Supplementary materials for MolGrapher: Graph-based Visual Recognition of Chemical Structures

Lucas Morin[1, 2] Martin Danelljan[2] Maria Isabel Agea[1] Ahmed Nassar[1]
Valery Weber[1] Ingmar Meijer[1] Peter Staar[1] Fisher Yu[2]
[1]IBM Research [2]ETH Zurich
{lum, ahn, vwe, inm, taa}@zurich.ibm.com martin.danelljan@vision.ee.ethz.ch i@yf.io

Here, we provide additional details and visualizations regarding: the synthetic training set in section 1, the benchmark dataset USPTO-30K in section 2, the MolGrapher implementation in section 3, and the MolGrapher evaluation in section 4. Table 1

1. Synthetic Training Set details

In this section, we describe the synthetic training set augmentations as well as the generation of atom-level annotations. Contrary to the image captioning methods, our solution benefits from atom-level annotations, allowing to train with much less examples. In particular, MolGrapher is trained on 0.3 M images while Image2Graph [8] uses 7.1 M images, and DECIMER 2.0 [7] uses 450 M images.

1.1. Augmentation

The synthetic training set is augmented at the molecule, rendering and image levels.

Molecule level. As presented in Figure 1, molecules are randomly transformed by: (1) displaying explicit hydrogens, (2) reducing of the size of bonds connected to explicit hydrogens, (3) displaying explicit methyls, (4) displaying explicit carbons, (5) selecting a molecular conformation, (6) removing implicit hydrogens of atom labels, (7) rotating triple bonds, (8) displaying explicit carbons connected to triple bonds, adding artificial superatom groups with (9) single or (10) multiple attachment points, (11) displaying wedge bonds using solid or dashed bonds, and (12) displaying single bonds as wavy bonds.

Table 1. **USPTO-30K comparison with other benchmarks.** We compare the number of samples, the number of classes, the molecule sizes and the proportion of molecules with stereochemistry.

Dataset	Number of samples	Molecule sizes		Stereo-chemistry proportion	Number of classes		
		[Min, Max]	Mean		Atom	Bond	Superatom
USPTO	5719	[10, 96]	28	20.5	24	6	234
Maybridge UoB	5740	[4, 34]	2.6	13	20	6	0
CLEF	992	[4, 42]	26	33.9	15	6	43
JPO	450	[5, 43]	20	1.0	9	6	15
USPTO-30K	**30000**	**[4, 543]**	**53**	**39.2**	**74**	**6**	**620**
USPTO-10K	10000	[4, 198]	31	29.3	37	6	0
USPTO-10K-Abb	10000	[4, 162]	31	31.2	45	6	620
USPTO-10K-L	10000	[71, 543]	96	57.1	52	6	0

Rendering level. As demonstrated in Figure 2, the rendering parameters used in RDKit [6] are randomly set: (1) the bond width, (2) the font, (3) the font size, (4) the atom label padding, (5) the molecule rotation, which does not rotate atom labels, (6) the display of atom indices and (7) their font size, (8) the hand-drawing style, (9) the charges positions, (10) the display of encircled charges and (11) their size, and (12) the display of aromatic cycles using circles.

Image level. As showcased in Figure 3, images undergo several image augmentations on the fly: (1) the addition of random captions, (2) the addition of random lines, (3) the addition of pepper patches, (4) rotation, scaling, shifting, (5) resolution downscaling, (6) gaussian blurring and (7) x-y shearing. Finally, images are inverted in order to be compatible with the zero-padding used by default for convolutional layers. The severity of augmentations is different for training the keypoint detector or the node classifier. Indeed, to only detect atoms positions, the keypoint detector does not need to precisely distinguish atom labels.

1.2. Atom-level Annotation

Together with the training images, we generate the graph ground-truth, i.e. the graph connectivity, the atoms and bonds labels, and their positions. RDKit allows to embed to the generated image some metadata, which stores the mapping between atom indices and their positions in the image. At the same time, we store a MolFile [3] containing the graph connectivity information and the class of each atom and bond. By combining both of them, we then create the graph ground-truth.

2. USPTO-30K Statistics and Visual Examples

In this section, we analyze the distribution of molecules in the dataset USPTO-30K, we compare its composition with other benchmarks, and visualize examples.

