Human-Readable SMILES: Translating Cheminformatics to Chemistry

Diego Garay-Ruiz[†*] and Carles Bo[†,‡]

[†]Institute of Chemical Research of Catalonia (ICIQ), The Barcelona Institute of Science and Technology (BIST), Av. Països Catalans, 16, 43007 Tarragona, Spain

[‡]Departament de Química Física i Inorgànica, Universitat Rovira i Virgili, Marcel·lí Domingo s/n, 43007 Tarragona, Spain

E-mail: dgaray@iciq.es

KEYWORDS cheminformatics, string representations, SMILES

ABSTRACT: Molecular string representations are a key asset in cheminformatics and are becoming increasingly relevant to the general chemical community, due to the steadily growing impact of Big Data and Machine Learning. Among all of the existing string representations that have been proposed, SMILES (Simplified Molecular Input Line Entry Specification) are probably the *de facto* standard as of today. Despite their convenience as a way to store unique molecular structures in databases, however, SMILES are not easy to understand for most chemists: that is, it is difficult for an untrained chemist to grasp the molecule that a SMILES is describing. To mitigate this, we propose the HumanSMILES algorithm: a simple procedure that can translate a SMILES string into a more interpretable name, inspired by common abbreviations and names employed in general organic chemistry. The Human-Readable SMILES can describe linear structures and general non-fused cyclic structures, with a set of naming rules that combines automated processing and chemical knowledge. The code is available open-source, as well as a web application.

INTRODUCTION

The nomenclature of compounds has been an essential part of Chemistry since its earliest days: discovering a novel substance was almost always immediately followed by choosing a name for it. Throughout the years, this naming process started to be less arbitrary and more rational, building on all the previous understanding of compositions and structures. However, even nowadays, when very systematic naming frameworks have been defined[1], a lot of the older naming conventions still prevail among scientists.

Properly 'naming' a chemical structure means to choose a unique combination of letters and numbers, unambiguously related to the structure in question. In this sense, molecular string representations such as SMILES (Simplified Molecular Input Line Entry Specification) [2–4] or InChIs (INternational CHemical Identifier)[5] could be also considered as a way of naming molecules, as stated by Weininberg in the very first introduction of SMILES as a "chemical language".

A major divergence between traditional names and molecular identifiers is their target: while names are made for humans, these strings are tailored for machines. Therefore, concepts as pronounceability, readability or adjustment to language rules are not strictly relevant for this kind of strings. In contrast, it is even more important for them to be complete, systematic and unambiguous, as computers cannot rely on chemical intuition or context to guess unprovided information as we humans might do.

The relevance of molecular identifiers is now bigger than ever. On the one hand, the sheer amount of chemical information that is now available requires of robust databases to adequately store all this data. String-based identifiers are extremely convenient for this kind of storage, as they provide unique identifiers that require minimal disk space, compared to e.g., molecular graphs that require more space. Current online databases (PubChem[6], ChemBL[7]) include SMILES and InChIs as part of the available data, while datasets (GDB-X[8,9], QM9[10,11]…) are often provided as CSV files where strings are indeed the only way to communicate structural information.

On the other hand, the importance of machine learning in chemistry is steadily growing[12–17]. Most ML-related tasks require to process very large amounts of data, making string-based identifiers an excellent choice for the input/output of the machine learning algorithm. SMILES, due to their early installment, are probably the most common string representation for most of these applications. While alternative identifiers have been developed to avoid some of their drawbacks in direct ML applications, such as DeepSMILES[18] or SELFIES[19], it seems likely that SMILES continue to be a standard on the field. Therefore, as this kind of applications keep extending to other fields of Chemistry, more and more chemists shall eventually employ string representations in general, and SMILES in particular.

