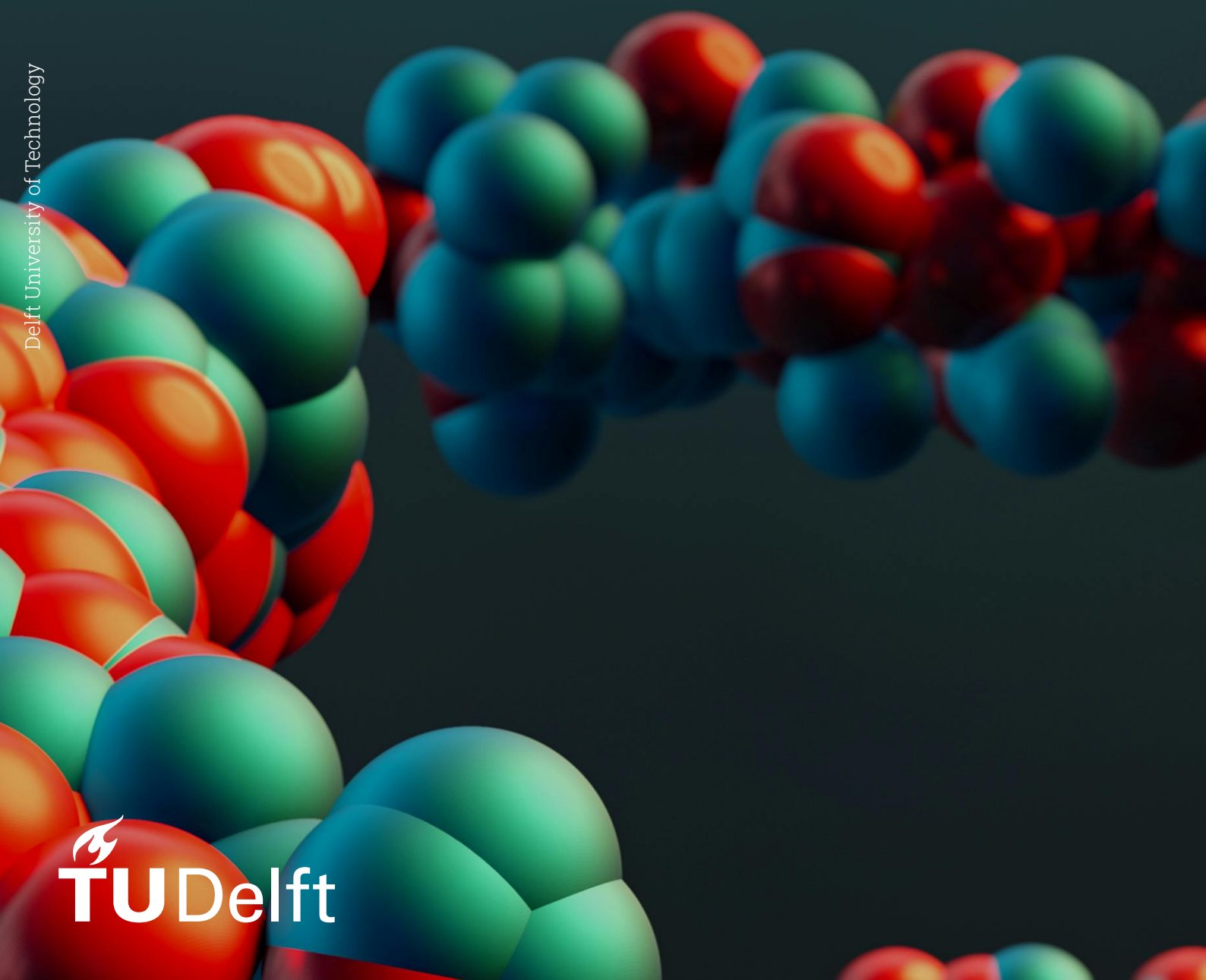
Improving Chemical Reaction Completion using Atom-Balance Constraints in Transformer Models

Master's Thesis

Minouk Tianne Willemieke Noordsij



Delft University of Technology

TU Delft

Improving Chemical Reaction Completion using Atom-Balance Constraints in Transformer Models

by

Minouk Tianne Willemieke Noordsij

to obtain the degree of Master of Science
at Delft University of Technology,
to be defended publicly on
June 27th, 2025 at 14:00.

Student number:	4788338
Project duration:	November 2024 - June 2025
Thesis committee:	
Prof. dr. ir. M.J.T. Reinders	Thesis advisor
Dr. J.M. Weber	Daily supervisor
G. Vogel, MSc	Daily co-supervisor
Dr. J. Yang	External Committee Member

TU Delft

The Delft Bioinformatics Lab

Acknowledgements

I would like to express my gratitude to Dr. Jana Weber for giving me the opportunity to conduct this research and for her supervision during the initial stage of my project. Her support and constructive feedback were of great value.

I extend my gratitude to Prof. Dr. Ir. Marcel Reinders for taking over the supervision of my project, providing me with insightful feedback and advice during our discussions.

I am particularly grateful to Gabriel Vogel for his dedicated supervision and for his constant engagement in brainstorming new ideas. His guidance and support were invaluable for conducting my research project.

Furthermore, I would like to thank Dr. Jie Yang for taking part in my thesis committee and evaluating my work.

I also wish to acknowledge my colleagues and fellow students at the Delft Bioinformatics Lab for welcoming me in the group and their offices and for showing genuine interest in my project. I have truely enjoyed doing research here.

Lastly I would like to thank my boyfriend, family and friends for their support and encouragement throughout the duration of my project.

Minouk Noordsij
Delft, June 2025

Improving Chemical Reaction Completion using Atom-Balance Constraints in Transformer Models

M.T.W. Noordsij[1], G. Vogel[1], M.J.T. Reinders[1], and J.M. Weber[1]

[1]Delft Bioinformatics Lab, Delft University of Technology, Delft, The Netherlands

June 19, 2025

Online databases contain extensive collections of (bio)chemical reactions serving as valuable resources for a variety of applications. However, these large datasets often suffer from incomplete reaction data missing, for example, co-reactants and by-products. Machine learning can help to predict these missing molecules in partial reactions. In this study, we adapt an existing transformer model to enhance its capability in completing these incomplete reactions. We retrain the model using a more diverse dataset of atom-balanced ground truth reactions and introduce both soft and hard atom-balance constraints to improve the completeness and chemical validity of the predictions. Our findings indicate that models trained with soft constraints in their loss function do not demonstrate improved balancing performance and require further tuning. Conversely, the implementation of hard atom-balance constraints during constrained beam search, where we restrict predicting tokens that violate the atom-balance of the prediction, effectively improves the performance of transformer-based models in reaction completion tasks. However, this approach also presents the risk of inaccurately balancing reactions; a limitation that is difficult to identify without chemical expertise, underscoring the necessity for reliable ground truth data to evaluate the predictions.