2.1. Statistics

Molecule sizes. Figure 4 presents the distribution of molecule sizes in USPTO-30K, the size of a molecule being

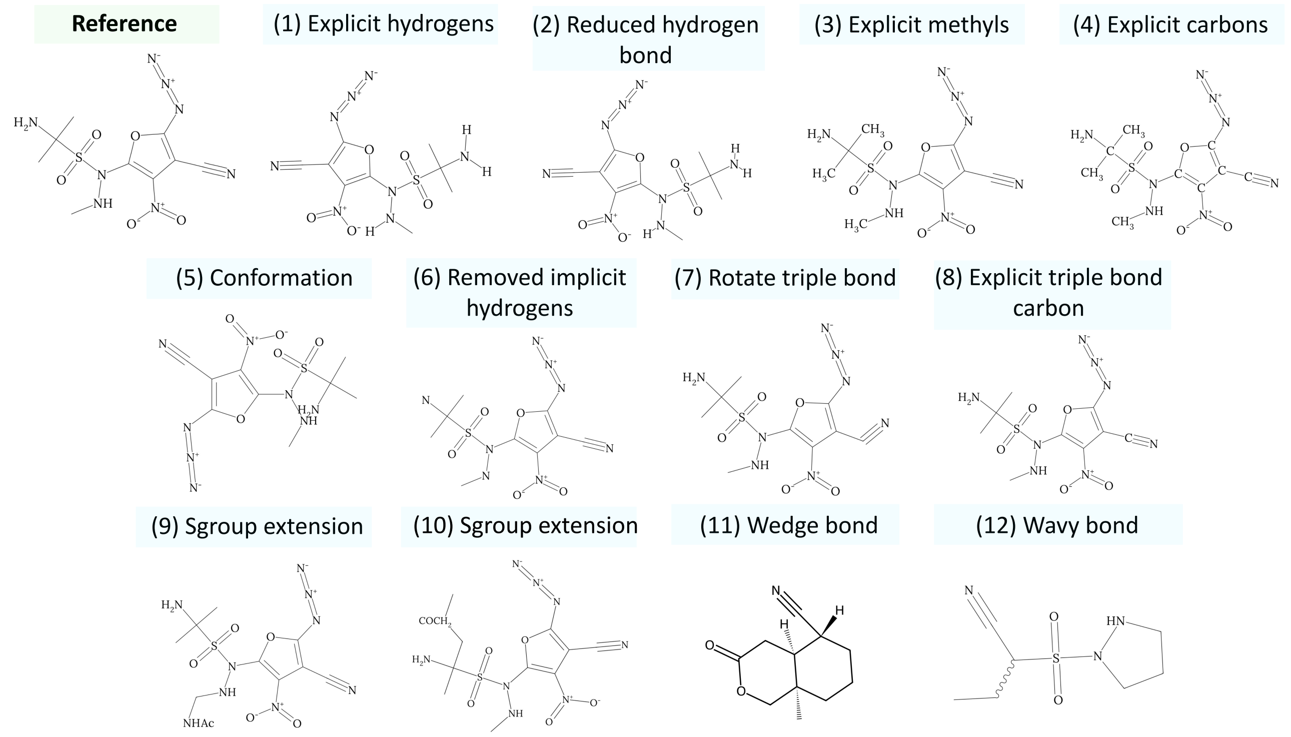

Figure 1. **Molecule augmentations.** The figure illustrates all molecule augmentations applied independently on a reference molecule.