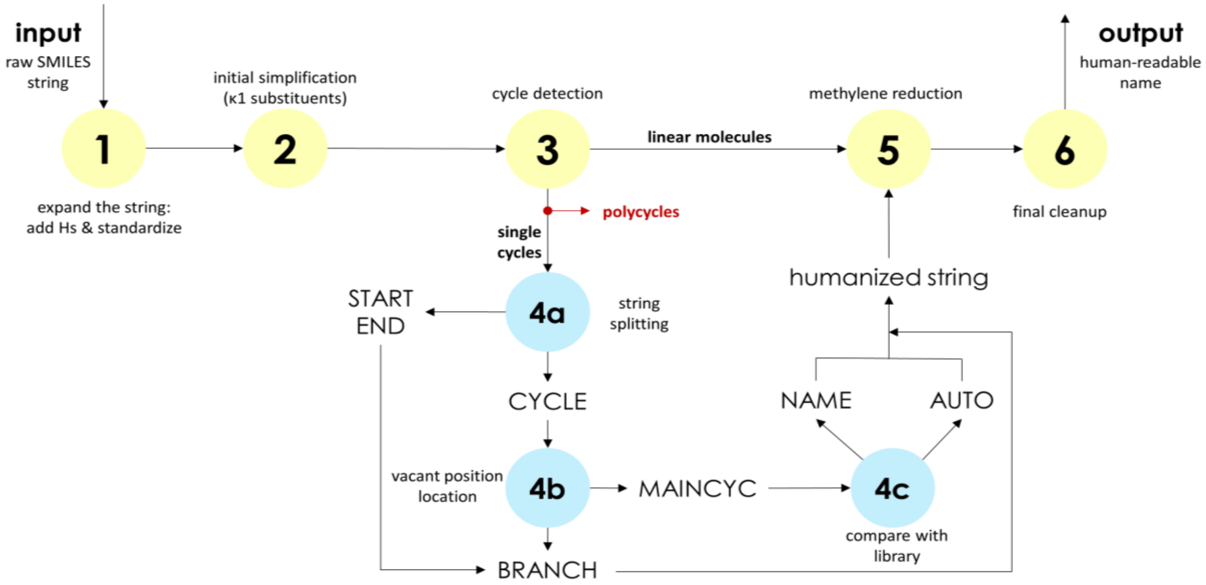


Figure 1. Functioning scheme for HumanSMILES. Yellow circles represent steps undertaken by all input strings, blue circles correspond to the processing of cyclic structures.

However, as we already stated, an undeniable drawback of these identifiers is that they lack human readability. Of course, it can be argued that it is the computer who must understand the representation, not the human behind it. Nevertheless, we think that improvements on string readability would lead to a better understanding of the methods and results of cheminformatics by the general chemical community, and consequently, to a more collaborative and better science. There is also a growing interest and a very active development in making scripting and programming-based tools easier to use and share. Tools such as Jupyter Notebooks[20], combining code, results and explanations in a single environment have become a staple in Data Science and its area-specific applications. In this context, the readability of the code snippets in the notebook becomes a very desirable asset, and even more so regarding how common is to share this kind of resources in online repositories. For the case of Chemistry, the availability of human-readable names for molecules supposes a step forward in this quest for readable code that can be easily reused and adapted by a wide audience.

In this spirit, we propose a translation algorithm for the conversion of SMILES defining small molecules to a molecular-formula-inspired string which can be read and understood by any chemist, even without any background on cheminformatics or string representations. To focus on simplicity and readability, only acyclic, single-cyclic and non-fused polycyclic molecules are supported, as the nomenclature of fused polycycles becomes quite convoluted (just like for their systematic naming[1]). The algorithm is proposed in the form of a Python library[21] based on the RDKit[22], which can be easily integrated with any other Python code to add this naming capacity to pre-existing workflows painlessly. The Human-Readable SMILES are not designed as any kind of *replacement* to molecular string representations, but as a *complement* to improve the general understandability of computational protocols relying on them.

THE ALGORITHM

A Python implementation of our proposed protocol, named HumanSMILES, is available on a GitLab repository[21] (dgarayr/humansmiles). Along this paper, we will explain the design of the algorithm and the steps that it takes from a given input SMILES string to the final human-readable name.

First, we present a simplified graphical depiction of the algorithm (Figure 1). From this chart, we can proceed to explain every individual step, depicted as the yellow and blue circles in Figure 1.

1. SMILES uniformization: read the input(s) as molecule(s) in RDKit and generate expanded SMILES with explicit hydrogens.
2. Simplification of common monodentate motifs: locate groups with convoluted SMILES and reduce them into simpler, single groups.
   - Phenyl group: **[c]1[cH][cH][cH][cH][cH]1 → [Ph]**
   - Trifluoromethyl: **[C]([F])([F])[F] → [CF3]**
   - tert-Butyl: **[C]([CH3])([CH3])[CH3] → [tBu]**

   This step works through direct queries to an inner dictionary, which can be expanded as needed.
3. Single-cycle detection: uses the "**]1**" string in the SMILES as a flag to locate structures which contain cycles.
   - Polycycles are discarded from the algorithm.
   - Single cycles are processed further to identify the cyclic fragment and locate branches (onto **steps 4a-4b-4c**).
   - Non-cyclic structures are directly simplified: the brackets in the SMILES are removed, as valences are already fulfilled and the individual groups can be easily identified. These go directly to **step 5**.