1. INTRODUCTION

Developments in text mining have significantly increased the availability of extensive online databases containing (bio)chemical reactions, such as the USPTO database sourced from patents, providing valuable resources for a wide range of applications [1]. These reactions can be used to train models for predictive chemistry that can predict products from reactants and vice versa, or predict reaction conditions [2, 3]. Additionally, reaction networks that are based on these reactions are used for pathway selection [4]. Moreover, they play a crucial role in sustainability assessments, employing mass-based evaluation strategies to analyse reaction efficiency and environmental impact [5, 6].

Despite the potential of these extensive datasets, they often suffer from incomplete reaction data, which particularly complicates these mass-based sustainability assessments. Many reactions lack necessary information, such as missing molecules on either the reactant or product side [7]. Additionally, stoichiometric data and contextual information about solvents, (bio)catalysts, and reaction conditions such as temperature and pressure, is often not fully available [3].

Figure 1 shows an example of a reaction with missing molecules (completion task at the top of the figure). These completion tasks can often be solved using rule-based algorithms that rely on chemical heuristics [8–10]. Curated balanced reactions can be used for training machine learning models to tackle various completion tasks. Beyond predicting small missing molecules, these models can also learn to perform forward prediction (inferring products from reactants) and retrosynthesis (inferring reactants from products).

However, a notable limitation of current machine learning algorithms is their failure to adhere to known chemical constraints,

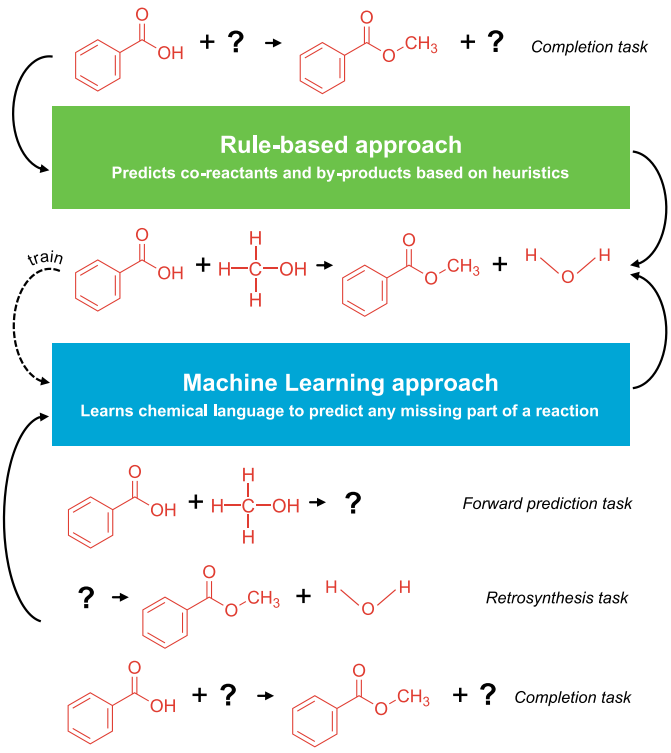


Fig. 1. Summary of chemical reaction completion methods. Rule-based or algorithmic approaches are used to curate imbalanced reaction data with small missing molecules. The curated reactions are used to train machine learning models on various reaction completion tasks, including forward prediction and retrosynthesis.

such as the atom-balance. Their architecture is not designed to keep track of the number of atoms in reactions, which can lead to atom-imbalanced predictions. Incorporating known chemical constraints into these predictive models has the potential to improve their performance and generalization capabilities [13].

A. Current approaches for reaction balancing

Multiple machine learning approaches for completing chemical reactions have already emerged. Table 1 shows an overview of these approaches. Several studies based their approach on the autoregressive encoder-decoder transformer model [7, 10, 11]. These studies use an adapted version of the Molecular Transformer, which was originally trained on forward prediction tasks [2]. Zhang et al. developed an encoder-only transformer model for reaction completion based on the BERT architecture [9]. Encoder-decoder models perform better on the reaction completion task than the encoder-only models, especially when predicting large molecules [10]. This is presumably because encoder-decoder models take into account previously predicted tokens when producing their output.

The performance of the current machine learning approaches is limited by two aspects. First, there is a lack of balanced reactions serving as ground truth data to optimally train the model. Part of the prediction tasks involve imbalanced reactions that are more complex than the ground truth data that current models are trained with. This ground truth data includes the reactions balanced by rule based methods, but these methods are not able to curate all reactions, especially in cases with complex missing molecules [9, 10]. The performance of machine learning models can be improved by training on ground truth data that includes more complex reactions. Second, the predictions by the machine learning models do not necessarily adhere to chemical constraints. These constraints include atom-balance and stoichiometry, and are ideally learned by the model, yet current models do not enforce chemical constraints, so their outputs can contain imbalanced reactions.

The USPTO dataset, which is used as ground truth dataset in current approaches, contains many imbalanced reactions [1]. However, the ground truth data should consist of a diverse and extensive number of chemically valid reactions to serve as training data for machine learning models. Existing studies address this limitation of the USPTO dataset in various ways. Several approaches used this dataset of more than 1 million entries for training the models, including the reactions with missing molecules [7, 11]. Other approaches, which train only on balanced reactions, first complete part of the imbalanced reaction SMILES to increase the number of valid ground truth reactions, following the approach outlined in Figure 1 [8–10].

Simple rule- or heuristic-based approaches can aid in identifying the missing molecules in partial reactions. These approaches are also summarized in Table 1. Various rule-based methods exist, which all employ a predetermined set of helper molecules often missing from reactions [10]. ChemBalancer, for example, initially attempts to balance the stoichiometric coefficients using a linear solver [9]. If unsuccessful, it iteratively seeks helper species on either the left- or right-hand side of the reaction. Another algorithmic strategy, specifically for carbon-imbalanced reactions, uses the maximum common subgraph (MCS) between reactants and products to identify missing molecules [8]. This strategy is combined with a rule-based approach, in which the molecules are first decomposed into ions, in a framework called SynRBL, with a machine learning model predicting the associated confidence score of the balanced output reactions. Application of these rule-based methods can expand the ground truth dataset, thus improving the performance of machine learning methods for completing more complex reactions.