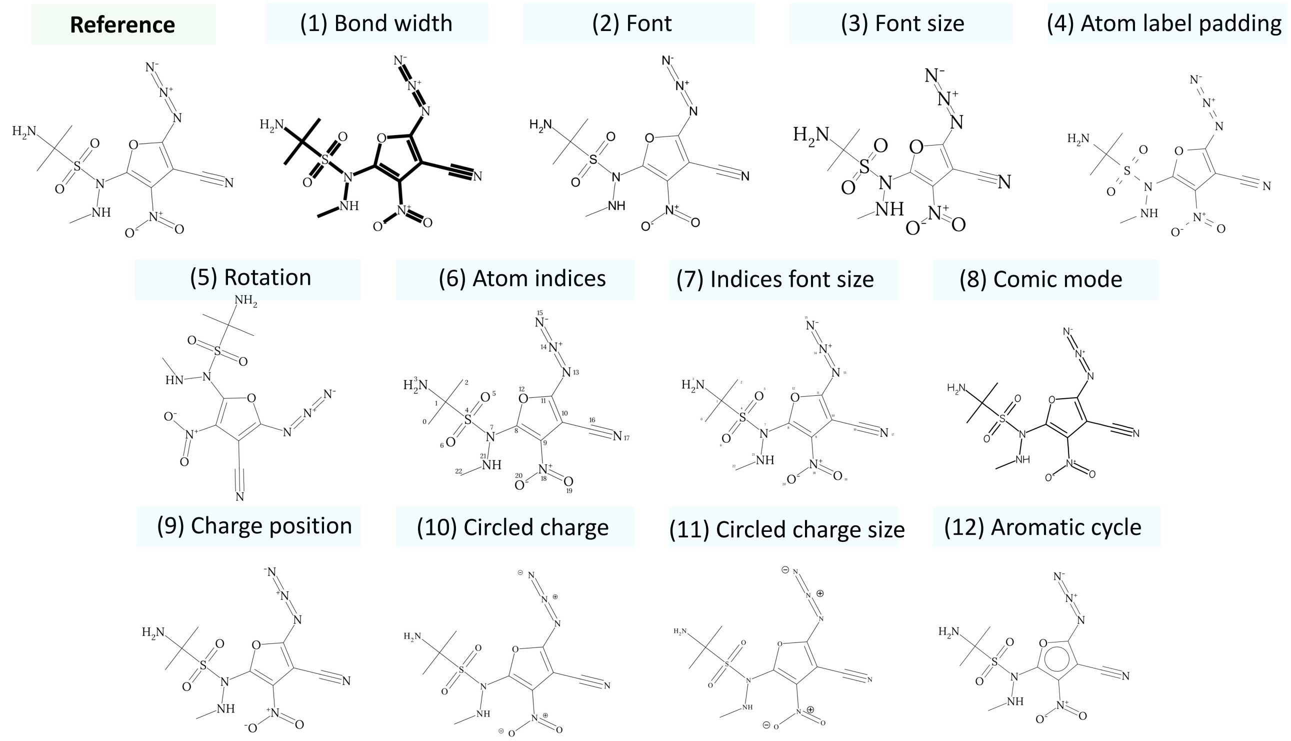

Figure 2. **Rendering augmentations.** The figure describes all rendering augmentations applied independently on a reference drawing.

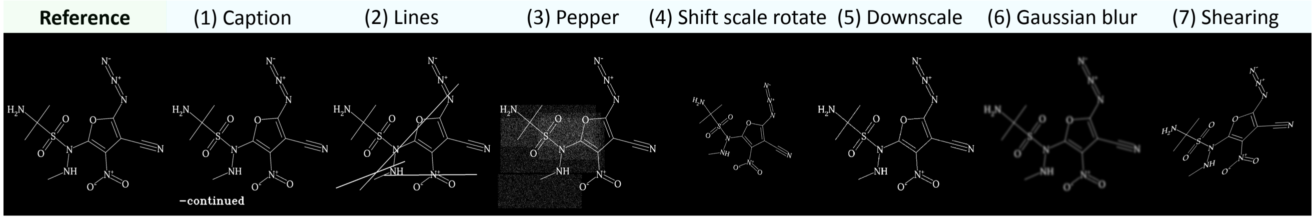

Figure 3. **Image augmentations.** The figure illustrates all image augmentations applied independently on a reference image. Images are inverted in order to be compatible with the zero-padding used by default for convolutional layers.

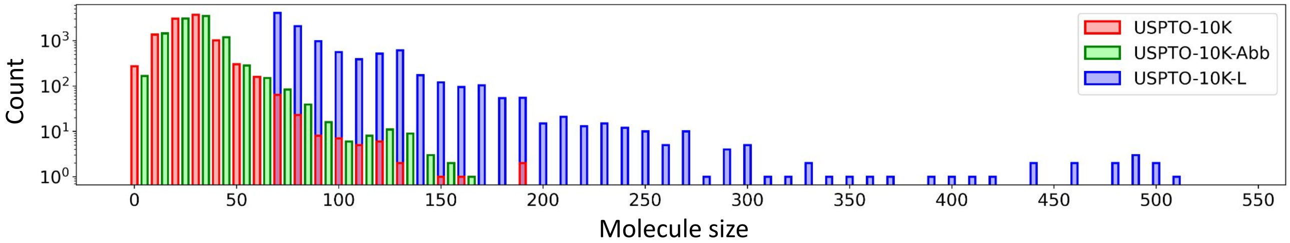

Figure 4. **USPTO-30K molecule sizes distribution.** The figure illustrates the distribution of molecule sizes (number of atoms) in USPTO-10K (red), USPTO-10K-Abb (green) and USPTO-10K-L (blue). The population of each size bin is expressed in logarithmic scale.

defined as its number of atoms.

Atoms and superatom groups. Figure 5 and Figure 6 show the distribution of atom classes and abbreviations in USPTO-30K. Randomly sampling images from all US patents from 2001 to 2020 allows us to cover a large diversity of atoms and superatom groups, including common but also exotic ones.

Comparison with other benchmarks. Table 1 presents the comparison of the compositions of USPTO-30K and other benchmark datasets. USPTO-30K also contains more samples, disentangles the study of clean and abbreviated molecules. In comparison with existing sets, USPTO-30K contains more samples, a greater range of molecule sizes, more than 3× as many atom classes, 2× as many superatom classes, and a higher proportion of molecules with stereochemistry information.

2.2. Visualization

Figure 7 illustrates some examples of images randomly sampled from USPTO-30K.

3. Implementation details

Atom and bond classes. The recognized atom classes are ‘no atom’, ‘C’, ‘N’, ‘O’, ‘S’, ‘F’, ‘Cl’, ‘P’, ‘Br’, ‘I’, ‘B’, ‘Si’, ‘Sn’, ‘Te’, ‘Sb’, ‘Bi’, ‘Se’, ‘Al’, ‘As’, ‘W’, ‘Hg’, ‘Ge’, ‘In’, ‘Na’, ‘Pb’, ‘Mg’, ‘Pt’, ‘Tl’, ‘Fe’, ‘Ru’, ‘Cr’, ‘Li’, ‘Ar’, ‘Pd’, ‘Zr’, ‘Zn’, ‘Mo’, ‘Xe’, ‘U’, ‘Po’, ‘Ni’, ‘K’, ‘Cs’, ‘At’, ‘Yb’, ‘Ti’, ‘Tc’ and ‘Os’. The recognized bond classes are ‘no bond’, ‘single’, ‘double’, ‘triple’, ‘wedge-solid’, ‘wedge-dashed’ and ‘aromatic’.

