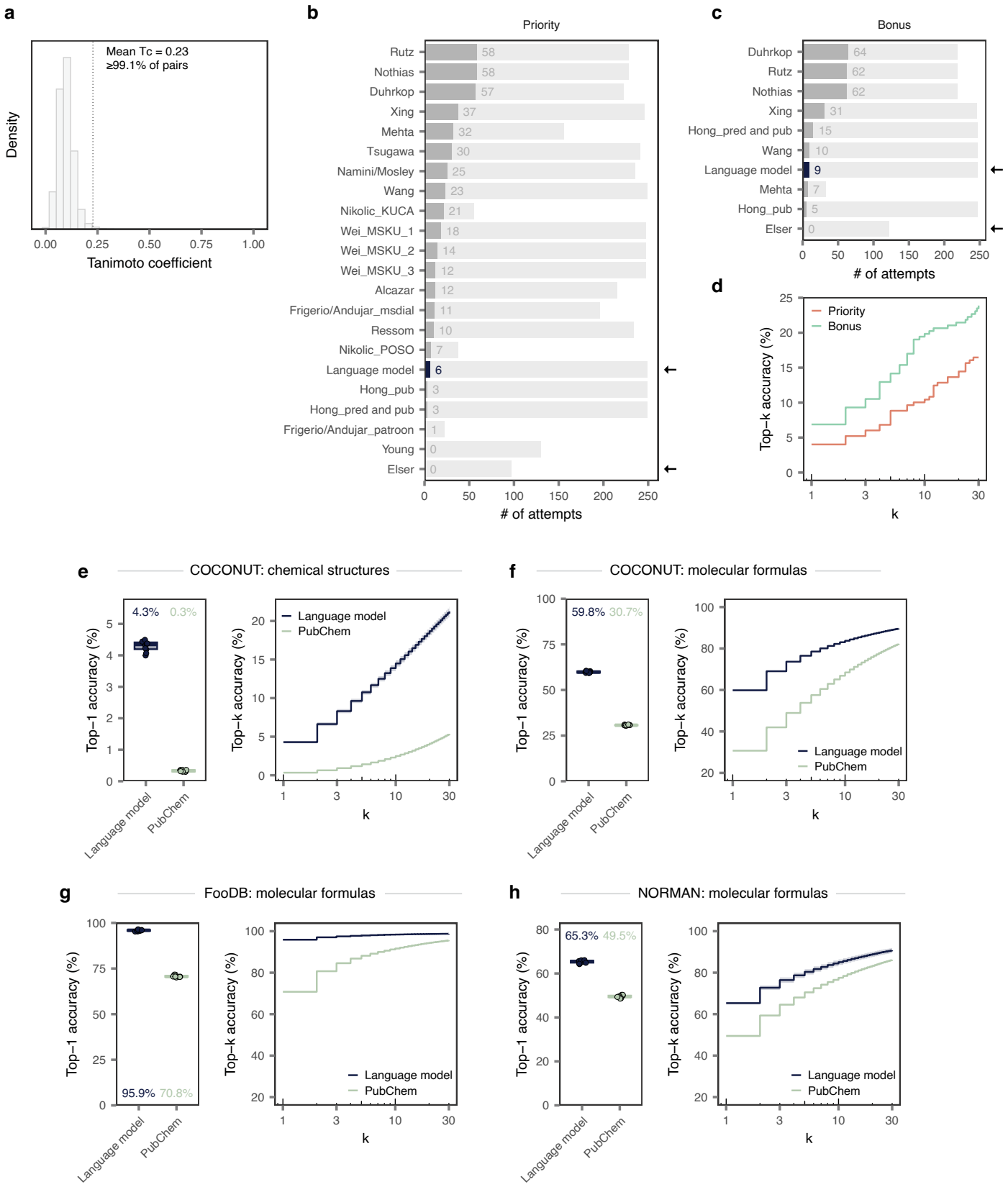
Extended Data Fig. 6 | Biased exploration of chemical space by language models trained on SELFIES with and without valency constraints. **a**, Molecular weights of molecules generated by chemical language models trained on SELFIES, shown separately for valid versus invalid SELFIES when parsing generated SELFIES without valency constraints, as compared to the molecules in the training set (for molecules parsed from $n = 10^7$ sampled SMILES or SELFIES). Horizontal line shows the median molecular weight of molecules in the training set. **b**, Effect sizes (Cohen's *d*) comparing the molecular weights of generated molecules from chemical language models trained on SELFIES to the molecules in the training set, shown separately for valid versus invalid SELFIES when parsing generated SELFIES without valency constraints ($n = 10$ each; $p = 2.7 \times 10^{-11}$, paired t-test). **c-d**, As in **a-b**, but showing octanol–water partition coefficients ($p = 3.2 \times 10^{-9}$). **e-f**, As in **a-b**, but showing topological complexity ($p = 1.6 \times 10^{-10}$). **g-h**, As in **a-b**, but showing topological polar surface area ($p = 6.4 \times 10^{-10}$). **i-j**, As in **a-b**, but showing the fraction of sp^3 carbons ($p = 2.9 \times 10^{-9}$). **k-l**, As in **a-b**, but showing the fraction of rotatable bonds ($p = 1.0 \times 10^{-12}$). **m-n**, As in **a-b**, but showing the fraction of stereocenters ($p = 9.4 \times 10^{-12}$). **o-p**, As in **a-b**, but showing the fraction of heteroatoms ($p = 0.026$). **q-r**, As in **a-b**, but showing the number of hydrogen donors ($p = 7.0 \times 10^{-10}$). **s-t**, As in **a-b**, but showing the number of hydrogen acceptors ($p = 6.9 \times 10^{-12}$). **u**, As in **a**, but showing the number of aromatic rings. **v**, As in **a**, but showing the number of aliphatic rings. **w**, Volcano plot showing differences in structural properties between molecules generated as valid versus invalid SELFIES when parsing generated SELFIES without valency constraints (statistical significance versus mean difference in effect size, paired t-test). Dotted line shows $p = 0.05$.