B. Transformer models for predictive chemistry

The Molecular Transformer was developed for forward prediction, to predict products from reactants and reagents [2]. The Reaction Balancer is a fine-tuned version of the Molecular Transformer trained to complete partialized reactions, where molecules can be missing on both sides of the reaction [10]. Figure S3 provides an overview of the two models and the data that

Reference	Rule-based approach	Machine learning approach
Vaucher et al. (2020) [7]	None, uses uncurated data as ground truth	Modified Molecular Transformer
Zipoli et al. (2024) [11]	None, uses uncurated data as ground truth	Modified Molecular Transformer: Multi-task model for forward, retro and reagents, catalysts, and solvents (RCS) tasks
Phan et al. (2024) [8]	Rule-based approach for carbon-balanced reactions and Maximum Common Subgraph (MCS) approach for carbon-imbalanced reactions (SynRBL).	An XGBoost machine learning model is used to predict the confidence scores associated with the predictions of SynRBL.
Zhang et al. (2024) [9]	Rule-based approach using linear solver and helper species (ChemBalancer)	RoBERTa based transformer model (ChemMLM)
Van Wijngaarden et al. (2024) [10]	Rule-based approach using helper species	Modified Molecular Transformer: Model trained on curated data (Reaction Balancer)

Table 1. Summary of current approaches for reaction balancing. Rule/heuristic-based approaches are used to curate incomplete reactions. In hybrid approaches, the resulting curated reactions are added to the ground truth data [9, 10]. Machine learning approaches for completing partial reactions are based on either the autoregressive encoder-decoder Molecular Transformer [2] or encoder-only RoBERTa transformer [12] model.

was used to train them. The Molecular Transformer was trained on the USPTO STEREO dataset of 1 million reactions, which was augmented to double the size by replacing the molecules by an equivalent random SMILES representation [2]. The training data included both separated and mixed reactants and reagents and the model demonstrated robust performance for both input types. Only 3.5% of the USPTO STEREO reactions are atom-balanced, so the model is not trained to predict an atom-balanced product side, but generates only the single main product. To train the Reaction Balancer, a rule based curation algorithm was used to balance the USPTO STEREO data, which resulted in an atom-balanced dataset with 495k entries [10]. This curated data was partialized into 4 million reactions by removing part of the molecules.

Both models share the same auto-regressive transformer architecture. The input reaction SMILES are tokenized using the following regular expression, which captures at most one atom per token (when disregarding hydrogen, which is represented implicitly):

```
(\[[^ \]]+]|Br?|Cl?|N|O|S|P|F|I|b|c|n|o|s|p|\(|\)|\.|
   =|#|-|\+|\\\\\/|:|~|@|\?|>|\*|\$|\%[0-9]2|[0-9])
```

By capturing at most one atom per token, this approach facilitates the counting of predicted atoms and theoretically allows for the imposition of restrictions on the prediction of certain tokens based on atom-balance constraints.

The transformer models are trained using the Adam optimizer with cross-entropy loss and an adaptive learning rate. The cross-entropy loss for a batch of N samples is captured by Equation 1:

$$\mathcal{L}_{\text{Cross-Entropy}} = -\frac{1}{N}\Sigma_{i=1}^{N}\Sigma_{j=1}^{C}(y_{i,j} \cdot log(p_{i,j})) \quad \textbf{(1)}$$

In this equation, C denotes the number of classes, which correspond to the tokens in the vocabulary within the transformer architecture. The ground truth label for sample i and class j is stored in the variable $y_{i,j}$, which has a value of one for the token that should be predicted and zero for other tokens. The associated probability of the model predicting this label, which is the logits normalized by the softmax function, is $p_{i,j}$. Only the probability of predicting the correct label contributes to the loss function, since $y_{i,j}$ is zero for all other classes. However, this makes it impossible to distinguish between wrongly predicted tokens in those cases where some wrong predictions should be penalized more than others. The logarithm of $p_{i,j}$ is taken, which ensures a large outcome when the probability of predicting the right label is very small. This makes cross-entropy loss suitable for gradient descent methods such as Adam optimization.

During inference, the transformer models use beam search decoding. This technique maintains the n (beam size) most probable outputs during sequence generation, providing a balance between speed and accuracy. By considering the top-n outputs, beam search aims to maximize overall sequence probability as opposed to greedily taking the most probable token at each step.

C. Constrained Machine Learning

Constrained machine learning is a methodology that ensures the outputs of a model comply with specific predefined rules or limitations. This approach is particularly valuable in applications where adherence to physical, chemical, or operational constraints is essential for generating valid and reliable predictions. Within the field of constrained machine learning, a distinction can be made between soft and hard constraints.

Soft constrained machine learning preserves flexibility in the model's predictions. The model is encouraged to conform to predefined rules through the incorporation of additional penalty terms in the loss function during the training process [14]. These penalties should guide the model towards desirable behaviour without strictly enforcing compliance. This technique has been effectively implemented in transformer architectures [15, 16].

In contrast, hard constrained machine learning imposes strict adherence to constraints. This can be achieved through various approaches, including introducing a projection layer in the architecture or constrained beam search [13, 17]. In the latter scenario, the inference process is modified to ensure that only outputs conforming to the specified constraints are considered.

In constrained beam search, the beam search algorithm is adapted to filter out any outputs that do not satisfy the defined rules, ensuring that the final predictions are optimal in terms of likelihood while being compliant with the constraints. Depending on the constraints, the beam search algorithm is adapted in different ways. Various studies have already incorporated constrained beam search in their models and a constrained beam search feature was implemented in HuggingFace's transformers library [17–21]. This feature allows users to specify a sequence of words that should be part of the model's output. It is achieved by forcing the token that is needed to obtain the specified sequence in the output at each iteration of the beam search while also extending the sequence with the most probable tokens as in original beam search.