Superatom groups recognition. Before applying PP-OCR [4], the molecule images are preprocessed by removing some of the bonds to simplify the text labels recognition. This is done by removing larger clusters of continuous filled pixels. Additionally, the mapping between recognized superatoms and submolecules is automatically created using the MolFiles provided by the United States Patent and Trademark Office (USPTO) [1]. In total, 819 abbreviations can be recognized, the most common ones being: ‘Me’, ‘CF3’, ‘CN’, ‘NC’, ‘OMe’, ‘Boc’, ‘OCH3’, ‘NO2’, ‘COOH’, ‘Ph’, ‘CO2H’, ‘O2N’, ‘H3CO’, ‘OEt’, ‘OCF3’, ‘NHBoc’, ‘N3’, ‘Et’, ‘HOOC’, ‘OBn’, ‘B(OH)2’, ‘CHF2’, ‘CO2Me’, ‘F3C’, ‘OAc’ and ‘t-Bu’. Abbreviations also include R-groups labels such as X’, ‘Y’, ‘Z’, ‘R1’, ‘R2‘ or ‘R10’.

Stereo-chemistry recognition. In 2-D molecule depictions, stereo-chemistry is represented using solid or dashed wedge bonds, which respectively point towards or away from the viewer. To recognize wedge bonds, the model first predicts a node with a class ‘wedge-solid’ or ‘wedge-dashed’. It is then needed to identify the direction of this bond. For this purpose, the ratio of filled pixels on both sides of the bond is computed. The side with the smallest amount of filled pixels is inside the depiction plane and the other one is outside.

Caption removal. Although MolGrapher is trained with synthetic images containing random captions, it may not cover the complexity encountered during inference, such as Japanese captions. Thus, the OCR toolkit PP-OCR [4] is used to detect and remove captions in images. Detected text cells which respect some criteria are replaced by a white

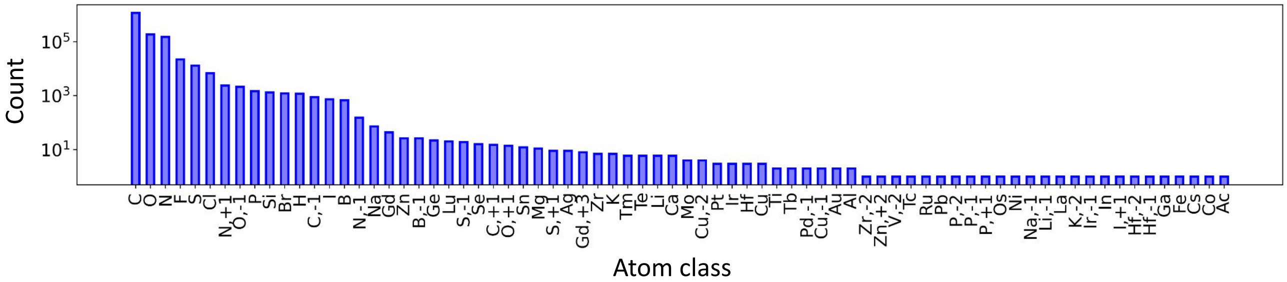

Figure 5. **USPTO-30K atom classes distribution.** The figure shows the distribution of the atom classes in USPTO-30K. The count of each atom class is expressed in logarithmic scale.

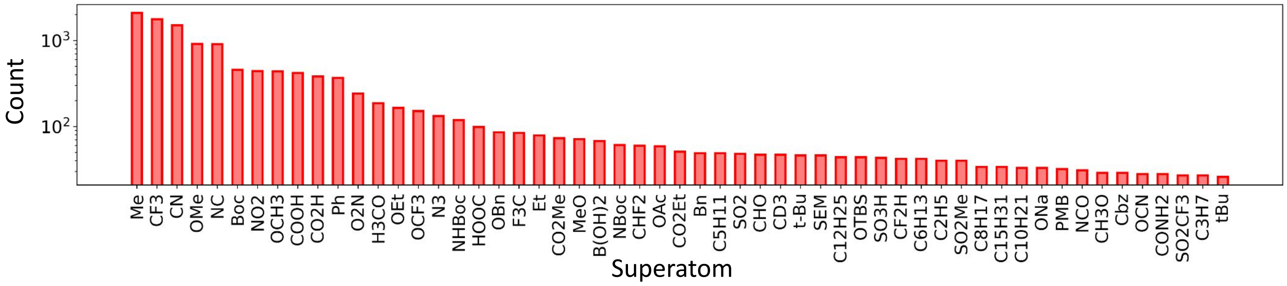

Figure 6. **USPTO-10K-Abb abbreviations distribution.** The figure shows the distribution of the superatoms in USPTO-10K-Abb. Only superatoms with more than 25 occurrences are displayed. The count of each superatom is expressed in logarithmic scale.

