Polymer informatics is a nascent field that allows access to the depth of the polymer universe and demonstrates the potency of machine learning (ML) models to overcome this challenge. ML frameworks have enabled substantial progress in the development of polymer property predictors and solving inverse problems in which polymers that meet specific property requirements are either identified from candidate sets or are freshly designed using genetic or generative algorithms.

An essential step in polymer informatics pipelines is the conversion of polymer chemical structures to numerical representations that are often called fingerprints, features, or descriptors (see blue boxes in Fig. 1a). Past handcrafted fingerprinting approaches allow access to the depth of the polymer universe and demonstrate the potency of machine learning (ML) models to overcome this challenge. ML frameworks have enabled substantial progress in the development of polymer property predictors and solving inverse problems in which polymers that meet specific property requirements are either identified from candidate sets or are freshly designed using genetic or generative algorithms.

An essential step in polymer informatics pipelines is the conversion of polymer chemical structures to numerical representations that are often called fingerprints, features, or descriptors (see blue boxes in Fig. 1a). Past handcrafted fingerprinting approaches allow access to the depth of the polymer universe and demonstrate the potency of machine learning (ML) models to overcome this challenge. ML frameworks have enabled substantial progress in the development of polymer property predictors and solving inverse problems in which polymers that meet specific property requirements are either identified from candidate sets or are freshly designed using genetic or generative algorithms.

An essential step in polymer informatics pipelines is the conversion of polymer chemical structures to numerical representations that are often called fingerprints, features, or descriptors (see blue boxes in Fig. 1a). Past handcrafted fingerprinting approaches allow access to the depth of the polymer universe and demonstrate the potency of machine learning (ML) models to overcome this challenge. ML frameworks have enabled substantial progress in the development of polymer property predictors and solving inverse problems in which polymers that meet specific property requirements are either identified from candidate sets or are freshly designed using genetic or generative algorithms.

An essential step in polymer informatics pipelines is the conversion of polymer chemical structures to numerical representations that are often called fingerprints, features, or descriptors (see blue boxes in Fig. 1a). Past handcrafted fingerprinting approaches allow access to the depth of the polymer universe and demonstrate the potency of machine learning (ML) models to overcome this challenge. ML frameworks have enabled substantial progress in the development of polymer property predictors and solving inverse problems in which polymers that meet specific property requirements are either identified from candidate sets or are freshly designed using genetic or generative algorithms.

An essential step in polymer informatics pipelines is the conversion of polymer chemical structures to numerical representations that are often called fingerprints, features, or descriptors (see blue boxes in Fig. 1a). Past handcrafted fingerprinting approaches allow access to the depth of the polymer universe and demonstrate the potency of machine learning (ML) models to overcome this challenge. ML frameworks have enabled substantial progress in the development of polymer property predictors and solving inverse problems in which polymers that meet specific property requirements are either identified from candidate sets or are freshly designed using genetic or generative algorithms.

An essential step in polymer informatics pipelines is the conversion of polymer chemical structures to numerical representations that are often called fingerprints, features, or descriptors (see blue boxes in Fig. 1a). Past handcrafted fingerprinting approaches allow access to the depth of the polymer universe and demonstrate the potency of machine learning (ML) models to overcome this challenge. ML frameworks have enabled substantial progress in the development of polymer property predictors and solving inverse problems in which polymers that meet specific property requirements are either identified from candidate sets or are freshly designed using genetic or generative algorithms.

An essential step in polymer informatics pipelines is the conversion of polymer chemical structures to numerical representations that are often called fingerprints, features, or descriptors (see blue boxes in Fig. 1a). Past handcrafted fingerprinting approaches allow access to the depth of the polymer universe and demonstrate the potency of machine learning (ML) models to overcome this challenge. ML frameworks have enabled substantial progress in the development of polymer property predictors and solving inverse problems in which polymers that meet specific property requirements are either identified from candidate sets or are freshly designed using genetic or generative algorithms.
a fully end-to-end machine-driven polymer informatics pipeline that uses and unleashes the true power of artificial intelligence methods. Multitask deep neural networks harness inherent correlations in multifidelity and multi-property data sets, scale effortlessly in cloud computing environments, and generalize to multiple prediction tasks.

Recent studies demonstrated the benefits of using Transformers in the molecule chemical space. For example, Wang et al. have trained a BERT model (the most common general language model) with a data set of molecule SMILES strings. Using BERT’s latent space representations of molecules as fingerprints, the authors show that their approach outperforms other fingerprinting methods (including fingerprints of an unsupervised recurrent neural network and a graph neural network). Similarly, Schwaller et al. have developed a Transformer model to predict retrosynthesis pathways of molecules from reactants and reagents that outperforms known algorithms in the reaction prediction literature. A very recent study by Xu et al. (performed almost at the same time as us, as can be confirmed by both our arXiv submissions) used a RoBERTa model (an evolution of the BERT Transformer model) for polymer property predictions. Their training strategy first involves the pretraining (unsupervised training) of the RoBERTa model using 5 million polymers and then a finetuning step (supervised training) to directly predict polymer properties. Although their work uses much smaller datasets than ours both for unsupervised and supervised training tasks, they find that their finetuned RoBERTa model outperforms graph neural networks, long short-term memory and other models; we do note that this recent work does not make direct comparisons of their Transformer-based model with the current state-of-the-art in hand-crafted fingerprinting and multi-task learning (which we do in the present contribution).

Another promising neural network architecture, namely, graph neural networks, which treats chemical structures as graphs, has been applied to the molecule and polymer chemical space in the past. In contrast to Transformers, graph neural networks represent atoms as nodes and bonds as edges of a graph, so as to encode immediate and extended connectivities between atoms. As a consequence, graph neural networks are not directly based on PSMILES strings like Transformers, but depend on an initial set of feature vectors (such as atom types, implicit valence, etc.) that need to be computed for and assigned to each node. For example, Park et al. compared predictions of a graph convolutional network and the popular extended-connectivity circular fingerprint method. Second, re-assembling the BRICS fragments in many different ways generates 100 million hypothetical polymers by randomly and enumeratively combing the fragments.
consequence of this is that in the case of Transformer-based approaches, the learned fingerprint is independent of the polymer properties (and so can be determined once and for all), where as graph neural network architectures are typically constructed such that the learned representations depend on the specific property under consideration.