Another way of enforcing constraints in the output of a machine learning model is by formulating Karush-Kuhn-Tucker (KKT) conditions. When independent constraints between input $\mathbf{x}$ and output $\mathbf{y}$ can be expressed in the form of $\mathbf{Ax} + \mathbf{By} = \mathbf{b}$, the predicted output $\hat{\mathbf{y}}$ can be projected orthogonally on the constrained solution space in case it does not satisfy the constraints. This method was used to model hard linear constraints to enforce mass balances in chemical reactor simulations [13]. The process of projecting on the constrained solution space takes the ground truth data into account in the training phase, which influences the model's learning process. In this way the constraints can be learned by the model. However, this method cannot be directly applied to constrain the atom-balance in reaction completion tasks, as these tasks require not only counting the missing atoms but also determining their appropriate placement within the string representations of molecules.

Another implementation of constrained beam search was used in the Molecular Transformer to ensure that only the atom types of the reactants can be predicted as part of products [2]. In this approach a mask is created that prevents predicting tokens representing atoms that are not part of the reactants. As each token in the vocabulary is associated with at most one atom type, except for hydrogen, which is not regarded in the constrained beam search, the tokens in vocabulary can be mapped to it corresponding atom type. Multiple tokens can have the same atom type, for example, `C`, `c`, `[C@H]` and `[C@@H]` all represent carbon atoms. The mask is a matrix with dimensions of *beam size* $\times$ *batch size* $\times$ *vocabulary size*, where the tokens representing atom types that are present the input molecules are assigned a value of 1 and other atom types are assigned a very small value (1e-15). Tokens that do not represent atoms, but indicate structural features, such as bonds, or molecule separation also have entry 1 in the mask. This mask is multiplied with the log probability matrix during each iteration during beam search, ensuring only valid tokens can be predicted. While this method guarantees that no new atom types appear in the product of the reaction, it

does not consider the number of atoms, so the resulting reaction can still be imbalanced. To address this limitation, we adopt a similar constrained beam search approach, but with a dynamically changing mask that tracks the atom-balance, ensuring that the model's outputs result in balanced reactions.

D. Contributions in this work

In this work we further adapt the Reaction Balancer to enhance its performance in completing partial reactions, focusing on the chemical reactions of the USPTO dataset, as well as biochemical reactions from different datasets. We also investigate the performance on forward prediction and retrosynthesis tasks, which distinguishes the abilities of machine learning approaches from the rule-base completion approaches. We retrain the model using a more diverse dataset of balanced ground truth reactions, and we introduce a loss function that considers atom-balance alongside constrained beam search to improve the chemical validity of the model. We evaluate the performance of the constrained models against the baseline models on both chemical and biochemical datasets.

In this study, we aim to address the following research question and its associated subquestions:

To what extent can soft- and hard atom-balance constraints improve the performance of transformer-based models for completion of partial (bio)chemical reactions?

1. How can atom-balance-based constraints be effectively implemented into a transformer model for reaction balancing?
2. What characteristics should the reaction data have to be suitable for training and benchmarking the proposed model?
3. How do selected constrained machine learning approaches compare in their performance across different classes of reactions and across chemical and biological datasets?

2. METHODOLOGY

A. Data

The chemical reaction data that is used in this work is stored as strings in the format of reaction SMILES. SMILES (Simplified Molecular Input Line Entry System) describes the structure of a molecule using short ASCII strings [22]. Molecules are described by the standard abbreviations of their atoms, where hydrogen is omitted because it can be deduced from the description of the molecule, with additional descriptors for bonds, rings, aromaticity, branching, stereochemistry and isotopes. In reaction SMILES the representations of molecules at each side of the reaction are separated by a period and the two sides of the reaction are separated by a double arrow (»). Solvents, catalysts and other reagents can be noted between the two arrows. Figure 2 illustrates how the molecules of a reaction correspond to their SMILES representation. Atom mappings that indicate which atoms correspond to each other on both sides of the reaction can be added by inserting a colon and a numeric label behind the atoms. SMILES representations of molecules are not unique in the sense that the same molecule can be represented by multiple strings. To obtain unique representations, SMILES strings can be canonicalized, which is important to be able to compare molecules and for consistency when they are used as input for computational tools [23].

Chemical reaction data from USPTO

The chemical reaction data that we use is from the USPTO database [1]. This openly available online database contains 1.8M text-mined reactions from United States patents between 1976 and 2016. Various subsets of the USPTO data have been created for different research projects, the subsets that are relevant for this work are schematically shown in Figure S1. USPTO STEREO contains unique reactions with one product, of which 96.5% are imbalanced [24]. This subset was used to train the Molecular Transformer and the Original Reaction Balancer [2, 10]. The USPTO 50k datasets are subsets with 50,000 randomly selected reactions that could be classified by NameRxn and atom mapped by NameRxn [25–27]. The distribution of the reaction classes in the datasets is not uniform as is shown in Figure S2

B. Performance of existing methods

To evaluate the performance of existing methods across different reaction classes, we apply the Reaction Balancer, its rule-based curation algorithm, and SynRBL to the USPTO 50k subsets illustrated in Figure S1. We exclude already balanced reactions from the datasets, resulting in 49,230 reactions for the 2015 dataset and 48,809 for the 2016 dataset [25, 26]. In the absence of ground-truth reactions, we consider reactions to be curated when they were atom-balanced by the applied methods. From the subset of reactions solved by the MCS-based approach of SynRBL, we

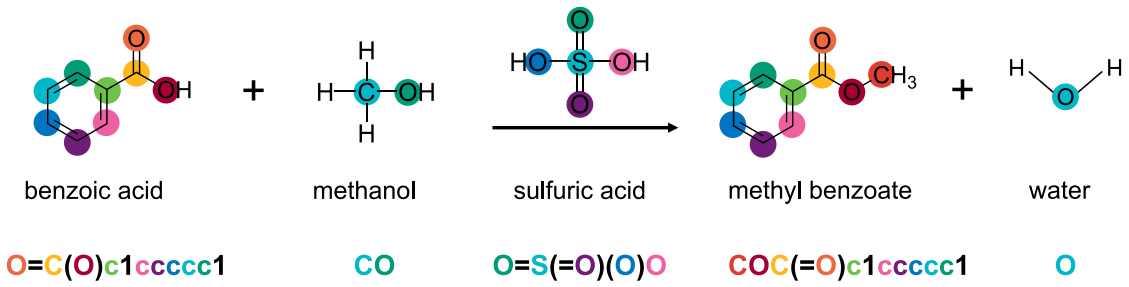


Fig. 2. Example of a chemical reaction with the structure, name and SMILES representation of each molecule. The coloured atoms in the structures correspond to the coloured symbols in the SMILES representation of the molecules. In the reaction SMILES the molecules are separated by periods and the set of reactants, reagents and products are separated by arrows.