https://doi.org/10.1038/s42256-024-00821-x



Extended Data Fig. 7 | Language models facilitate structure elucidation of complex natural products. a, Experimental framework for structure elucidation of complex natural products from minimal analytical data (here, an accurate mass measurement) based on the sampling frequency of generated molecules. **b**, Top-1 accuracy, left, and top-k accuracy, right, of structure elucidation for held-out natural products by a chemical language model trained on the LOTUS database[63], as compared to searching by accurate mass in the PubChem database. Error bars and shaded areas show the mean and standard deviation across $n = 10$ cross-validation folds, respectively. **c**, Examples of natural product structures correctly elucidated by the language model. **d**, Tanimoto coefficients comparing chemical structures prioritized by the language model to the true held-out natural product ($n = 137{,}400$), as compared to searching by accurate mass in the PubChem database or the LOTUS training set, or a random molecule sampled from the output of the language model without regard to sampling frequency. **e**, As in **b**, but showing the accuracy of molecular formula annotation rather than full structure elucidation. **f**, Number of candidate molecular formulas proposed by the language model for each held-out natural product ($n = 137{,}400$), as compared to searching by accurate mass in PubChem or the LOTUS training set. **g**, As in **b**, but showing the accuracy of a language model trained on food-derived compounds from the FooDB database. **h**, As in **d**, but showing the Tanimoto coefficients of prioritized chemical structures from a language model trained on the FooDB database ($n = 52{,}880$ compounds). **i**, As in **b**, but showing the accuracy of a language model trained on environmental compounds from the NORMAN suspect list exchange. **j**, As in **d**, but showing the Tanimoto coefficients of prioritized chemical structures from a language model trained on the NORMAN dataset ($n = 63{,}732$ compounds). **k**, Top-1 accuracy, left, and top-k accuracy, right, of structure elucidation for held-out natural products by chemical language models trained on the LOTUS database, represented as SMILES or SELFIES. For models trained on SELFIES, results are shown separately for structures parsed with or without valency constraints ('unconstrained SELFIES'). **l**, Tanimoto coefficients comparing chemical structures prioritized by the language model to the true held-out natural product, for the same three language model variants as in **k** ($n = 137{,}400$ compounds). **m**, Number of candidate structures generated for each held-out natural product by chemical language models, for the same three language model variants as in **k** ($n = 137{,}400$ compounds).