We note that self-supervised graph neural networks\(^7\) have recently been developed that learn the molecule graph through atom, bond, and subgraph masking, an approach similar to Transformers.

This work has several critical ingredients. First, we generate a data set of 100 million hypothetical polymers by enumeratively combining chemical fragments extracted from a list of more than 13,000 synthesized polymers. Next, we train polyBERT, a DeBERTa\(^{39}\) based encoder-only Transformer, using this hypothetical polymer data set to become a polymer chemical linguist. During training, polyBERT learns to translate input PSMILES strings to numerical representations that we use as polymer fingerprints. Finally, we map the polyBERT fingerprints to about 3 dozen polymer properties using our multitask ML framework to yield fully machine-driven ultrafast polymer property predictors. For benchmarking, the performance (both accuracy and speed) of this new end-to-end property prediction pipeline is compared with the state-of-the-art handcrafted Polymer Genome\(^8\) (PG) fingerprint based pipeline pioneered previously. Using the ultrafast polyBERT polymer informatics pipeline, we are in a position to predict the properties of the 100 million hypothetical polymers intending to find property boundaries of the polymer universe. This work contributes to expediting the discovery, design, development, and deployment of polymers by harnessing the true power of language, data, and artificial intelligence models.

**Results**

**Data sets**

Figure 1c sketches the two-step process for fabricating 100 million hypothetical PSMILES strings. We use the Breaking Retrosynthetically Interesting Chemical Substructures (BRICS)\(^{32}\) method (as implemented in RDKit\(^5\)) to decompose previously synthesized 13,766 polymers (all monomers of the data set outline in Table 1, see below) into 4,424 unique chemical fragments. Random and enumerative compositions of these fragments yield 100 million hypothetical PSMILES strings that we first canonicalize (see “Methods” section) and then use for training polyBERT. The hypothetical PSMILES strings are chemically valid polymers but, mostly, have never been synthesized before.

Once polyBERT has completed its unsupervised learning task using the 100 million hypothetical PSMILES strings, multitask supervised learning maps polyBERT polymer fingerprints to multiple properties to produce property predictors. We use the property data set in Table 1 for training the property predictors. The data set contains 28,061 (≈80%) homopolymer and 7,456 (≈20%) copolymer (total of 35,517) data points of 29 experimental and computational polymer properties that pertain to 11,145 different monomers and 1338 distinct copolymer chemistries, respectively. Each of the 7,456 copolymer data points involves two distinct comonomers at various compositions. Our copolymer data points are for random copolymers, which are adequately handled by our adopted fingerprinting strategy (see “Methods” section). Alternating copolymers are treated as homopolymers with appropriately defined repeat units for fingerprinting purposes. Other flavors of copolymers may also be encoded by adding additional fingerprint components. All data points in the data set have been used in past studies\(^10,13,22,49\) and were produced using computational methods or obtained from literature and other public sources. Supplementary Figs. S3–S8 show histograms for each property.

polyBERT

polyBERT iteratively ingests 100 million hypothetical PSMILES strings to learn the polymer chemical language, as sketched in Fig. 1b. Using 100 million PSMILES strings is the latest example of training a chemistry-related language model with a large data set and follows the trend of growing data sets in this discipline, with ChemBERTa using 10 million, SMILES-BERT using 18.7 million, and ChemBERTa-2 using 77 million SMILES strings.\(^40\) polyBERT is a DeBERTa\(^{39}\) model (as implemented in Huggingface’s Transformer Python library\(^4\)) with a supplemental three-stage preprocessing unit for PSMILES strings. We chose the DeBERTa model as the foundation of polyBERT because it outperformed other BERT-like models (BERT, RoBERTa\(^{32}\), and DistillBERT\(^{30}\)) in our tests (see Supplementary Discussion) and standardized performance task\(^30\). First, polyBERT transforms a input PSMILES string into its canonical form (e.g., \(\text{[CCOCCO]} \rightarrow \text{CCOC}\)) using the canonicalize_psmiles Python package developed in this work. Details can be found in the Methods section. Second, polyBERT tokenizes canonical PSMILES strings using the SentencePiece\(^2\) tokenizer and a total of 265 tokens. The tokens include common PSMILES characters such as the upercased and lowercased \(\text{PS}\) elements of the periodic table of elements, numbers ranging from 0 to 9, and special characters like \((+\), (,), (−), (=), among others. This ensures that the tokenizer covers the entire PSMILES strings vocabulary and is a similar approach to that in ref. \(^{50}\). A full token list can be found at the GitHub repository (see the Data and Code Availability section). Third, polyBERT masks \(\approx 15\%\) (default parameter for masked language models) of the tokens to create a self-supervised training task. In this training task, polyBERT is taught to predict the masked tokens using the non-masked surrounding tokens by adjusting the weights of the Transformer encoders (fill-in-the-blanks task). We use 80 million PSMILES strings for training and 20 million PSMILES strings for validation. The validation F1-score is \(> 0.99\). This exceptionally good F1-score indicates that polyBERT finds the masked tokens in almost all cases. The total CO\(_2\) emissions for training polyBERT on our hardware are estimated to be 12.6 kgCO\(_2\)-eq (see CO\(_2\) Emission and Timing section).

The training with 80 million PSMILES strings renders polyBERT an expert polymer chemical linguist who knows grammatical and syntactical rules of the polymer chemical language. polyBERT learns patterns and relations of tokens via the multi-head self-attention mechanism and fully connected feed-forward network of the Transformer encoders\(^3\). The attention mechanism instructs polyBERT to devote more focus to a small but essential part of a PSMILES string, polyBERT’s learned latent spaces after each encoder block are numerical representations of the input PSMILES strings. The polyBERT fingerprint is the average over the token dimension (sentence average) of the last latent space (dotted line in Fig. 1b). We use the Python package SentenceTransformers\(^2\) for extracting and computing polyBERT fingerprints.