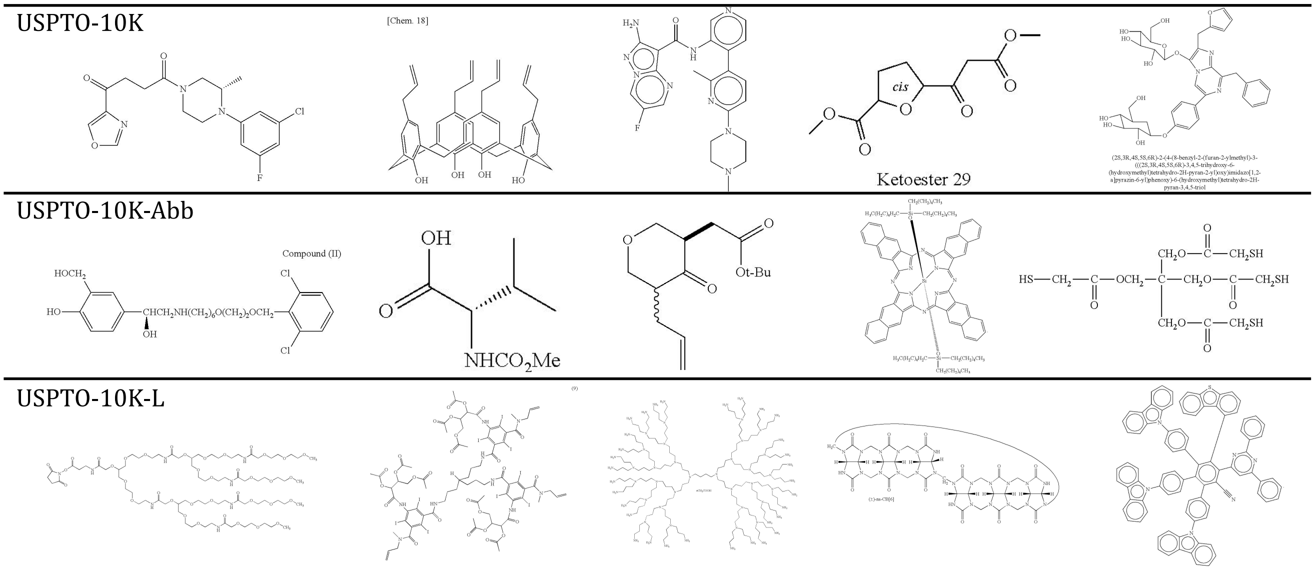

Figure 7. **USPTO-30K example images.** The figure shows example images randomly selected in USPTO-10K, USPTO-10K-Abb and USPTO-10K-L.

background. For instance, if a recognized text sequence contains more than five consecutive lowercase characters, it should necessarily be a caption.

Keypoint refinement. PP-OCR is also used to refine keypoint predictions for superatom groups having multiple attachment points, i.e. connected to the rest of the molecule by multiple bonds. The keypoint detector learns to detect a keypoint at the extremities of each bond. For long abbreviated groups with multiple attachment points, each of its outgoing bond can be located at different positions along the abbreviation. This situation results in several detected keypoints for the same superatom. Therefore, if the initial prediction is an invalid molecule, PP-OCR is used to merge keypoints that are located within the same detected text cell.