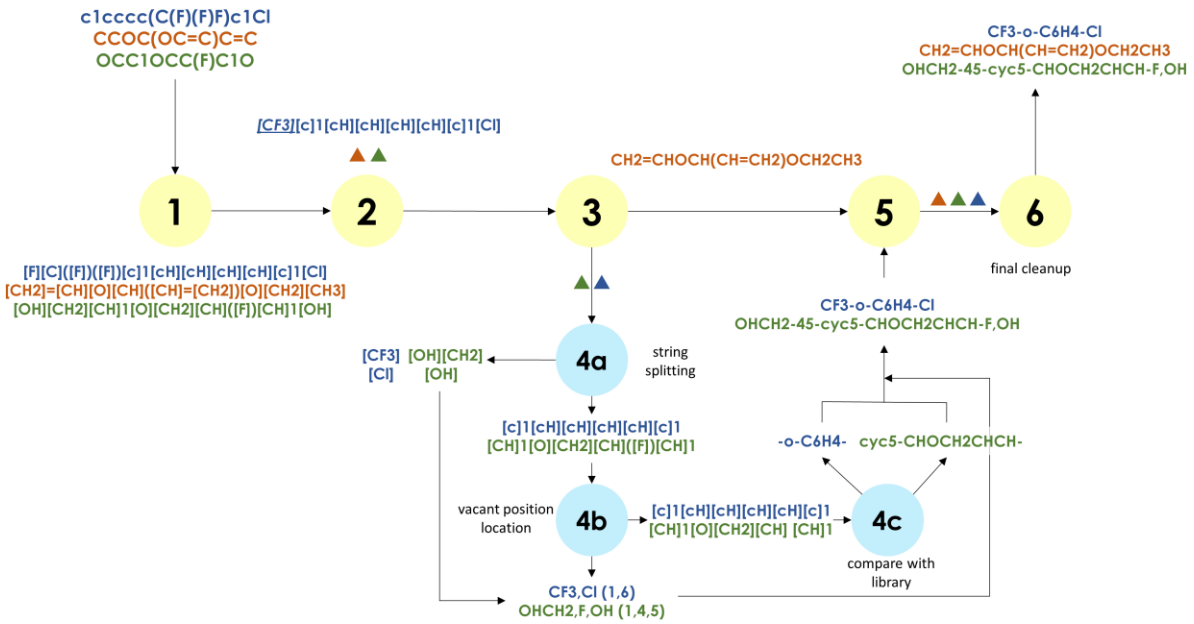


Figure 2. Selected string transformation examples in HumanSMILES. Triangles represent strings that are not changed in given step.

4. Cycle processing.
   a. The " **]1"** delimiters present in the SMILES to define single cycles are used to split the string in three parts: START (before 1st match), END (after last match) and CYCLE (in between matches).
   b. Into the CYCLE fragment, the valences of the most common elements inside cycles (C, N, O and S) are analyzed to locate substitution points. These substituents are also detected by matching parentheses and brackets inside the string. Then, the string is broken to isolate the main cycle (MAINCYC) from the substituents (BRANCH), storing position information. START and END fragments are joined with the BRANCH in a list of substituents.
   c. The MAINCYC is compared with an inner dictionary containing common & identifiable cyclic motifs: all substituted benzenes and pyridines. If it matches any of the stored motifs, a name is taken from the dictionary. When MAINCYC is not in the database, a custom name is generated automatically: the MAINCYC string is simplified (bracket removal, as in **step 3c**), and it is preceded by a **cycN-** prefix stating the number of atoms in the cycle. The final cycle name is joined with the BRANCH and START strings: position information precedes the cycle name to specify how it is substituted as in common molecular formulas.
5. Methylene simplification: the processed string (from either acyclic or single-cycle structures) is recursively searched for consecutive CH2 groups, which are converted to **(CH2)N** strings.

6. Clean-up and wrap up: final strings are searched for final commas, hyphens or empty parentheses resulting from the auto-naming process that can be safely removed.

To better illustrate how our translation procedure works, we selected three representative SMILES, for which we will show every transformation along the flowchart in Figure 1. These three main pathways along the code are shown in Figure 2 (polycycles, which would be discarded after **step 3**, have been omitted for brevity).