Fingerprints

For acquiring analogies and juxtaposing chemical relevancy, we compare polyBERT fingerprints with the handcrafted Polymer Genome\(^8\) (PG) fingerprints that numerically encode polymers at three different length scales. A description of PG fingerprints can be found in “Methods” section. The PG fingerprint vector for the data set in this work has 945 components and is sparsely populated (93.9% zeros). The reason for this ultra sparsity is that many PG fingerprint components count chemical groups in polymers\(^8\). A fingerprint component of zero indicates that a chemical group is not present. In contrast, polyBERT fingerprint vectors have 600 components and are fully dense (0% zeros). Fully dense and lower-dimensional fingerprints are often advantageous for ML models whose computation time scales superlinear (\(O(n^2), n \geq 1\)) with the data set size (\(n\)) such as Gaussian process or kernel ridge techniques. Moreover, in the case of neural networks, sparse and high-dimensional input vectors can cause unnecessary high memory load that reduces training and inference speed. We note that the dimensionality of polyBERT fingerprints is a parameter that can be
chosen arbitrarily to yield the best training result. A summary of the key figures can be found in Supplementary Table S4.

Figure 2 shows Uniform Manifold Approximation and Projection (UMAP) plots for all homo- and copolymer chemistries in Table 1. The colored triangles in the first column indicate the coordinates of three selected polymers for polyBERT and PG fingerprints. We observe for both fingerprint types that the orange and blue triangles are very close, while the green triangle is separate. We also note that polymers corresponding to the orange and blue triangles, namely poly(but-1-ene) and poly(pent-1-ene), have similar chemistry (different by only one carbon atom), but poly(4-vinylpyridine) represented by a green triangle, is different. This chemically intuitive positioning of fingerprints suggests the chemical relevancy of fingerprint distances. The cosine fingerprint distances reported in Supplementary Fig. S1 allow for the same conclusion.

The second, third, and fourth columns of Fig. 2 display the same UMAP plots as in the first column. Colored dots indicate the property values of $T_g$, $T_m$, and $E_{gr}$, while light gray dots show polymer fingerprints with unknown property values. We observe localized clusters of similar color in each plot pertaining to polymers of similar properties. Although this finding is not surprising for the PG fingerprint because it relies on handcrafted chemical features that purposely position similar polymers next to each other, it is remarkable for polyBERT. With no chemical information and purely based on training on a massive amount of PSMILES strings, polyBERT has learned polymer chemical information and purely based on training on a massive amount of PSMILES strings, polyBERT has learned polymer chemical information and purely based on training on a massive amount of PSMILES strings, polyBERT has learned polymer chemical information and purely based on training on a massive amount of PSMILES strings, polyBERT has learned polymer chemical information and purely based on training on a massive amount of PSMILES strings, polyBERT has learned polymer chemical information and purely based on training on a massive amount of PSMILES strings, polyBERT has learned polymer chemical information and purely based on training on a massive amount of PSMILES strings, polyBERT has learned polymer chemical information and purely based on training on a massive amount of PSMILES strings, polyBERT has learned polymer chemical information and purely based on training on a massive amount of PSMILES strings, polyBERT has learned polymer chemical information and purely based on training on a massive amount of PSMILES strings, polyBERT has learned polymer chemical information and purely based on training on a massive amount of PSMILES strings, polyBERT has learned polymer chemical information and purely based on training on a massive amount of PSMILES strings.

### Table 1 | Training data set for the property predictors. The properties are sorted into categories, showed at the top of each block. The data set contains 29 properties (dielectric constants $k_f$ are available at 9 different frequencies $f$). HP and CP stand for homopolymer and copolymer, respectively.

| Property                           | Symbol | Unit          | Source$^a$ | Data range               | Data points |
|-----------------------------------|--------|---------------|------------|--------------------------|-------------|
|                                    |        |               |            | HP                       | CP          | All          |
| **Thermal**                       |        |               |            |                          |             |
| Glass transition temp.             | $T_g$  | K             | Exp.       | [8e+01, 9e+02]           | 5163        | 3312         | 8495         |
| Melting temp.                      | $T_m$  | K             | Exp.       | [2e+02, 9e+02]           | 2132        | 1523         | 3655         |
| Degradation temp.                  | $T_d$  | K             | Exp.       | [3e+02, 1e+03]           | 3584        | 1064         | 4648         |
| **Thermodynamic & physical**       |        |               |            |                          |             |
| Heat capacity                      | $c_p$  | Jg $^{-1}$ K$^{-1}$ | Exp.       | [8e-01, 2e+00]         | 79          | 79           |             |
| Atomization energy                 | $E_{at}$ | eV atom$^{-1}$ | DFT        | [-7e+00, -5e+00]         | 390         | 390          |             |
| Limiting oxygen index              | $O_l$  | %             | Exp.       | [1e+01, 7e+01]           | 101         | 101          |             |
| Crystallization tendency (DFT)     | $X_c$  | %             | DFT        | [1e-01, 1e+02]           | 432         | 432          |             |
| Crystallization tendency (exp.)    | $X_c$  | %             | Exp.       | [1e+00, 1e+02]           | 111         | 111          |             |
| Density                            | $\rho$ | g cm$^{-3}$   | Exp.       | [8e-01, 2e+00]           | 910         | 910          |             |
| **Electronic**                     |        |               |            |                          |             |
| Band gap (chain)                   | $E_{gc}$ | eV            | DFT        | [2e-02, 1e+01]           | 4224        | 4224         |             |
| Band gap (bulk)                    | $E_{gb}$ | eV            | DFT        | [4e-01, 1e+01]           | 597         | 597          |             |
| Electron affinity                  | $E_{ea}$ | eV            | DFT        | [4e-01, 5e+00]           | 368         | 368          |             |
| Ionization energy                  | $E_i$  | eV            | DFT        | [4e+00, 1e+01]           | 370         | 370          |             |
| Electronic injection barrier       | $E_{eb}$ | eV            | DFT        | [2e+00, 7e+00]           | 2610        | 2610         |             |
| Cohesive energy density            | $\delta$ | cal cm$^{-3}$ | Exp.       | [2e+01, 3e+02]           | 294         | 294          |             |
| **Optical & dielectric**           |        |               |            |                          |             |
| Refractive index (DFT)             | $n_c$  | -             | DFT        | [1e+00, 3e+00]           | 382         | 382          |             |
| Refractive index (exp.)            | $n_e$  | -             | Exp.       | [1e+00, 2e+00]           | 516         | 516          |             |
| Dielec. constant (DFT)             | $k_c$  | -             | DFT        | [3e+00, 9e+00]           | 382         | 382          |             |
| Dielec. constant at freq. $f$      | $k_f$  | -             | Exp.       | [2e+00, 1e+01]           | 1187        | 1187         |             |
| **Mechanical**                     |        |               |            |                          |             |
| Young’s modulus                    | $E$    | MPa           | Exp.       | [2e-02, 4e+03]           | 592         | 322          | 914          |
| Tensile strength at yield          | $\sigma_y$ | MPa         | Exp.       | [3e-05, 1e+02]           | 216         | 78           | 294          |
| Tensile strength at break          | $\sigma_b$ | MPa         | Exp.       | [5e-03, 2e+02]           | 663         | 318          | 981          |
| Elongation at break                | $\epsilon_b$ | -         | Exp.       | [3e-01, 1e+03]           | 868         | 260          | 1128         |
| **Permeability**                   |        |               |            |                          |             |
| $O_2$ gas permeability             | $\mu_{O_2}$ | barrier     | Exp.       | [5e-06, 1e+03]           | 390         | 210          | 600          |
| $CO_2$ gas permeability            | $\mu_{CO_2}$ | barrier    | Exp.       | [1e-06, 5e+03]           | 286         | 119          | 405          |
| $N_2$ gas permeability             | $\mu_{N_2}$ | barrier    | Exp.       | [3e-05, 5e+02]           | 384         | 99           | 483          |
| $H_2$ gas permeability             | $\mu_{H_2}$ | barrier    | Exp.       | [2e-02, 5e+03]           | 240         | 46           | 286          |
| $He$ gas permeability              | $\mu_{He}$ | barrier    | Exp.       | [5e-02, 2e+03]           | 239         | 58           | 297          |
| $CH_4$ gas permeability            | $\mu_{CH_4}$ | barrier | Exp.       | [4e-04, 2e+03]           | 331         | 47           | 378          |
|                                    |        |               |            |                          | 28,061      | 7456         | 35,517       |