Inference speed. On a machine equipped with one

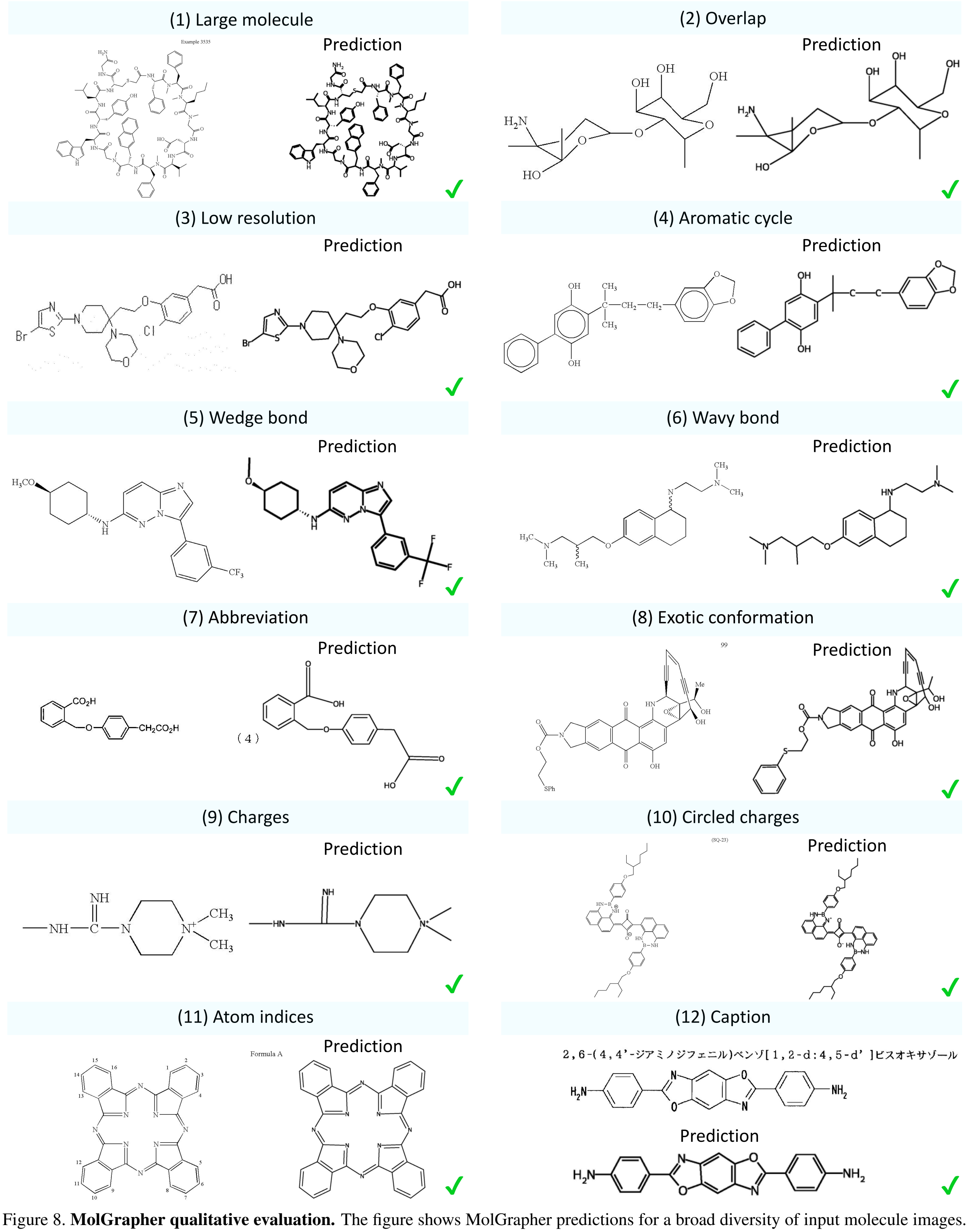

Figure 8. **MolGrapher qualitative evaluation.** The figure shows MolGrapher predictions for a broad diversity of input molecule images. Input images (left) are displayed together with predicted molecules (right). The model can correctly handle challenging features, such as large molecule or overlapping bonds.