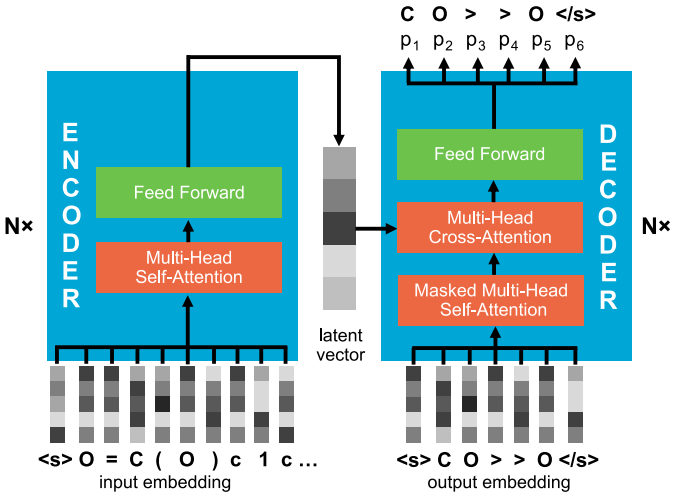


Fig. 3. Overview of the training setup for the transformer architecture used in this study. This simplified representation omits certain details, such as positional encoding and the add-and-norm components. These details are available in the original transformer paper [28].

retain only those curated reactions that achieved a confidence score of at least 90%.

C. Architecture

For our models, we use the same transformer architecture as employed by the Molecular Transformer and Reaction Balancer. A schematic overview of this autoregressive encoder-decoder model is shown in Figure 3. We adopt the same hyperparameters as those used in the previous studies. The implementation is based on the OpenNMT version 0.4.1, with minor modifications to ensure compatibility with PyTorch version 1.13.1 [29, 30].

D. Training data preprocessing

We train three different models. To avoid data leakage from the reactions that were already seen during training of the Molecular Transformer or the Original Reaction Balancer, we retrain the models from scratch. We train one model on a large chemical dataset and one model on a smaller biochemical dataset. Because the biochemical dataset is smaller, as a third model, we fine-tune the model trained on chemical reactions on the smaller biochemical dataset. These models are summarized in Table 2 and the preprocessing of the data is described below.

Chemical Reaction Balancer

Before training the model, several preprocessing steps are applied to the raw USPTO dataset to ensure data quality and suitability [31]. These steps are summarized graphically in Figure S4.

The first step involves standardizing the data, during which all molecules are canonicalized using RDKit [32]. Part of the reactants are atom-mapped to their corresponding atoms in the products. Similar to the preprocessing conducted for USPTO STEREO, we move reactants that have no atom-mapping to the reagents [24]. Subsequently we discard any recorded atom-mappings in the data. Next, we group reactions that share the same set of reactants and products, while separately recording all sets of reagents. We additionally discard 640 reactions that RDKit could not canonicalize, resulting in 1.1 million unique standardized reactions.

From these standardized reactions, we retain only those that have no overlapping patent numbers with either of the USPTO 50k datasets, as we use the USPTO 50k data to evaluate our models. This conservative approach ensures that the model has not encountered reactions similar to those in the USPTO 50k dataset. The importance of considering dataset structure when creating data splits was recently highlighted by a study, which found that using random splits in chemical reaction prediction models leads to overoptimistic performance estimates [33]. The set of standardized reactions includes many imbalanced reactions. To curate these reactions for training, we employ the rule-based algorithmic approach SynRBL that successfully curated 460,563 unique reactions (with a confidence level of $\geq$90% for its MCS-based approach) [8]. We discard reactions where the reactants are identical to the products, resulting in 453,597 balanced reactions, which are split in training (90%) and validation (10%) sets.

These balanced reactions are partialized into training instances, following a method similar to that of the Original Reaction Balancer [10]. For each balanced reaction, a maximum of 10 partialized reactions is created, including the balanced reaction itself without any removed molecules, allowing the model to learn to recognize balanced reactions. In the remaining partialized reactions, molecules are removed such that they accounted for no more than 50% of the total atoms in the reaction. After partialization we obtain 3,726,463 training and 414,079 validation reactions.

To teach the model to recognize reagents, we include reagents as reactants in some of the partialized reactions. The model is designed to predict these reagents on both sides of the reaction. If recorded reagents are present, we assign a 50% chance of including them in each partialized reaction. When reagents are added, one set of reagents is randomly selected, with each reagent being included with a probability of one divided by the number of reagents in that set. This means that if a reagent is the only one in a set, it is always added; if there are two or more, each has a lower probability of being added, which can lead to no, one, or multiple reagents being included in the partialized reaction. To prevent bias in the order of molecules within a reaction, the molecules on each side are shuffled.

Biochemical Reaction Balancer

The Biochemical Reaction Balancer is trained using reaction data from the Rhea database [34]. This is an open source expert-curated knowledgebase containing 17,098 enzymatic and transport reactions in SMILES format. Several preprocessing steps are applied to this data, as illustrated in Figure S5. First, we standardize the reactions and filter out those that are imbalanced or belong to reaction classes containing asterisks that indicate unspecified side groups in a molecule. Upon inspecting the imbalanced reactions, we identified a subset that could be balanced by altering the notation of `S`, which RDKit recognizes as hydrogen sulfide (H_2S), to `[S]`, which denotes sulfur (S). Next, we discard transport reactions where the reactants were identical to the products. This preprocessing results in a dataset of 12,091 standardized balanced reactions. These reactions are subsequently divided into training (72%), validation (8%), and test (20%) datasets and partialized using the same approach as described for the chemical reaction data above. After partialization we obtain 78,240 training and 8,743 validation reactions.

Fine-tuned Biochemical Reaction Balancer

To train the Fine-tuned Biochemical Reaction Balancer, we use the trained Chemical Reaction Balancer model. We fine-tune this model with the reactions from the Rhea dataset that we also use for training the Biochemical Reaction Balancer. To manage GPU memory limitations, we reduce the dataset to include only reactions with a maximum of 250 tokens in both the input and target sequences. This adjustment decreases the number of training reactions from 78,240 to 69,991 and the number of validation reactions from 8,743 to 7,795.