$^a$Experiments (Exp.); density functional theory (DFT).

$^b f \in [1, 2, 3, 4, 5, 6, 7, 9, 15]$ is the log$_{10}$(frequency in Hz), e.g., $k_3$ is the dielectric constant at a frequency of 1 kHz.
attention and feed-forward network layer (see Fig. 1b). Figure 3a–c displays the normalized attention maps summed over all 12 attention heads and 12 encoders of polyBERT for the same PSMILES strings as in Fig. 2. Large dots indicate high attention scores, while small dots show weak attention scores. The attention scores can be interpreted as the importance of knowing the position and type of another token (or chemical motif) and its impact on the current token’s latent space. The [CLS], _ and [SEP] tokens are auxiliary tokens. The first two tokens indicate the beginning of PSMILES strings and the last token shows the end of PSMILES strings. We notice high attention scores for the [CLS], _, and first [*] tokens in all panels a to c that imply the connection of the auxiliary tokens to the beginning of PSMILES strings. Also, we observe at least intermediate attention scores to next and next-to-next neighbors (first and second off-
network layers of polyBERT (see Fig.1b) for the same polymers as in components) of the neuron activations in the feed-forward neural network layers disable and enable certain attention layers incorporate positional and relational knowledge and the feed-forward neural network layers account for more than 60% of the parameters. Each of the four components represent a set of distinct neurons that is active for specific tokens (x-axes). For example, the fourth set of neurons is active if polyBERT predicts latent spaces for the auxiliary tokens. The third set of neurons fire in the case of the first two c tokens and the first set of neurons are active for side chain c or c atoms, except in the case of the cn token, which has its own set of neurons (second set of neurons). In total, the attention layers incorporate positional and relational knowledge and the feed-forward neural network layers disable and enable certain routes through polyBERT. Both factors modulate the polyBERT fingerprints.

Not surprisingly, the computations of polyBERT and PG fingerprints scale nearly linearly with the number of PSIMILES strings although their performance (i.e., pre-factor) can be quite different, as shown in the log-log scaled Fig. 4. The computation of polyBERT (GPU) is over two orders of magnitude (215 times) faster than computing PG fingerprints. polyBERT fingerprints may be computed on CPUs and GPUs. Because of the presently large efforts in industry to develop faster and better GPUs, we expect the computation of polyBERT fingerprint to become even faster in the future. Time is very important for high-throughput polymer informatics pipelines that identify polymers from large candidate sets. With an estimate of 0.30 ms/PSIMILES for the multitask deep neural networks (see Property Prediction section), the total time using the polyBERT-based pipeline to predict 29 polymer properties sums to 1.06 ms/polymer/GPU.

Property prediction

For benchmarking the property prediction accuracy of polyBERT and PG fingerprints, we train multitask deep neural networks for each property category defined in Table 1. In our previous study, we observed that these property categories resulted in the development of models exhibiting superior performance. Multitask deep neural networks have demonstrated best-in-class results for polymer property predictions while being fast, scalable, and readily amenable if more data points become available. Unlike single-task models, multitask models simultaneously predict numerous properties (tasks) and harness inherent but hidden correlations in data to improve their performance. Such correlation exists, for instance, between $T_g$ and $T_m$, but the exact correlation varies across specific polymer chemistries. Multitask models learn and improve from these varying correlations in data. The training protocol of the multitask deep neural networks follows state-of-the-art methods involving five-fold cross-validation and a consolidating meta learner that forecasts the final property values based upon the ensemble of cross-validation predictors. More details about multitask deep neural networks are provided in the Methods section. Their training process is outlined in Supplementary Figure S2.

Figure 5a shows the coefficient of determination ($R^2$) averages and standard deviations across the five validation data sets of the cross-validation process for 29 polymer properties. The averages are independent of the data set splits, while the standard deviations show the variance of the prediction performance for the different splits. Smaller standard deviations indicate data sets with homogeneously distributed data points in the learning space. Large standard deviations stem from inhomogeneously distributed data points of usually smaller data sets. Cross-validation is shown to establish an independence of the data set splits for polymer predictions. Root-Mean-Square Error (RMSE) and $R^2$ values for the cross-validation and meta learner models can be found in Supplementary Table S1–S3 for all polymers, homopolymers, and copolymers, respectively. We find the prediction accuracy to be better for thermal and mechanical properties of copolymers (relative to that for homopolymers) and slightly worse for the gas permeabilities, similar to previous findings. Overall, PG performs best ($R^2 = 0.81$) but is very closely followed by polyBERT ($R^2 = 0.80$). This overall performance order of the fingerprint types is persistent with the category averages and properties, except for $X_c$, $X_n$, and $c_r$, where polyBERT slightly outperforms PG fingerprints. We note that polyBERT and PG fingerprints are both practical routes for polymer featurization because their $R^2$ values lie close together and are generally high. polyBERT fingerprints have the accuracy of the handcrafed PG fingerprints but are over two orders of magnitude faster (see Fig. 4).