- Linear molecules (dark orange) do only require minor changes after uniformization of the string: mainly, removing the brackets that delimit individual groups in the SMILES.
- The trifluoromethyl group in the blue string is simplified to [CF3] in **step 2**.
- Both cycles (in green and in blue) are split in order to separate the fragment that describes the cyclic part alone (MAINCYC) and the branches, which may appear before, after or inside the cyclic-defining string in the SMILES. The blue string, corresponding to a disubstituted benzene, is identified in the library and receives a pre-defined name. The green string, in contrast, is a more complex heterocycle that is not stored anywhere in the library, and thus receives the automatic **cyc5-** nomenclature.

Apart from the main library, we have also set up a web application[23] in a Heroku instance, allowing the user to quickly test HumanSMILES interactively before downloading the code and its dependencies. This application has been designed as a straightforward demo, where the user inputs a single SMILES string at a time and receives the corresponding Human-Readable SMILES and the RDKit-drawn molecular structure.

USE CASES

To showcase the capabilities of our protocol, we will be presenting two main use cases.

1. Generation of a named database of monodentate ligands.
2. Application to a large dataset of small molecules.

Case I: Substituent database.

During the development of a project on the small-scale automatization of QM calculations for catalyst design, we needed to generate a reasonable database with the 3D structures for several common substituent groups. As a starting point, we selected the list produced in a recent work by P. Ertl[24] , including the SMILES defining the 6278 most common monodentate substituents taken from >700000 bioactive molecules in the ChemBL database.

While generating the corresponding 3D structures from the SMILES is a trivial task to accomplish with RDKit, the resulting database was not indexed properly: neither SMILES nor index numbers seemed clear or informative enough as DB keys. In contrast, the ’translated’ Human-Readable SMILES that we propose are much clearer, giving the chemist an immediate idea of which substituent is being requested at a time. In this way, not only the overall automatization code gets more readable, but also the results become easier to organize, as simple and ’comfortable’ strings are available at every part of a given workflow. In the specific case of our automatization script, the Human-Readable SMILES do not only allow to easily index the ligand database, but also to have interpretable names for all intermediate files and folders, simplifying the further processing and curation of this data.

In the original list of SMILES, the point of substitution of every structure is marked with the string “**[R]**”, which is not recognized by RDKit. A first pre-processing step replaces this string by the “**[*]”** notation to properly read all structures as molecules. From there, the rest of the protocol does not require further changes to deal either with radicals or with “full” molecules. The linking point of the substituent will be the first item in the string for all the resulting Human-Readable SMILES.

From the 6278 substituents in the original list, 5228 are accepted and named (**83.3%**): the remaining 16.7% corresponds to polycyclic structures that are not integrated in the naming heuristics.

Table 1. Percentage of substitutions made in step 2 of the algorithm for Ertl's monodentate substituent DB.

Substituent	% of struc.		
–CN	2.47	–CF_3	3.19
–tBu	1.61	–OCH_3	7.75
–Ph	5.37	–$CONH_2$	2.54
–NO_2	1.68	–COOH	6.14
–SO_2	3.98	–CCl_3	0.02

A focal point of our naming strategy is the substitution of common motifs present in the expanded SMILES (**step 2** in Figure 1). Therefore, it is interesting to check how many of these substitutions are actually applied across the DB, as shown in Table 1.

Most of the tested groups, except for the trichloromethyl, have an important presence along the database: we might highlight the methoxy, the carboxylate and the phenyl groups, each supposing >5% of the structures.

The other main point in the string humanization is the detection of cyclic patterns (**steps 3 – 4**). Here, from our 5228 named structures, 3729 of them (71.3%) are processed as cycles (not including here monosubstituted phenyl groups, which are handled in the previous step). Specifically, 1706 are auto-named through the **cycN +** simplified SMILES syntax, 1680 are polysubstituted benzene derivatives and 343 are pyridine derivatives.

Regarding the compaction of methylenes (**step 5**), almost a quarter of the entries of the ligand database (23.7%) have consecutive CH_2 groups and thus can be quite abbreviated in this stage.