Model variations

For the Chemical and Biochemical Reaction Balancer setup, we train three models: a baseline model and a two constrained models, in which we adapt the loss function with a different weight, as will be explained in Section 2E. For the fine-tuned Biochemical Reaction Balancer we only train a baseline model. Additionally, we introduce hard constraints during inference in our experiments, outlined in Section 2F. The Chemical Reaction Balancer model was trained for 500,000 steps which took approximately 36h for the baseline and 43h for both constrained models on a single GPU and the Biochemical Reaction Balancer model was trained for 250,000 steps which took approximately 15h for the baseline and 21h and 11h for the two constrained models on a single GPU. The Fine-tuned Biochemical Reaction Balancer was fine-tuned from the Chemical Reaction Balancer for 250,000 steps with Rhea data.

E. Soft constraints on the loss function

The cross-entropy loss described in Section 1B only considers the ground truth token and does not differentiate between incorrectly predicted tokens. Therefore, we extend the loss function to incorporate an additional term that penalizes violations of the atom-balance. This modification provides the model with specific feedback regarding atom-balance, guiding it toward learning how to balance reactions. This additional loss term reflects the divergence between the probability distributions of the atom types, arrow and end-of-sequence token based on context ($p_{context}$) and the actual predictions (p_{pred}).

First, $p_{context}$, as shown in Equation 2 and 4, can be pre-computed based on the input and the left-hand context. For each atom type, the number of atoms appearing in the products is subtracted from the number occurring as reactants. The same process is applied to the left-hand context of the predicted token, which includes the tokens preceding the current prediction that are visible (i.e., not masked) during the prediction process. This left-hand context always consists of the non-masked part of the ground truth due to the teacher-forcing mechanism inherent in the transformer decoder, which makes it possible to pre-compute $p_{context}$. The resulting values for the atom-balance of the input and left-hand context are added and in case the prediction is on the reactant side and the atom type is not missing on the reactant side, so the outcome of this sum is greater or equal to zero, the probability value is set to zero. In all other cases the probability values for each atom type are normalized by dividing them by the total sum of the distribution, resulting in a probability distribution that sums to 1. The probability for the arrow token is always set to zero; if the model can predict the arrow token without violating the atom-balance, we do not compute the atom-balance loss and instead return a value of zero. The probability for the end-of-sequence token is set to 1 only when the prediction is on the product side and all atom types are balanced.

Second, p_{pred}, shown in Equation 3 and 5, is computed based on the predicted probabilities over the vocabulary. These predicted probabilities are grouped by atom type and also include the probabilities for the arrow and end-of-sequence tokens. Similar to $p_{context}$, in case the prediction is on the reactant side and

$$p^{\text{context}}_{\text{react}}(i) = \begin{cases} \frac{|i_{\text{src,react}} - i_{\text{src,prod}} + i_{\text{left-hand,react}} - i_{\text{left-hand,prod}}|}{||p^{\text{context}}||} & \text{if } i \in \text{atoms and } i_{\text{src,react}} - i_{\text{src,prod}} + i_{\text{left-hand,react}} - i_{\text{left-hand,prod}} < 0 \\ 0 & \text{if } i \in \text{atoms and } i_{\text{src,react}} - i_{\text{src,prod}} + i_{\text{left-hand,react}} - i_{\text{left-hand,prod}} \geq 0 \\ 0 & \text{if } i = \texttt{>} \text{ or } i = \texttt{<\textbackslash s>} \end{cases} \tag{2}$$

$$p^{\text{pred}}_{\text{react}}(i) = \begin{cases} \frac{\sum_{t \in \text{tokens of i}} p(t)}{||p^{\text{pred}}||} & \text{if } i \in \text{atoms and } i_{\text{src,react}} - i_{\text{src,prod}} + i_{\text{left-hand,react}} - i_{\text{left-hand,prod}} < 0 \\ 0 & \text{if } i \in \text{atoms and } i_{\text{src,react}} - i_{\text{src,prod}} + i_{\text{left-hand,react}} - i_{\text{left-hand,prod}} \geq 0 \\ \frac{p(i)}{||p^{\text{pred}}||} & \text{if } i = \texttt{>} \text{ or } \texttt{<\textbackslash s>} \end{cases} \tag{3}$$

$$p^{\text{context}}_{\text{prod}}(i) = \begin{cases} \frac{|i_{\text{src,react}} - i_{\text{src,prod}} + i_{\text{left-hand,react}} - i_{\text{left-hand,prod}}|}{||p^{\text{context}}||} & \text{if } i \in \text{atoms} \\ 0 & \text{if } i = \texttt{>} \\ 1 & \text{if } i = \texttt{<\textbackslash s>} \text{ and } \forall x \in \text{atoms s.t. } p_{context}(x) = 0 \\ 0 & \text{if } i = \texttt{<\textbackslash s>} \text{ and } \exists x \in \text{atoms s.t. } p_{context}(x) > 0 \end{cases} \tag{4}$$

$$p^{\text{pred}}_{\text{prod}}(i) = \begin{cases} \frac{\sum_{t \in \text{tokens of i}} p(t)}{||p^{\text{pred}}||} & \text{if } i \in \text{atoms} \\ \frac{p(i)}{||p^{\text{pred}}||} & \text{if } i = \texttt{>} \text{ or } \texttt{<\textbackslash s>} \end{cases} \tag{5}$$

$$\mathcal{L} = \begin{cases} \mathcal{L}_{\text{Cross-Entropy}} + \lambda \sum_i |p^{\text{pred}}(i) - p^{\text{context}}(i)| & \text{if } \exists x \in \text{atoms s.t. } p_{context}(x) > 0 \text{ or if } p_{context}(\texttt{<\textbackslash s>}) = 1 \\ \mathcal{L}_{\text{Cross-Entropy}} & \text{else} \end{cases} \tag{6}$$

the atom type is not missing on the reactant side the value is set to zero and in all other cases p_{pred} is normalized by dividing all values by the total size of the distribution.

The computation of the loss is summarized in Equation 6. The original cross-entropy loss is always calculated, while the additional term is computed for the reactant side only when there are missing atoms, and for the product side, it is always included. This additional loss term represents the absolute difference between the two probability distributions, yielding a value between 0 (when the distributions are identical) and 2 (when the non-zero values have no overlap). To balance the two loss terms, the additional loss term is multiplied by a factor, denoted as λ, which is a hyperparameter that must be set manually.