Figure 5b shows high $R^2$ values for each meta learner (one for each category), suggesting an exceptional prediction performance across all properties. We train the meta learners on unseen 20% of the data set and validate using 80% of the data set (also used for cross-validation). The reported validation $R^2$ values thus only partly measure the generalization performance with respect to the full data set. Meta learners can be conceived as taking decisive roles in selecting the best values from the predictions of the five cross-validation models. We use the meta learners for all property predictions in this work. Supplementary Figs. S9–S14 show the meta learners’ parity plots.

The ultrafast and accurate polyBERT-based polymer informatics pipeline allows us to predict all 29 properties of the 100 million hypothetical polymers that were originally created to train polyBERT. Figure 5c shows the minimum, mean, and maximum for each property. Histograms are given in Supplementary Figs. S15–S20. Given the vast size of our data set and consequential chemical space of the 100 million hypothetical polymers, the minimum and maximum values can be interpreted as potential boundaries of the total polymer property space. In addition, a data set of this magnitude presents numerous opportunities for obtaining fascinating insights and practical applications. For example, it can be utilized in future studies to establish standardized benchmarks for testing and evaluating ML models in the domain of polymer informatics. The data set may also reveal structure-property information that provides guidance for design rules, helps to
identify unexplored areas to search for new polymers, or facilitates direct selection of polymers with specific properties through nearest neighbor searches, as evidenced in a recent study. The data set may also contain subspaces of distinct polymer classes, such as biodegradable or low-carbon polymer classes. However, these aspects are beyond the scope of this study. The data set with 100 million hypothetical polymers including the predictions of 29 properties is available for academic use. The total CO2 emissions for predicting 29 properties of 100 million hypothetical polymers are estimated to be 5.5 kgCO2eq (see CO2 Emission and Timing section).

Other advantages of polyBERT: beyond speed and accuracy

The feed-forward network (last layer in Fig. 1b), which predicts masked tokens during the self-supervised training of polyBERT, enables the mapping of numerical latent spaces (i.e., fingerprints) to PSMILES strings. However, because we average over the token dimension of the last latent space to compute polyBERT fingerprints, we cannot unambiguously map the current fingerprints back to PSMILES strings.

A modified future version of polyBERT that provides PSMILES strings encoding and fingerprint decoding could involve inserting a dimensionality-reducing layer after the last Transformer encoder. Fingerprint decoders are important elements of design informatics pipelines that invert the prediction pipeline to meet property specifications. We note that the current choice of computing polyBERT fingerprints as pooling averages stems from basic dimensionality reduction considerations that require no modification of the DeBERTa architecture.

A second advantage of the polyBERT approach is interpretability. Analyzing the chemical relevancy of polyBERT fingerprints (as discussed in the Fingerprints section) in greater detail can reveal chemical functions and interactions of structural parts of the polymers. As illustrated with the examples of the three polymers in Fig. 3, deciphering...
and visualizing the attention layers of the Transformer encoders can reveal such information. Saliency methods\textsuperscript{57} may also be used to directly explain the relationships between structural parts of the PSMILES strings (inputs) and polymer properties (outputs).

Yet another advantage of the polyBERT approach is its coverage of the entire chemical space. Molecule SMILES strings are a subset of polymer SMILES strings and differ by only two stars (\texttt{[*\textendash\textendash]}) symbols that indicate the two endpoints of the polymer repeat unit. polyBERT has no intrinsic limitations or functions that obstruct predicting fingerprints for molecule SMILES strings. Our first experiments show consistent and well-conditioned fingerprints for molecule SMILES strings using polyBERT that required only minimal changes in the canonicalization routine.

**Discussion**

Here, we show a generalizable, ultrafast, and accurate polymer informatics pipeline that is seamlessly scalable on cloud hardware and suitable for high-throughput screening of huge polymer spaces. polyBERT, which is a Transformer-based NLP model modified for the polymer chemical language, is the critical element of our pipeline. After training on 100 million hypothetical polymers, the polyBERT-based informatics pipeline arrives at a representation of polymers and predicts polymer properties over two orders of magnitude faster but at the same accuracy as the best pipeline based on handcrafted PG fingerprints.

The total polymer universe is gigantic, but currently limited by experimentation, manufacturing techniques, resources, and economical aspects. Contemplating different polymer types such as homopolymers, copolymers, and polymer blends, novel undiscovered polymer chemistries, additives, and processing conditions, the number of possible polymers in the polymer universe is truly limitless. Searching this extraordinarily large space enabled by property predictions is limited by the prediction speed. The accurate prediction of 29 properties for 100 million hypothetical polymers in a reasonable time demonstrates that polyBERT is an enabler to extensive explorations of this gigantic polymer universe at scale. polyBERT paves the pathway for the discovery of novel polymers 100 times faster (and potentially even faster with newer GPU generations) than state-of-the-art informatics approaches - but at the same accuracy as slower handcrafted fingerprinting methods – by leveraging Transformer-based ML models originally developed for NLP. polyBERT fingerprints are dense and chemically pertinent numerical representations of polymers that adequately measure polymer similarity. They can be used for any polymer informatics task that requires numerical representations of polymers such as property predictions (demonstrated here), polymer structure predictions, ML-based synthesis assistants, etc. polyBERT fingerprints have a huge potential to accelerate past polymer informatics pipelines by replacing the handcrafted fingerprints with polyBERT fingerprints. polyBERT may also be used to directly design polymers based on fingerprints (that can be related to properties) using polyBERT’s decoder that has been trained during the self-supervised learning. This, however, requires retraining and structural updates to polyBERT and is thus part of a future work.

**Methods**

**PSMILES canonicalization**

The string representations of homopolymer repeat units in this work are PSMILES strings. PSMILES strings follow the SMILES\textsuperscript{24} syntax definition but use two stars to indicate the two endpoints of the polymer repeat unit (e.g., \texttt{[*\textendash\textendash]}) for polyethylene. The raw PSMILES syntax is non-unique; i.e., the same polymer may be represented using many PSMILES strings; canonicalization is a scheme to reduce the different PSMILES strings of the same polymer to a single unique canonicalized PSMILES string. polyBERT requires canonicalized PSMILES strings because polyBERT fingerprints change with different writings of PSMILES strings. In contrast, PG fingerprints are invariant to the way of writing PSMILES strings and, thus, do not require canonicalization. Figure 6 shows three varieties of PSMILES strings that leave the polymer unchanged. The translational variety of PSMILES strings allows to move the repeat unit window of polymers (cf., white and red box). The multiplicative variety permits to write polymers as multiples of the repeat unit (e.g., twofold repeat unit of Nylon 6), while the permutational variety stems from the SMILES syntax definition\textsuperscript{7} and allows syntactical permutations of PSMILES strings that leave the polymer unchanged.