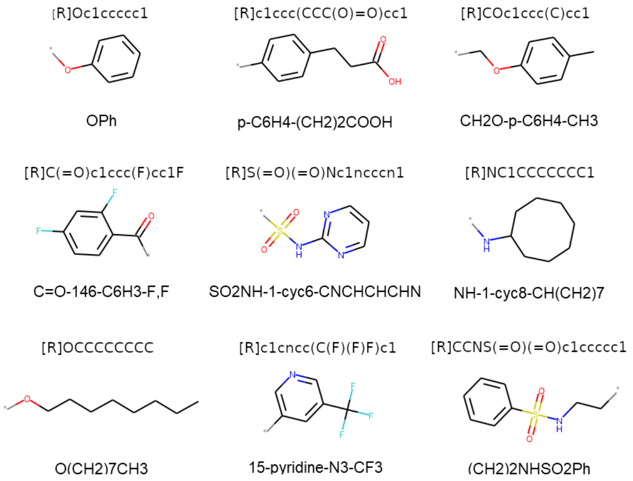


Figure 3. Selection of Human-Readable SMILES for several structures along Ertl's database, highlighting the most relevant transformations. Original SMILES are shown above every structure for reference.

A small selection on the kind of names generated for this database is provided in Figure 3, while a more complete reference can be found in the Supporting Information. These examples were selected to highlight the most relevant features of the protocol, such as i) the C6H(6-**n**) string used to name **n**-substituted benzene derivatives, ii) the high degree of compaction for long methylene chains, both in linear and in cyclic molecules, iii) the ability to seamlessly name non-recognized heterocyclic structures, iv) the simple nomenclature for common groups such as the phenyl (Ph), the trifluoromethyl (CF3) or the sulfoxide (SO2) or v) the proper labelling of the substituted positions in cycles.

Case II: Large molecular dataset.

As we did already mention, molecular datasets are a crucial asset in Chemistry, and are becoming more and more important as Big Data and Machine Learning keep developing. While the available models get more powerful and easy to use and the datasets grow in size, the interpretability of how the model works in the end can be lost. Because of this, another important topic in Machine Learning and its applications is, precisely, to aim for more understandable approaches and models, inspecting the 'innards' of the program instead of considering it a black box.

Our approach to string simplification might enter in this quest for interpretability: for example, the intermediate structures generated by a given model could be converted to SMILES and then translated to comprehensible strings. This additional plain text output is much easier to analyze at a glance than traditional molecular string representations, while also requiring less storage and overhead than, for example, generating image-based visualizations for molecular graphs. To assess the adequacy of our current protocol for this goal, we will be testing the GDB-10[8,9] database to determine how many of the present molecules are eventually converted (recalling that polycycles are not supported and will be immediately discarded) and which kind of names are generated in such a large set.

Here, we will discuss some important aspects about the application of the HumanSMILES protocol to the GDB-10 database, comprising more than three million and a half molecules. The structures in the dataset contain from 1 to 10 non-H atoms, which may be C, O, F or N.

First of all, we have that humanization can be applied to **70.0%** of the SMILES in the database: the remaining 30.0% corresponds to discarded polycyclic structures. The percentage of acceptance is higher for the smaller GDB-n subsets, that include simpler molecules overall and therefore do not comprise as many polycycles. For example, the naming ratio is 79.9% for GDB-9, with 500k structures, and 88.5% for GDB-8, with 70k. Nevertheless, even for the larger collection GDB-10, more than two thirds of the structures, or approximately 2.5 million molecules, are satisfactorily named.

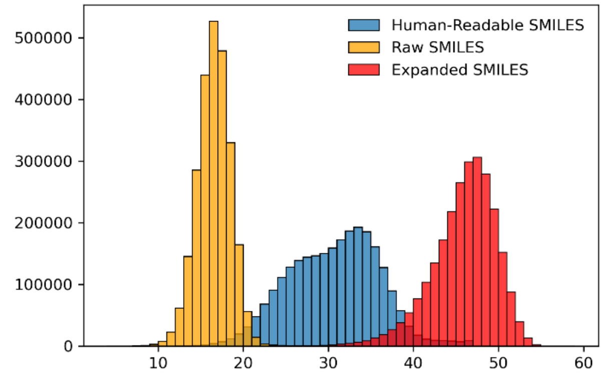


Figure 4. Histograms for character counts in "raw" SMILES from the database, Human-Readable SMILES and expanded explicit-hydrogen-containing SMILES.

Due to the volume of data here, it will not make much sense to assess the quality of the generated names through individual inspection, while some examples are still provided in the Supporting Information. In contrast, we will be presenting some statistics comparing the number of characters of the input SMILES, the Human-Readable SMILES and the non-simplified, expanded SMILES with explicit hydrogens that are used in the first step of the protocol.