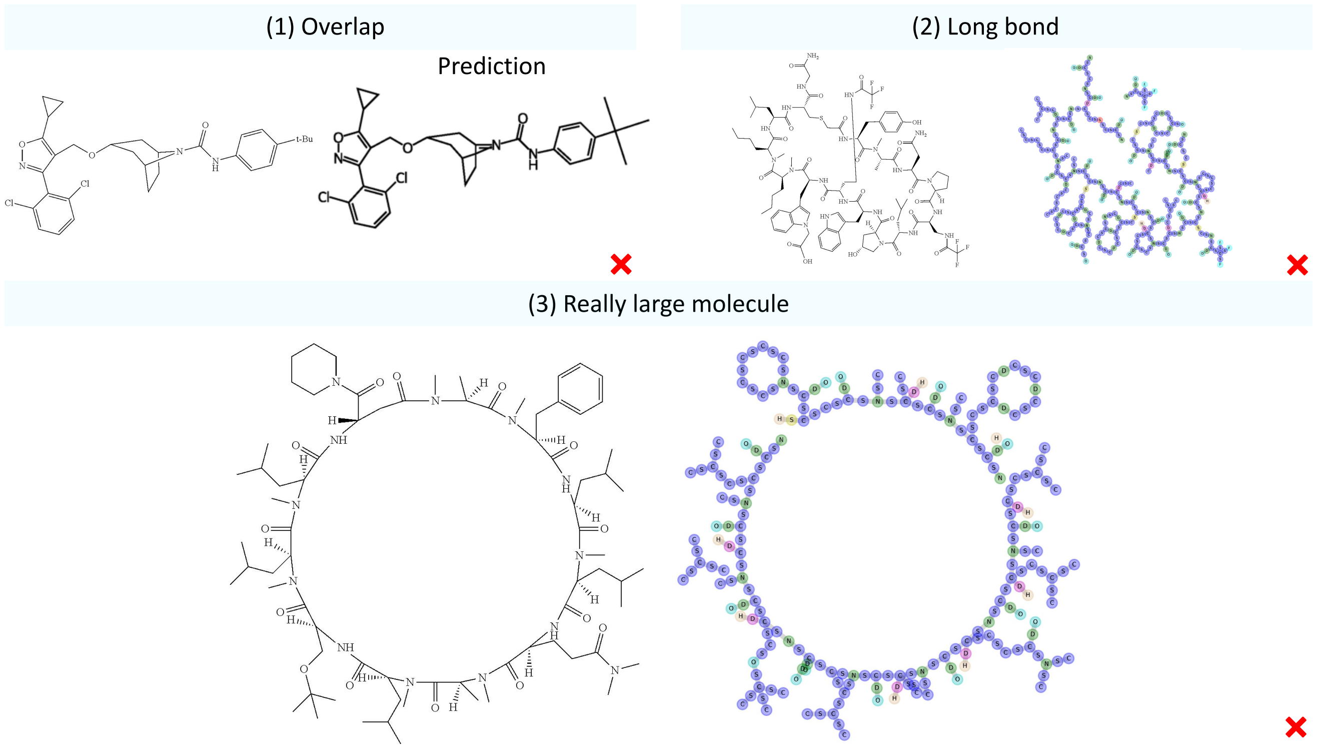

Figure 9. **MolGrapher failure cases.** The figure illustrates some examples of failure cases of MolGrapher. Input images (left) are displayed together with predicted molecules (right). The predicted graph (right) is displayed when it can not be converted to a valid single-fragment molecule.

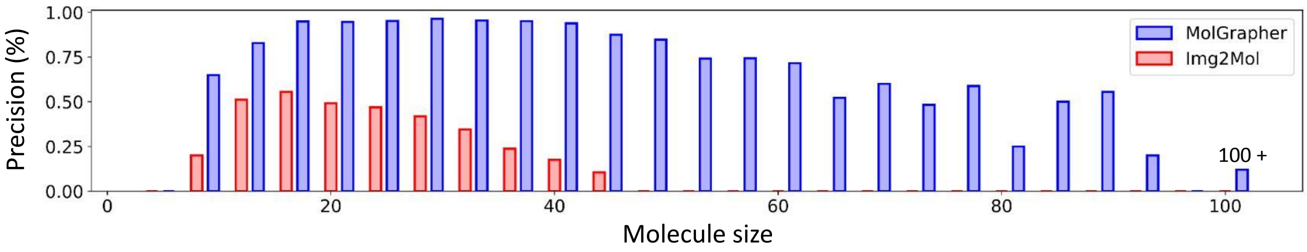

Figure 10. **USPTO-10K performance analysis.** The figure details the performance of MolGrapher (blue) and Img2Mol (red) on USPTO-10K with respect to the molecule size.

NVIDIA A100 GPU and AMD EPYC 7763 CPU, processing images by batches of 20, MolGrapher annotates an average of 3.1 images per second. This runtime measurement was obtained by running MolGrapher on USPTO-30K.

4. MolGrapher detailed analysis

4.1. Qualitative evaluation

Qualitative examples. Figure 8 shows a sample of predictions for challenging input images. In practice, MolGrapher can be used to annotate low-quality images, unconventional drawings, images noised by additional information such as captions, atom indices, or stereo-chemistry annotations. MolGrapher recognizes abbreviated groups, stereochemistry and aromaticity.

Failure cases. A sample of failure cases is presented in Figure 9. In particular, example (1) is a molecule with challenging overlaps, leading to an incorrect double bond prediction. Example (2) shows a large molecule that contains a particularly long bond, which also overlaps with the molecule. This long bond is not detected because the supergraph construction algorithm discards it, according to its threshold of maximum bond length. Modifying the supergraph construction and generating bond overlaps, and long bond augmentations in the synthetic training, could mitigate these issues. Example (3) is an exceedingly large molecule with more than 110 atoms. Although most of the structure is correctly recognized, few bonds are occasionally miss-

Table 2. **SOTA comparison with same training sets.** We evaluate the performance of MolGrapher by training it on various synthetic datasets of different sizes and comparing it to different methods. †: without hyper-parameters tuning.

Method	Synthetic training set	USPTO	Maybridge-UoB	CLEF-2012	JPO
CEDe	CEDe @ 10K	79.0	74.1	68.0	49.4
MolGrapher	CEDe @ 10K	**84.5**	**89.8**	**79.6**	**59.2**
MolGrapher	MolGrapher @ 10K	**87.5**	**93.8**	**86.2**	**61.5**
Graph Generation †	MolGrapher @ 300K	31.3	71.6	25.3	24.2
Img2Mol †	MolGrapher @ 300K	15.5	23.2	10.0	11.3
MolGrapher	MolGrapher @ 300K	**91.5**	**94.9**	**90.5**	**67.5**

Table 3. **Ablation study of MolGrapher modules.** To isolate the errors stemming from each component, we replace the other two components with Ground-Truth (GT) oracle predictions.

Keypoint detection	Node classification	Superatom recognition	Synthetic test set @ 10K
Ours	**Ours**	**Ours**	94.3
Ours	GT	GT	**95.8**
GT	**Ours**	GT	97.2
GT	GT	**Ours**	98.1

ing. Further post-processing rules could be implemented to force the attachment of unconnected fragments.