F. Constrained beam search

As explained in Section 1C, during beam search, a mask can be used to restrict certain tokens from being predicted. In the Molecular Transformer, a constrained beam search option was implemented for preventing atom types that are not part of the reactants from being predicted as part of the product. In this work we adapt constrained beam search to enforce the output to be balanced. We restrict predicting tokens that violate the validity of the atom-balance during beam search. For each beam a mask is constructed that restricts the selected tokens. Because the transformer makes predictions in an auto-regressive fashion, it first predicts molecules on the reactant side and subsequently on the product side. We restrict predicting the arrow-token marking the transition between these sides until all missing atoms on the reactant side are predicted based on the products of the input reaction. Note that the decoder can add more reactants before the arrow tokens are predicted; we do not enforce predicting the arrow token. On the product side, we restrict predicting the end-of-sequence tokens until the reaction is balanced and we restrict predicting balanced atom types. Because the atom-balance can change for each additional predicted token, the mask is updated during each iteration.

In contrast to the mask that restricts atom types in the output of the Molecular Transformer, the atom-balance enforcing mask needs to be dynamic because it is based on counts. To initialize the mask, the atoms of the imbalanced input reaction are counted and if there are missing atoms on the left-hand (reactant) side of the reaction, the arrow token (>) cannot be predicted. This mask is updated for every token that is predicted, to check whether there are still missing atoms on the reactant side and if there are no missing atoms, the arrow token will be stopped being constrained. Once the arrow tokens are predicted, the total number of atoms and charge on the reactant side are determined to calculate the exact number of atoms per atom type that are still needed on the right-hand (product) side to balance the reaction. Until the reaction is both atom and charge balanced, the end-of-sequence token (`</s>`) is restricted. All tokens representing atom types that are already balanced are constrained as well. Because the atom-balance can change for each additional predicted token, the mask is updated during each iteration.

For tokens that are not in the training vocabulary of the model, the atom type cannot be identified. If out-of-vocabulary tokens appear in an input sequence, they are represented by the unknown (`<unk>`) token. Because it is not possible to derive the atom-balance from a sequence with unknown tokens, constrained beam search is disabled for reactions with out-of-vocabulary tokens. Additionally, the model can predict unknown tokens as part of its output, which are subsequently replaced by the input token with the highest input weight. This can lead to violations of the atom-balance, because out-of-vocabulary tokens representing atoms are not identified when constructing the atom-balance. Therefore, we always constrain the unknown token in constrained beam search.

Constrained beam search ensures that the model's output is atom-balanced, with the exception of hydrogen, causing it to be a hard constraint. However, it does not guarantee that the predicted outcomes align with the ground truth or consist of valid molecules. Additionally, by restricting tokens that might otherwise have a high probability, the use of constrained beam search can result in predicting sequences with lower probability outcomes.

G. Evaluation

We benchmark our models using datasets for which ground truth data is available. Additionally, we assess the models' ability to balance reactions without ground truth data. We also compare the models across different reaction classes. A summary of the datasets used for training and evaluating the models is presented in Table 2.

For benchmarking the Chemical Reaction Balancer models,

Model	Training data	Benchmark data	Balancing data	Class data
Chemical Reaction Balancer (CRB)	• USPTO excluding overlap with USPTO 50k curated by SynRBL *(454k*)*	• Original and partialized validation set from SynRBL *(5k + 5k*)* • Rhea test set *(2k*†)* • ECReact *(28k*†)*	• Uncurated USPTO excluding overlap with USPTO 50k *(147k†)*	• USPTO 50k *(92k)*
Biochemical Reaction Balancer (BRB)	• Rhea train set *(10k*)*	• Rhea test set *(2k*†)* • ECReact *(28k*†)*	-	-
Fine-tuned Biochemical Reaction Balancer (ft-BRB)	• USPTO excluding overlap with USPTO 50k curated by SynRBL *(454k*)* • Rhea train set *(10k*†)*	• Original and partialized validation set from SynRBL *(5k + 5k*)* • Rhea test set *(2k*†)* • ECReact *(28k*†)*	• Uncurated USPTO excluding overlap with USPTO 50k *(147k†)*	• USPTO 50k *(92k)*

Table 2. Overview of the transformer models developed in this work and the datasets used for their training and evaluation. The number of reactions of the datasets is denoted between brackets for each dataset. The reactions of the datasets indicated with * are partialized before being used as training data. Datasets indicated with † are filtered to only contain reactions of maximally 250 tokens for both input and target sequence (if available).

we use the validation set of SynRBL, which was manually verified by the authors of the study [8]. This validation set, comprising of 5,420 reactions, is sourced from three datasets, including the USPTO 50k dataset of 2016 [26, 35, 36]. We select only the reactions listed in the `expected_reaction` column of their dataset that we confirm to be atom-balanced, resulting in a total of 4,610 reactions. In this original validation set, the completion task involves reconstructing the reaction from the raw data, where both the main reactant and product are always present. We test our models on this original validation set as well as on a partialized version, where we randomly remove molecules that account for no more than 50% of the total atoms in the reaction. This partialized dataset contains 41,530 reactions and includes both forward prediction, retrosynthesis, and completion tasks with a combination of missing molecules.

For benchmarking the Biochemical Reaction Balancer models, we use a held-out test set comprising 20% of the Rhea data [34]. The reactions in this test set are partialized in the same manner as the training data, yielding 21,897 partialized reactions. For benchmarking on an out-of-distribution dataset, we use the biochemical reactions from ECReact, where we exclude any overlap with Rhea and imbalanced reactions [37]. The resulting dataset contains 28,146 unique reactions that are partialized into 249,257 reactions.

To evaluate the models' ability to balance reactions from an external dataset without ground truth, we use the uncurated portion of the USPTO data, which comprises 147,558 reactions. To compare performance across different reaction classes, we take the 92,205 unique reactions with reaction class labels from the USPTO 50k datasets of 2015 and 2016. During standardization of the 2016 dataset we use the reported reactant set identified by NameRxn to discard the reagents from the reactant side. For the 2015 dataset the set of reactants is not explicitly recorded, so we keep all reactants. We discard five unique reactions for which the standardized reactions belong to multiple reaction classes.