The histograms in Figure 4 show how the raw SMILES, right as present in the database, are remarkably shorter than the Human-Readable SMILES, with these being also much shorter than the *expanded* SMILES they are based on. While the compactness of SMILES is desirable in terms of storage, it contributes to their relative obscurity: SMILES are so compact, no matter the described structure, that it becomes difficult to grasp the molecule they refer to. On the other side, while the expanded SMILES are very explicit, they are consistently quite long, even for simpler structures: the left tail of the red histogram is mostly overlapping with the right tail of the blue one. The distribution for the Human-Readable SMILES, in contrast, appears in between the two SMILES variants: more explicit and easier to interpret than the very short raw SMILES but not as long and clunky as the expanded ones. Moreover, it is also *wider* than the other two distributions: instead of being either very short or very long, the character span of the Human-Readable SMILES depends more on the complexity and size of the represented molecule.

Libraries based on the exploration of the chemical space up to a given number of atoms, such as GDB-10, are expected to have quite a large number of cyclic and heterocyclic structures. Therefore, most of them will not follow any of the stored templates, but will be auto-named instead (step **4c**). In fact, in our current subset, we have that **69.1%** of the named structures are indeed auto-named cycles, compared to a very minor **0.09%** of polysubstituted benzenes and **0.30%** of substituted pyridines. This reflects how the auto-naming of cycles is essential for the generalization of the protocol. Furthermore, the explicit statement of the number of atoms in the ring through the **cycN-** prefix provides an immediate idea of ring size which is lacking from SMILES.

As for the simplification of common monodentate substituents (step **2**), which we already commented for the ligand-database situation, we have:

Table 2. Percentage of substitutions made in step 2 of the algorithm for the GDB-10 molecules.

Substituent	% of struc.		
–CN	5.97	$–CF_3$	0.24
–NC	2.44	$–OCH_3$	3.49
–tBu	0.33	$–CONH_2$	1.69
–Ph	0.01	–COOH	1.31
$–NO_2$	0.11	$–CCl_3$	0.02

We see that the most common motif is the nitrile group: combining the two possible CN and NC forms, more than 8%

of the named molecules contain this substructure. The rest of the queries in the pre-simplification stage are much rarer in this specific dataset: this is particularly notorious for groups like Ph or CF_3, which were quite abundant along the dataset used in **case 1**, originally built considering the *most common* substituents along structures in the ChemBL. However, we must recall the constraints of the dataset: as GDB-10 only considers molecules with up to 10 non-hydrogenic atoms, relatively large groups (6 non-H atoms in phenyl, 4 in CF_3 or tBu) do not allow for as many combinations as the smaller groups like the nitrile or the methoxy (second most abundant group, spanning 3.5% of entries). More so, the kind of brute-force approach to explore the chemical space employed to build this kind of datasets does not consider the stability or commonness of structures, also contributing to the large presence of rare heterocycles and substituents instead of ubiquitous chemical motifs like the benzene ring. In light of this, the capability of the current procedure to provide sensible names for complex structures stands out.

CONCLUSIONS

The HumanSMILES strategy provides a tool to generate human-readable names for small molecules that can be easily embedded in more complex workflows. The improvement on readability allows to make cheminformatic methods and results more understandable for the general chemical community, contributing to facilitate the collaboration between groups with different backgrounds and providing an alternative asset for communicating results.

The current Python-based implementation has already been demonstrated to generate reasonable names for very different structures. Due to its open-source nature and its design, each user may adequate the details of the naming process to their specific needs, e.g. adding additional base strings to the substitution libraries.

The first use case demonstrates the good performance of HumanSMILES for a small and precurated dataset including a few thousands of very common structural fragments. Not only a high naming ratio (>80%) is attained despite the apparent simplicity of the protocol, but also the detailed analysis on the percentages of functional group substitutions and cycle types highlights the adequacy of the proposed rules to handle usual chemical structures.

The application to the GDB-10 database shows the generality of the protocol, achieving a good naming ratio (70%) for a much larger dataset on the range of millions of molecules. At this much larger scale, it becomes possible to assess the language-related aspects of HumanSMILES, showing how the generated names fall right in the middle between the very compact SMILES and their very long expanded form including explicit hydrogens, finding a good balance between readability and practicality.

All in all, the combination of human knowledge (inner libraries of common substituents and cycle motifs) and automated processing (cycle location and vacant position detection) lying at the core of HumanSMILES shows as the major factor contributing to the overall versatility of the tool.