For this work, we developed the canonicalize_psmiles Python package that finds the canonical form of PSMILES strings in four steps; (i) it finds the shortest PSMILES string by searching and removing repetition patterns, (ii) it connects the polymer endpoints to create a periodic PSMILES string, (iii) it canonicalizes the periodic PSMILES string using RDKit’s\textsuperscript{34} canonicalization routines, (iv) it breaks the periodic PSMILES string to create the canonical PSMILES string. The canonicalize_psmiles package is available at https://github.com/Ramprasad-Group/canonicalize_psmiles.

**Polymer fingerprinting**

Fingerprinting converts geometric and chemical information of polymers (based upon the PSMILES string) to machine-readable numerical representations in the form of vectors. These vectors are the polymer fingerprints and can be used for property predictions, similarity searches, or other tasks that require numerical representations of polymers.

We compare the polyBERT fingerprints, developed in this work, with the handcrafted Polymer Genome (PG) polymer fingerprints. PG fingerprints capture key features of polymers at three hierarchical length scales\textsuperscript{38}. At the atomic scale (1st level), PG fingerprints track the occurrence of a fixed set of atomic fragments (or motifs)\textsuperscript{39}. The block scale (2nd level) uses the Quantitative Structure-Property Relationship (QSPR) fingerprints\textsuperscript{36} for capturing features on larger length-scales as implemented in the cheminformatics toolkit RDKit\textsuperscript{37}. The chain scale (3rd level) fingerprint components deal with “morphological descriptors” such as the ring distance or length of the largest side-chain\textsuperscript{38}. The PG fingerprints are developed within the Ramprasad research group and used, for example, at https://PolymerGenome.org. More details can be found in References\textsuperscript{33,34}.
As discussed recently\textsuperscript{a,b}, we sum the composition-weighted polymer fingerprints to compute copolymer fingerprints \( f = \sum_{i}^{N} f_i c_i \), where \( N \) is the number of comonomers in the copolymer, \( f_i \) the \( i \)th comonomer fingerprint vector, and \( c_i \) the fraction of the \( i \)th comonomer. This approach renders copolymer fingerprints invariant to the order in which one may sort the comonomers and satisfies the two main demands of uniqueness and invariance to different (but equivalent) periodic unit specifications. While the current fingerprinting scheme is most appropriate for random copolymers, other copolymer flavors may be encoded by adding additional fingerprint components. Contrary to homopolymer fingerprints, copolymer fingerprints may not be interpretable (e.g., the composition-weighted sum of the fingerprint component “length of largest side-chain” of two homopolymers has no physical meaning).

Multitask neural networks

Multitask deep neural networks simultaneously learn multiple polymer properties to utilize inherent correlations of properties in data sets. The training protocol of the concatenation-conditioned multitask predictors follows state-of-the-art techniques involving five-fold cross-validation and a meta learner that forecasts the final property values based upon the ensemble of cross-validation predictors\textsuperscript{15,33}. Supplementary Figure S2 details this process. After shuffling, we split the data set into two parts and use 80\% for the five-fold cross-validation models and for validating the meta learners. 20\% of the data set is used for training the meta learners. We use the Hyperband method\textsuperscript{58} of the Python package KerasTuner\textsuperscript{59} to fully optimize all hyperparameters of the neural networks, including the number of layers, number of nodes, dropout rates, and activation functions. The Hyperband method finds the best set of hyperparameters by minimizing the Mean Squared Error (MSE) loss function. We perform data set stratification of all splits based on the polymer properties. The multitask deep neural networks are implemented using the Python API of TensorFlow\textsuperscript{20,21}.

\textbf{CO\textsubscript{2} emission and timing}

Experiments were conducted using a private infrastructure, which has an estimated carbon efficiency of 0.432 kgCO\textsubscript{2}eq kWh\textsuperscript{-1}. A total of 31 h of computations were performed on four Quadro-GP100-16GB (thermal design power of 235 W) for training polyBERT. Total emissions are estimated to be 12.6 kgCO\textsubscript{2}eq. About 8 h of computations on four GPUs were necessary for training the cross-validation and meta learner models with an estimated emission of 3.3 kgCO\textsubscript{2}eq for polyBERT and Polymer Genome fingerprints, respectively. The total emissions for predicting 29 properties for 100 million hypothetical polymers are estimated to be 5.3 kgCO\textsubscript{2}eq, taking a total of 13.5 h. Estimations were conducted using the Machine Learning Impact calculator presented in ref. 61.

\textbf{Reporting summary}

Further information on research design is available in the Nature Portfolio Reporting Summary linked to this article.

\textbf{Data availability}

The data set of 100 million hypothetical polymers with 29 predicted properties is available for academic use at https://doi.org/10.5281/zenodo.7766806.

\textbf{Code availability}

The polyBERT code is available for academic use at https://github.com/Ramprasad-Group/polyBERT\textsuperscript{4} and Zenodo\textsuperscript{5}. The trained polyBERT model is available at https://huggingface.co/kuelumbus/polyBERT. The Python package for canonicalizing PSMILES strings is available at https://github.com/Ramprasad-Group/canonicalize_psmiles. polyBERT-based property predictions will be made accessible through the polymer informatics platform Polymer Genome at https://PolymerGenome.org.