4.2. Quantitative evaluation

SOTA comparison on same training data. To provide a comprehensive fair comparison with the available methods in Table 2. We trained MolGrapher on the synthetic training set of CEDe [5]. MolGrapher still outperforms CEDe on all benchmarks by a large margin. To assess the contribution of our training data, we also train MolGrapher on the same number (10k) of images as CEDe generated using our pipeline (resulting in 30× less data). We then trained open-source methods on our full dataset (300k). Our method outperforms them by a substantial margin.

Ablation study. We perform an ablation study on MolGrapher's components in Table 3. In order to isolate the errors induced by each component, we replace the other two components with Ground-Truth (GT) oracle predictions. We use a synthetic test set of 10k images, since the ground-truth graph with keypoint locations are not available on real datasets. We observe that the main source of errors is the keypoint detection, followed by the node classification.

4.3. Molecule Size Analysis

Figure 10 shows the precision of MolGrapher and Img2Mol [2] on USPTO-10K with respect to the molecule size. MolGrapher maintains a reasonable performance even for remarkably large molecules of 90 atoms. The locality of our approach allows to scale with respect to the molecule size. On the other hand, Img2Mol fails to recognize correctly any molecule of more than 50 atoms.

4.4. Error Prediction

Contrary to image captioning methods, which are trained to always generate valid SMILES sequences, our model identifies incorrect predictions in many cases. Table 4 presents the 'detected error rate' of MolGrapher, i.e. the proportion of errors that can be detected among the total number of errors. Indeed, by performing low-level predictions, we can detect that the predicted graph is not convertible into a valid single-fragment molecule. Table 4 also reports the 'filtered precision', which is the precision of MolGrapher on the filtered benchmarks, in which the detected errors are not considered. To annotate scientific literature at large scale, and extract a viable source of knowledge, identifying incorrect predictions is critical.

Table 4. **MolGrapher error detection.** We evaluate the 'detected error rate', i.e. the proportion of MolGrapher errors that can be detected because the predicted graph is not convertible to a valid single-fragment molecule. The 'filtered precision' is the precision computed on the filtered benchmarks, in which detected errors are not considered.

MolGrapher	USPTO-10K	USPTO-10K-Abb	USPTO-10K-L	JPO
Detected error rate	20.3	15.1	52.5	40.1
Precision	93.3	82.8	31.3	67.5
Filtered precision	**94.7**	**85.4**	**67.4**	**80.5**

5. Limitations and Future Works

Currently, MolGrapher is unable to recognize markush structures, i.e. depictions of sets of molecules using positional and frequency variation indications. As future work, we aim to generalize the MolGrapher graph structure to represent markush structures or even reactions.

References

[1] United states patent and trademark office. http://uspto.gov. Accessed: 1 January 2023.

[2] Djork-Arné Clevert, Tuan Le, Robin Winter, and Floriane Montanari. Img2mol – accurate smiles recognition from molecular graphical depictions. Chem. Sci., 12:14174–14181, 9 2021.

[3] Arthur Dalby, James G. Nourse, W. Douglas Hounshell, Ann K. I. Gushurst, David L. Grier, Burton A. Leland, and John Laufer. Description of several chemical structure file formats used by computer programs developed at molecular design limited. Journal of Chemical Information and Computer Sciences, 32(3):244–255, May 1992.

[4] Yuning Du, Chenxia Li, Ruoyu Guo, Xiaoting Yin, Weiwei Liu, Jun Zhou, Yifan Bai, Zilin Yu, Yehua Yang, Qingqing Dang, and Haoshuang Wang. PP-OCR: A practical ultra lightweight OCR system. CoRR, abs/2009.09941, 2020.

[5] Rodrigo Hormazabal, Changyoung Park, Soonyoung Lee, Sehui Han, Yeonsik Jo, Jaewan Lee, Ahra Jo, Seung Hwan Kim, Jaegul Choo, Moontae Lee, and Honglak Lee. CEDe: A collection of expert-curated datasets with atom-level entity annotations for optical chemical structure recognition. In Sanmi Koyejo, S. Mohamed, A. Agarwal, Danielle Belgrave, K. Cho, and A. Oh, editors, Thirty-sixth Conference on Neural Information Processing Systems Datasets and Benchmarks Track, 2022.

[6] Greg Landrum. Rdkit: Open-source cheminformatics software. http://www.rdkit.org/. Accessed: 1 January 2023.

[7] Kohulan Rajan, Henning Otto Brinkhaus, M Isabel Agea, Achim Zielesny, and Christoph Steinbeck. Decimer.ai - an open platform for automated optical chemical structure identification, segmentation and recognition in scientific publications. feb 2023.

[8] Sanghyun Yoo, Ohyun Kwon, and Hoshik Lee. Image-to-graph transformers for chemical structure recognition. In ICASSP 2022 - 2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pages 3393–3397, 2022.