https://doi.org/10.1038/s42256-024-00821-x

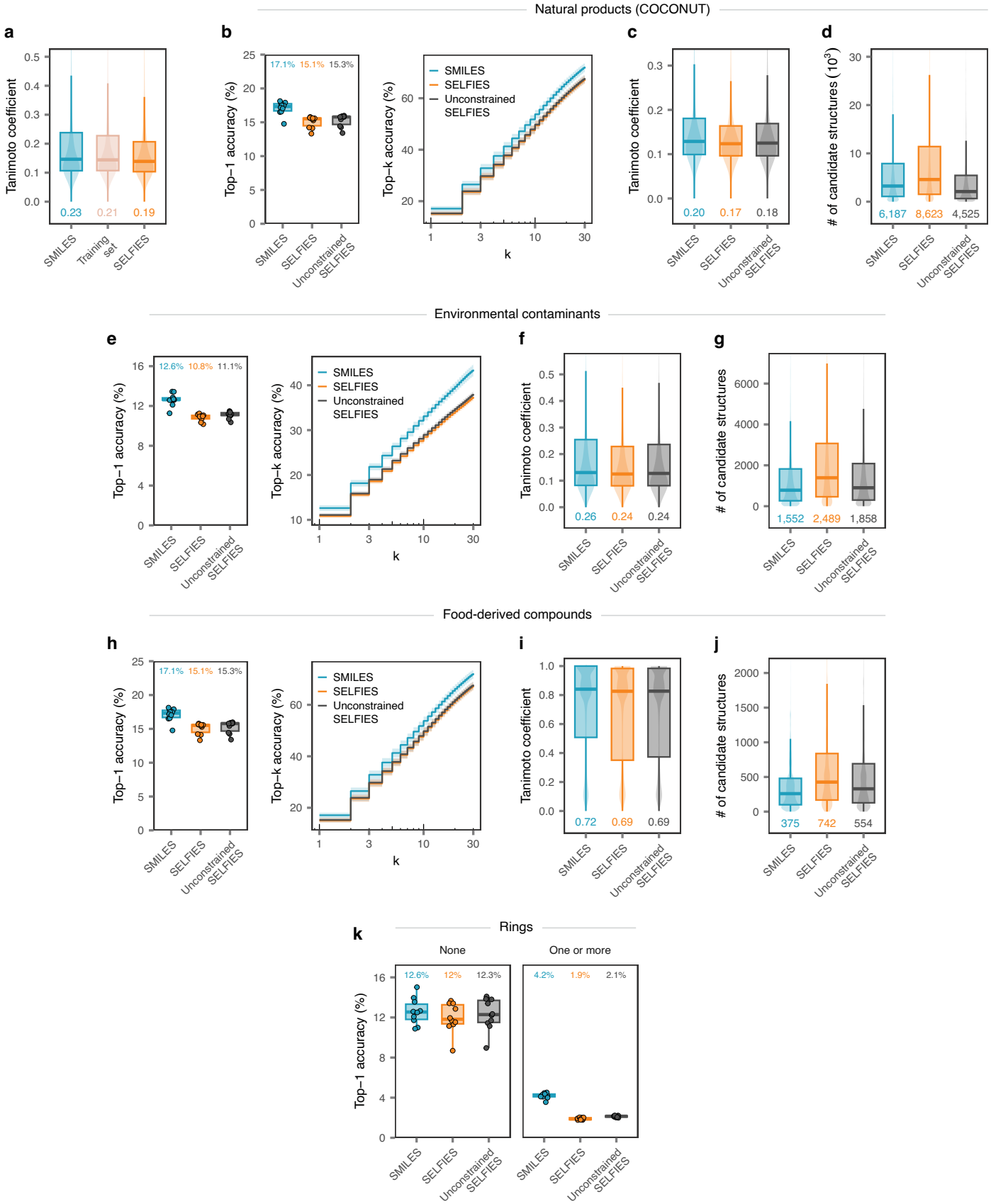


Extended Data Fig. 8 | See next page for caption.

https://doi.org/10.1038/s42256-024-00821-x

Extended Data Fig. 8 | Language models contribute to structure elucidation of unknown molecules. a, Distribution of Tanimoto coefficients between random pairs of molecules from the LOTUS database of natural products. Vertical line shows the mean Tanimoto coefficient comparing chemical structures prioritized by the language model to the true held-out natural product, greater than 99.1% of pairs. **b**, Accuracy of a chemical language model trained in cross-validation on the LOTUS natural product database versus 19 submissions to the CASMI 2022 'priority' competition (dark grey and coloured bars, number of correct solutions; light grey bars, number of attempts). Arrows highlight *de novo* structure elucidation methods. **c**, As in **b**, but showing the CASMI 2022 'bonus' competition. **d**, Top-k accuracy of a chemical language model trained in cross-validation on the LOTUS database in the CASMI 'priority' and 'bonus' competitions. **e**, Top-1 accuracy, left, and top-k accuracy, right, of structure elucidation for held-out natural products by a chemical language model trained on the COCONUT database of natural products, as compared to searching by accurate mass in PubChem. Error bars and shaded areas show the standard deviation across $n = 10$ cross-validation folds. **f**, As in **e**, but showing the accuracy of molecular formula annotation rather than full structure elucidation. **g**, As in **f**, but showing molecular formula annotation for food-derived compounds from the FooDB database. **h**, As in **f**, but showing molecular formula annotation for environmental compounds from the NORMAN suspect list exchange.