\textbf{References}

1. Plastics Europe. https://plasticseurope.org/knowledge-hub/plastics-the-facts-2021/.
2. Batra, R., Song, L. & Ramprasad, R. Emerging materials intelligence ecosystems propelled by machine learning. Nat. Rev. Mater. \textbf{6}, 655–678 (2021).
3. Chen, L. et al. Polymer informatics: Current status and critical next steps. Mater. Sci. Eng., R. Rep. \textbf{144}, 100595 (2021).
4. Audus, D. J. & de Pablo, J. J. Polymer informatics: Opportunities and challenges. ACS Macro Lett. \textbf{6}, 1078–1082 (2017).
5. Adams, N. & Murray-Rust, P. Engineering polymer informatics: Towards the computer-aided design of polymers. Macromol. Rapid Commun. \textbf{29}, 615–632 (2008).
6. Kuenneth, C., Schertz, W. & Ramprasad, R. Copolymer informatics with multitask deep neural networks. Macromolecules \textbf{54}, 5957–5961 (2021).
7. Kuenneth, C. et al. Polymer informatics with multi-task learning. Patterns \textbf{2}, 100238 (2021).
8. Doan Tran, H. et al. Machine-learning predictions of polymer properties with polymer genome. J. Appl. Phys. \textbf{128}, 171104 (2020).
9. Chen, G., Tao, L. & Li, Y. Predicting polymers’ glass transition temperature by a chemical language processing model. Polymers \textbf{13}, 1898 (2021).
10. Pilania, G., Iverson, C. N., Lookman, T. & Marrone, B. L. Machine-learning-based predictive modeling of glass transition temperatures: A case of polyhydroxyalkanoate homopolymers and copolymers. J. Chem. Inf. Model. \textbf{59}, 5013–5025 (2019).
11. Kuenneth, C. et al. Bioplastic design using multitask deep neural networks. Commun. Mater. \textbf{3}, 96 (2022).
12. Barnett, J. W. et al. Designing exceptional gas-separation polymer membranes using machine learning. Sci. Adv. \textbf{6} (2020) https://doi.org/10.1126/sciadv.aaaz4301.
13. Kim, C., Batra, R., Chen, L., Tran, H. & Ramprasad, R. Polymer design using genetic algorithm and machine learning. Comput. Mater. Sci. \textbf{166}, 110607 (2021).
14. Kern, J., Chen, L., Kim, C. & Ramprasad, R. Design of polymers for energy storage capacitors using machine learning and evolutionary algorithms. J. Mater. Sci. \textbf{56}, 19623–19635 (2021).
15. Gurnani, R. et al. polyG2G: A novel machine learning algorithm applied to the generative design of polymer dielectrics. Chem. Mater. \textbf{33}, 7008–7016 (2021).
16. Batra, R. et al. Polymers for extreme conditions designed using syntax-directed variational autoencoders. Chem. Mater. \textbf{32}, 10489–10500 (2020).
17. Wu, S. et al. Machine-learning-assisted discovery of polymers with high thermal conductivity using a molecular design algorithm. npj Comput. Mater. \textbf{5}, 66 (2019).
18. Le, T., Epa, V. C., Burden, F. R. & Winkler, D. A. Quantitative structure-property relationship modeling of diverse materials properties. Chem. Rev. \textbf{112}, 2889–2919 (2012).
19. Rogers, D. & Hahn, M. Extended-connectivity fingerprints. J. Chem. Inf. Model. \textbf{50}, 742–754 (2010).
20. Mannodi-Kanakkithodi, A., Pilania, G., Huan, T. D., Lookman, T. & Ramprasad, R. Machine learning strategy for accelerated design of polymer dielectrics. Sci. Rep. \textbf{6}, 20952 (2016).
21. Huan, T. D., Mannodi-Kanakkithodi, A. & Ramprasad, R. Accelerated materials property predictions and design using motif-based fingerprints. Phys. Rev. B \textbf{92}, 014106 (2015).
22. Moriwaki, H., Tian, Y.-S., Kawashita, N. & Takagi, T. Mordred: a molecular descriptor calculator. J. Cheminf. \textbf{10}, 4 (2018).
23. Vaswani, A. et al. Attention is all you need. arXiv (2017), https://doi.org/10.48550/arXiv.1706.03762.
24. Weininger, D. SMILES, a chemical language and information system. 1. Introduction to methodology and encoding rules. J. Chem. Inf. Model. 28, 31–36 (1988).
25. Chithrananda, S., Grand, G., Ramsundar, B. ChemBERTa: Large-scale self-supervised pretraining for molecular property prediction. arXiv (2020), https://doi.org/10.48550/arXiv.2010.09885.
26. Wang, S., Gue, Y., Wang, Y., Sun, H., Li, P. Wang. SMILES-BERT. Proceedings of the 10th ACM International Conference on Bioinformatics, Computational Biology and Health Informatics, New York, NY, USA; pp 429–436 (2019), https://doi.org/10.1145/3307339.3342186.
27. Li, J. & Jiang, X. Mol-BERT: An effective molecular representation with BERT for molecular property prediction. Wirel. Commun. Mobile Comput. 2021, 1–7 (2021).
28. Devlin, J., Chang, M.-W., Lee, K., Toutanova, K. BERT: Pre-training of deep bidirectional transformers for language understanding. arXiv (2018), https://doi.org/10.48550/arXiv.1810.04805.
29. Schwaller, P. et al. Molecular transformer: A model for uncertainty-calibrated chemical reaction prediction. ACS Central Sci. 5, 1572–1583 (2019).
30. Schwaller, P. et al. Predicting retrosynthetic pathways using transformer-based models and a hyper-graph exploration strategy. Chem. Sci. 11, 3316–3325 (2020).
31. Xu, C., Wang, Y., Farimani, A. B. TransPolymer: A transformer-based language model for polymer property predictions. arXiv (2022), https://doi.org/10.48550/arXiv.2209.01307.
32. Liu, Y. et al. RobBERTa: A robustly optimized BERT pretraining approach. arXiv (2019), https://doi.org/10.48550/arXiv.1907.11692.
33. Gilmer, J., Schoenholz, S. S., Riley, P. F., Vinyals, O., Dahl, G. E. Neural message passing for quantum chemistry. arXiv (2017), https://doi.org/10.48550/arXiv.1704.01212.
34. Park, J. et al. Prediction and interpretation of polymer properties using the graph convolutional network. ACS Polymers Au 2, 213–222 (2022).
35. Gurnani, R., Kuenneth, C., Toland, A. & Ramprasad, R. Polymer informatics at scale with multitask graph neural networks. Chem. Mater. 35, 1560–1567 (2023).
36. Aldeghi, M. & Coley, C. W. A graph representation of molecular ensembles for polymer property prediction. Chem. Sci. 13, 10486–10498 (2022).
37. Fang, X. et al. Geometry-enhanced molecular representation learning for property prediction. Nat. Mach. Intell. 4, 127–134 (2022).
38. Wang, Y., Wang, J., Cao, Z. & Barati Farimani, A. Molecular contrastive learning of representations via graph neural networks. Nat. Mach. Intell. 4, 279–287 (2022).
39. He, P., Liu, X., Gao, J., Chen, W. DeBERTa: Decoding-enhanced BERT with disentangled attention. arXiv (2020), https://doi.org/10.48550/arXiv.2006.03654.
40. Degen, J., Wegscheid-Gerlach, C., Zaliani, A. & Rarey, M. On the art of compiling and using ‘Drug-Like’ chemical fragment spaces. ChemMedChem 3, 1503–1507 (2008).
41. Landrum, O. others, RDKit: Open-source cheminformatics. (2006).
42. Jha, A., Chandrasekaran, A., Kim, C. & Ramprasad, R. Impact of dataset uncertainties on machine learning model predictions: the example of polymer glass transition temperatures. Model. Simul. Mater. Sci. Eng. 27, 024002 (2019).
43. Kim, C., Chandrasekaran, A., Jha, A. & Ramprasad, R. Active-learning and materials design: the example of high glass transition temperature polymers. MRS Commun. 9, 860–866 (2019).
44. Kim, C., Chandrasekaran, A., Huan, T. D., Das, D. & Ramprasad, R. Polymer genome: A data-powered polymer informatics platform for property predictions. J. Phys. Chem. C 122, 17575–17585 (2018).
45. Patra, A. et al. A multi-fidelity information-fusion approach to machine learn and predict polymer bandgap. Comput. Mater. Sci. 172, 109286 (2020).
46. Chen, L. et al. Frequency-dependent dielectric constant prediction of polymers using machine learning. npj Comput. Mater. 6, 61 (2020).
47. Venkatram, S., Kim, C., Chandrasekaran, A. & Ramprasad, R. Critical assessment of the hildebrand and hansol solubility parameters for polymers. J. Chem. Inf. Model. 59, 4188–4194 (2019).
48. Zhu, G. et al. Polymer genome-based prediction of gas permeabilities in polymers. J. Polym. Eng. 40, 451–457 (2020).
49. PolyInfo. https://polymer.nims.go.jp/en/.
50. Ahmad, W., Simon, E., Chithrananda, S., Grand, G., Ramsundar, B. ChemBERTa-2: Towards Chemical Foundation Models. arXiv (2022), https://doi.org/10.48550/arXiv.2209.01712.
51. Wolf, T. et al. Transformers: State-of-the-Art Natural Language Processing. Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations. Stroudsburg, PA, USA; pp 38–45, (2020), https://doi.org/10.18653/v1/2020.emnlp-demos.6.
52. Sanh, V., Debut, L., Chaumond, J., Wolf, T. DistillBERT, a distilled version of BERT: smaller, faster, cheaper and lighter. arXiv (2019), https://doi.org/10.48550/arXiv.1910.10108.
53. Kudo, T., Richardson, J. SentencePiece: A simple and language independent subword tokenizer and detokenizer for Neural Text Processing. arXiv (2018), https://doi.org/10.48550/arXiv.1808.06226.
54. Reimers, N., Gurevych, I. Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks. arXiv (2019), https://doi.org/10.48550/arXiv.1908.10084.
55. McInnes, L., Healy, J., Melville, J. UMAP: Uniform Manifold Approximation and Projection for Dimension Reduction. arXiv (2018), https://doi.org/10.48550/arXiv.1802.03426.
56. Alammar, J. Ecco: An Open Source Library for the Explainability of Transformer Language Models. Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing: System Demonstrations. Stroudsburg, PA, USA; pp 249–257, (2021), https://doi.org/10.18653/v1/2021.acl-demo.30.
57. Bastings, J., Filippova, K. The elephant in the interpretability room: Why use attention as explanation when we have saliency methods? arXiv (2020), https://doi.org/10.48550/arXiv.2010.05607.
58. Li, L., Jamieson, K., DeSalvo, G., Rostamizadeh, A. & Talwalkar, A. Hyperband: A novel bandit-based approach to hyperparameter optimization. J. Mach. Learn. Res. 18, 1–52 (2016).
59. O’Malley, T., Bursztein, E., Long, J., Chollet, F., Jin, H., Invernizzi, L. & Keras Tuner. (2019), https://github.com/keras-team/keras-tuner.
60. Martin, A. et al. TensorFlow: Large-scale machine learning on heterogeneous systems. (2015); https://www.tensorflow.org.
61. Lacoste, A., Luccioni, A., Schmidt, V., Dandres, T. Quantifying the carbon emissions of machine learning. arXiv (2019), https://doi.org/10.48550/arXiv.1910.09700.
62. Kuenneth, C., Ramprasad, R. polyBERT: A chemical language model to enable fully machine-driven ultrafast polymer informatics. Zenodo (2023), https://doi.org/10.5281/zenodo.7969082.