DATA & SOFTWARE AVAILABILITY

The HumanSMILES Python code is available free of charge at GitLab https://gitlab.com/dgarayr/humansmiles

The demo web application is hosted at the Heroku platform and can be accessed at https://humansmiles-web.herokuapp.com/

The datasets employed in the two use cases are not redistributed in the repository, but can be found at:

1. https://github.com/peter-ertl/craigplot
2. https://gdb.unibe.ch/downloads/

ACKNOWLEDGMENT

The authors thank the ICIQ Foundation, CERCA Program and AGAUR (grant 2017SGR00290) of the Generalitat de Catalunya, and the Spanish Ministerio de Ciencia, Innovacion y Universidades through project CTQ2017-88777-R and CTQ2020-XXYYY-Z for financial support. D.G.R. thanks AGAUR, the Secretaria d'Universitats i Recerca of the Generalitat de Catalunya and the European Social Fund for a FI predoctoral grant. We also thank M. Álvarez-Moreno for his help and guidance on the web application development.

REFERENCES

(1) Favre, H. A.; Powell, W. H. *Nomenclature of Organic Chemistry: IUPAC Recommendations and Preferred Names 2013*; Favre, H. A., Powell, W. H., Eds.; Royal Society of Chemistry: Cambridge, 2014.

(2) Weininger, D. SMILES, a Chemical Language and Information System: 1: Introduction to Methodology and Encoding Rules. *Journal of Chemical Information and Computer Sciences* **1988**, *28* (1), 31–36. https://doi.org/10.1021/ci00057a005.

(3) Weininger, D.; Weininger, A.; Weininger, J. L. SMILES. 2. Algorithm for Generation of Unique SMILES Notation. *Journal of Chemical Information and Computer Sciences* **1989**, *29* (2), 97–101. https://doi.org/10.1021/ci00062a008.

(4) James, C. A. OpenSMILES specification http://opensmiles.org/opensmiles.html (accessed Mar 12, 2021).

(5) Heller, S. InChI – the Worldwide Chemical Structure Standard. *Journal of Cheminformatics* **2014**, *6* (S1), 1–9. https://doi.org/10.1186/1758-2946-6-s1-p4.

(6) Kim, S.; Chen, J.; Cheng, T.; Gindulyte, A.; He, J.; He, S.; Li, Q.; Shoemaker, B. A.; Thiessen, P. A.; Yu, B.; Zaslavsky, L.; Zhang, J.; Bolton, E. E. PubChem in 2021: New Data Content and Improved Web Interfaces. *Nucleic Acids Research* **2021**, *49* (D1), D1388–D1395. https://doi.org/10.1093/nar/gkaa971.

(7) Gaulton, A.; Bellis, L. J.; Bento, A. P.; Chambers, J.; Davies, M.; Hersey, A.; Light, Y.; McGlinchey, S.; Michalovich, D.; Al-Lazikani, B.; Overington, J. P. ChEMBL: A Large-Scale Bioactivity Database for Drug Discovery. *Nucleic Acids Research* **2012**, *40* (D1), 1100–1107. https://doi.org/10.1093/nar/gkr777.

(8) Fink, T.; Bruggesser, H.; Reymond, J. L. Virtual Exploration of the Small-Molecule Chemical Universe below 160 Daltons. *Angewandte Chemie - International*

Edition **2005**, *44* (10), 1504–1508. https://doi.org/10.1002/anie.200462457.

(9) Fink, T.; Raymond, J. L. Virtual Exploration of the Chemical Universe up to 11 Atoms of C, N, O, F: Assembly of 26.4 Million Structures (110.9 Million Stereoisomers) and Analysis for New Ring Systems, Stereochemistry, Physicochemical Properties, Compound Classes, and Drug Discove. *Journal of Chemical Information and Modeling* **2007**, *47* (2), 342–353. https://doi.org/10.1021/ci600423u.

(10) Ruddigkeit, L.; van Deursen, R.; Blum, L. C.; Reymond, J. L. Enumeration of 166 Billion Organic Small Molecules in the Chemical Universe Database GDB-17. *Journal of Chemical Information and Modeling* **2012**, *52* (11), 2864–2875. https://doi.org/10.1021/ci300415d.