https://doi.org/10.1038/s42256-024-00821-x

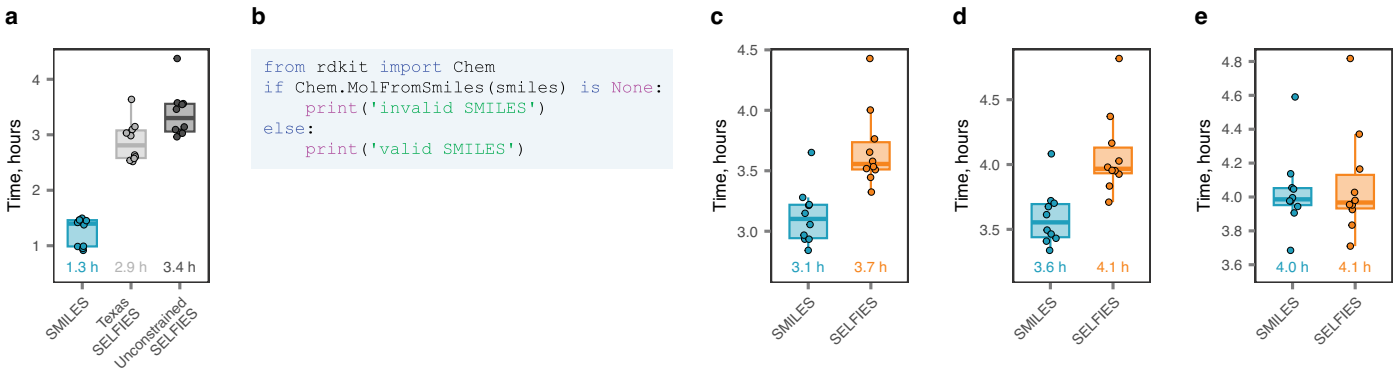


Extended Data Fig. 9 | See next page for caption.

https://doi.org/10.1038/s42256-024-00821-x

Extended Data Fig. 9 | Generating and discarding invalid outputs improves structure elucidation. a, Tanimoto coefficients comparing chemical structures prioritized by the language model to true held-out natural products (n = 137,400) from the LOTUS database, for language models trained on SMILES versus SELFIES, as compared to searching by accurate mass in the training set. **b**, Top-1 accuracy, left, and top-k accuracy, right, of structure elucidation for held-out natural products by n = 10 chemical language models trained on the COCONUT database, represented as SMILES or SELFIES. For models trained on SELFIES, results are shown separately for structures parsed with or without valency constraints ('unconstrained SELFIES'). **c**, Tanimoto coefficients comparing chemical structures prioritized by the language model to the true held-out natural product (n = 405,947), for the same three language model variants shown in **b**. **d**, Number of candidate structures generated for each held-out natural product (n = 405,947) by chemical language models, for the same three language model variants as in **b**. **e-g**, As in **b-d**, but for food-derived compounds from the NORMAN suspect list exchange. **h-j**, As in **b-d**, but for environmental compounds from the FooDB database. **k**, Top-1 accuracy of structure elucidation for held-out natural products by n = 10 chemical language models trained on the LOTUS database, represented as SMILES or SELFIES, shown separately for acyclic natural products versus natural products containing at least one ring system (p = 0.016, two-way ANOVA).

https://doi.org/10.1038/s42256-024-00821-x



Extended Data Fig. 10 | Discarding invalid SMILES is fast and easy. a, Time (in hours) required to parse $n = 10$ samples of 10 million SMILES or SELFIES strings and discard invalid outputs. For SELFIES, results are shown only when parsing SELFIES with valency constraints that allow for invalid output generation. **b**, Python code snippet demonstrating the ease with which it can be determined if a SMILES string is invalid. **c**, Time (in hours) required to train $n = 10$ chemical language models on samples of 100,000 molecules from the ChEMBL database, represented as SMILES versus SELFIES. Training was performed on Dell EMC C4140 GPU compute nodes equipped with NVIDIA Tesla V100 GPUs. Language models trained on SELFIES required an average of 0.6 h longer to train, relative to models trained on SMILES representations of the same molecules. **d**, As in **c**, but showing the total time (in hours) required to train chemical language models, sample 500,000 outputs from the trained models, and discard invalid outputs. **e**, As in **c**, but showing the total time (in hours) required to train chemical language models, sample 4 million outputs from the trained models, and discard invalid outputs. Filtering invalid SMILES would not increase computational requirements, relative to a model trained on SELFIES, until approximately 4 million molecules.