Acknowledgements
C.K. thanks the Alexander von Humboldt Foundation for financial support. We acknowledge funding from the Office of Naval Research through a Multidisciplinary University Research Initiative grant (N00014-17-1-2656) and the National Science Foundation (#1941029).

Author contributions
C. K. designed, trained and evaluated the machine learning models and drafted this paper. The work was conceived and guided by R. R. All authors discussed results and commented on the manuscript.
Competing interests
R.R. is the founder of the company Matmerize, Inc., that intends to provide polymer informatics services. A provisional patent has been filed by the Georgia Tech Research Corporation, Atlanta, GA; Inventors: Rampi Ramprasad, and Christopher Kuenneth; Application Number: 63/374,761; Status: patent pending; Aspect covered in the patent application: transformer based informatics pipeline for polymer representation and property predictions.

Additional information
Supplementary information The online version contains supplementary material available at https://doi.org/10.1038/s41467-023-39868-6.

Correspondence and requests for materials should be addressed to Rampi Ramprasad.

Peer review information: Nature Communications thanks the anonymous, reviewer(s) for their contribution to the peer review of this work. A peer review file is available.

Reprints and permissions information is available at http://www.nature.com/reprints

Publisher’s note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license, and indicate if changes were made. The images or other third party material in this article are included in the article’s Creative Commons license, unless indicated otherwise in a credit line to the material. If material is not included in the article’s Creative Commons license and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this license, visit http://creativecommons.org/licenses/by/4.0/.

© The Author(s) 2023