(11) Ramakrishnan, R.; Dral, P. O.; Rupp, M.; von Lilienfeld, O. A. Quantum Chemistry Structures and Properties of 134 Kilo Molecules. *Scientific Data* **2014**, *1*, 1–7. https://doi.org/10.1038/sdata.2014.22.

(12) Schlexer Lamoureux, P.; Winther, K. T.; Garrido Torres, J. A.; Streibel, V.; Zhao, M.; Bajdich, M.; Abild-Pedersen, F.; Bligaard, T. Machine Learning for Computational Heterogeneous Catalysis. *ChemCatChem* **2019**, *11* (16), 3581–3601. https://doi.org/10.1002/cctc.201900595.

(13) Toyao, T.; Maeno, Z.; Takakusagi, S.; Kamachi, T.; Takigawa, I.; Shimizu, K. I. Machine Learning for Catalysis Informatics: Recent Applications and Prospects. *ACS Catalysis* **2020**, *10* (3), 2260–2297. https://doi.org/10.1021/acscatal.9b04186.

(14) Ahneman, D. T.; Estrada, J. G.; Lin, S.; Dreher, S. D.; Doyle, A. G. Predicting Reaction Performance in C–N Cross-Coupling Using Machine Learning. *Science* **2018**, *360* (6385), 186–190. https://doi.org/10.1126/science.aar5169.

(15) Funes-Ardoiz, I.; Schoenebeck, F. Established and Emerging Computational Tools to Study Homogeneous Catalysis—From Quantum Mechanics to Machine Learning. *Chem* **2020**, *6* (8), 1904–1913. https://doi.org/10.1016/j.chempr.2020.07.008.

(16) Strieth-Kalthoff, F.; Sandfort, F.; Segler, M. H. S.; Glorius, F. Machine Learning the Ropes: Principles, Applications and Directions in Synthetic Chemistry. *Chemical Society Reviews* **2020**, *49* (17), 6154–6168. https://doi.org/10.1039/c9cs00786e.

(17) Fey, N. Lost in Chemical Space? Maps to Support Organometallic Catalysis. *Chemistry Central Journal* **2015**, *9* (1), 1–10. https://doi.org/10.1186/s13065-015-0104-5.

(18) O'Boyle, N.; Dalke, A. DeepSMILES. *ChemRxiv* **2018**, 7097960.v1.

(19) Krenn, M.; Häse, F.; Nigam, A.; Friederich, P.; Aspuru-Guzik, A. Self-Referencing Embedded Strings (SELFIES): A 100% Robust Molecular String Representation. *Machine Learning: Science and Technology* **2020**, *1* (4), 045024. https://doi.org/10.1088/2632-2153/aba947.

(20) Kluyver, T.; Ragan-Kelley, B.; Pérez, F.; Granger, B.; Bussonnier, M.; Frederic, J.; Kelley, K.; Hamrick, J.; Grout, J.; Corlay, S.; Ivanov, P.; Avila, D.; Abdalla, S.; Willing, C.; development team, J. Jupyter Notebooks - a Publishing Format for Reproducible Computational Workflows. In *Positioning and Power in Academic Publishing: Players, Agents and Agendas*; Loizides, F., Scmidt, B., Eds.; IOS Press: Netherlands, 2016; pp 87–90.

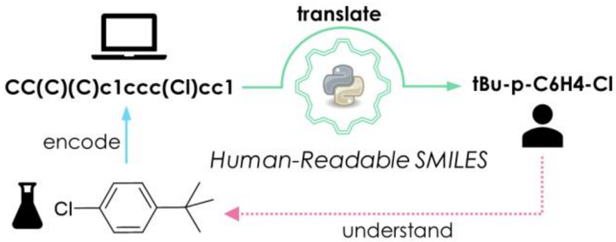
(21) Garay-Ruiz, D. HumanSMILES https://gitlab.com/dgarayr/humansmiles (accessed Mar 12, 2021).

(22) Landrum, G. RDKit: Open-source Cheminformatics.

(23) Garay-Ruiz, D. HumanSMILES WebApp https://humansmiles-web.herokuapp.com/ (accessed Mar 17, 2021).

(24) Ertl, P. Craig Plot 2.0: An Interactive Navigation in the Substituent Bioisosteric Space. *Journal of Cheminformatics* **2020**, *12* (1), 10–15. https://doi.org/10.1186/s13321-020-0412-1.

Table of Contents graphic


translate
CC(C)(C)c1ccc(Cl)cc1
tBu-p-C6H4-Cl
encode
Human-Readable SMILES
Cl
understand