To keep the inference runtime feasible, the reactions of the Rhea, ECReact and uncurated USPTO evaluation sets are filtered to only contain reactions of maximally 250 tokens for both input and target sequence (if available). After filtering the Rhea test dataset contains 19,806 reactions, the ECReact dataset contains 204,222 reactions and the uncurated USPTO dataset contains 146,573 reactions.

Round-trip accuracy

To estimate accuracy on the uncurated USPTO reactions, we employ a modified version of round-trip accuracy, a metric that was introduced in the Original Reaction Balancer work [10]. In this approach, the predicted output molecules are used as input for a new prediction task. If the result of this reverse prediction matches the initially predicted reaction, it is defined to be round-trip accurate. For completion tasks with only a small set of missing molecules, the output set may be too small to form a partial reaction. To address this, we add additional molecules to meet a predefined round-trip minimum percentage of atoms that should be present in the modified partial reaction. We set a round-trip minimum of 50% of the atoms for the modified reactions to ensure that these reactions cannot be the full balanced reactions. We observe that when the round-trip minimum exceeds 50%, for a subset of the partialized reactions no molecules can be removed, meaning the model does not need to predict any missing molecules and the accuracy can appear overoptimistic.

3. RESULTS AND DISCUSSION

A. Performance of existing methods

The results of experiments on existing methods demonstrate that the algorithmic approach, SynRBL, outperforms solely rule-based methods across reaction classes. As illustrated in Figures S6A and B, SynRBL consistently shows higher performance and a more balanced distribution across reaction classes, with the only exception being the reactions from class 7 (reductions) in the 2015 dataset. Most of the reactions of this class that are solved by the MCS-based method have a confidence score below 90%. Additionally there is a subset consisting of 4.8% of the reactions of this dataset that only missed molecular oxygen (O_2) or hydrogen (H_2) that were not solved by SynRBL. This also partly explains the superior balancing performance of the curation algorithm in this class since both molecular oxygen and hydrogen are part of its set of helper species. In general, the reactions solved by SynRBL using only the rule-based component show a distribution comparable to those balanced by the curation algorithm, as can be seen in Figures S6C and D. This similarity in performance can also be observed for the Original Reaction Balancer, which is trained on reactions curated by the curation algorithm, resulting in comparable outcomes. These findings suggest that the performance of a machine learning model can be improved by training it on a more diverse set of ground truth reactions. As detailed in Section 2, we curate USPTO reactions, excluding those present in either USPTO 50k dataset, using SynRBL. From the reactions identified through the MCS-based approach, we retain only those with a confidence score exceeding 90%, which constituted the majority of the dataset, as shown in the confidence distribution in Figure S7.

B. Performance of the trained models

An overview of the evaluation results is presented in Table 3. This table shows the performance of our trained models and SynRBL on the evaluation data stated in Table 2, which includes the percentages of balanced reactions (both fully balanced (Bal.) and balanced disregarding hydrogen (Bal. -H)), the accuracy (Acc.) in case ground truth data is available and the percentage of reactions that contain an unknown token, leading to constrained beam search being disabled, (Unk.) for the experiments with constrained beam search. For the validation set of SynRBL and for the USPTO 50k data we report the SynRBL accuracy (SynAcc.), which is the percentage of reactions that exactly match the prediction of SynRBL in case SynRBL predicts a balanced reaction with a confidence of at least 90% for its MCS-based approach. Additionally, the training performance of the models on both the Rhea and USPTO validation sets of the training data is presented in Table 4. The reported token accuracy and loss values during training are presented in Figures S8 and S9. The remainder of this section will discuss these results and present additional analysis to interpret our findings.

Remarks on the pre-processing steps and experiments

During the pre-processing steps and throughout our experiments, we made several noteworthy observations. Firstly, the USPTO dataset includes a few reagents that could not be parsed by RDKit and were discarded, these are: `[NH4+]=S`, `C=Cl`, and `C1C=CC([Sn-](F)(F)(C2C=CC=CC=2)C2C =CC=CC=2)=CC=1`. Secondly, we notice that in the pre-processing step of SynRBL, brackets around uncharged atoms are removed, along with any recorded hydrogen atoms within those brackets. We observe that this alteration can sometimes change the identity of the

Data		BL CRB	BL CRB CBS	C (λ=1) CRB	C (λ=1) CRB CBS	C (λ=0.1) CRB	C (λ=0.1) CRB CBS	BL BRB	BL BRB CBS	C (λ=1) BRB	C (λ=1) BRB CBS	C (λ=0.1) BRB	C (λ=0.1) BRB CBS	BL ft-BRB	BL ft-BRB CBS	SynRBL
Original validation	Bal.	90.24%	92.58%	87.61%	91.56%	90.41%	92.84%	-	-	-	-	-	-	47.38%	70.09%	**99.57%**
	Bal. -H	91.34%	95.42%	89.02%	94.38%	91.45%	94.99%	-	-	-	-	-	-	51.28%	82.52%	**99.57%**
	Acc.	80.02%	80.72%	78.76%	81.61%	80.43%	81.15%	-	-	-	-	-	-	33.49%	44.01%	**82.60%**
	SynAcc.	85.94%	86.96%	84.75%	87.48%	86.29%	87.27%	-	-	-	-	-	-	33.32%	44.66%	97.03%
	Unk.	-	0.13%	-	0.13%	-	0.13%	-	-	-	-	-	-	-	0.13%	-
Partialized validation	Bal.	88.48%	92.65%	78.30%	89.04%	88.68%	**92.73%**	-	-	-	-	-	-	33.55%	56.49%	39.40%
	Bal. -H	90.68%	**96.24%**	80.00%	92.53%	90.98%	96.19%	-	-	-	-	-	-	36.76%	70.72%	39.75%
	Acc.	73.62%	75.11%	63.30%	69.56%	73.81%	**75.40%**	-	-	-	-	-	-	22.66%	30.65%	25.43%
	Unk.	-	3.82%	-	3.82%	-	3.82%	-	-	-	-	-	-	-	3.82%	-
Uncurated USPTO	Bal.	35.98%	**56.25%**	30.82%	48.13%	34.93%	54.54%	-	-	-	-	-	-	10.10%	34.97%	28.73%
	Bal. -H	38.77%	**74.33%**	34.35%	67.62%	38.31%	72.53%	-	-	-	-	-	-	12.15%	56.81%	28.73%
	Unk.	-	0.14%	-	0.14%	-	0.14%	-	-	-	-	-	-	-	0.14%	-
USPTO 50k	Bal.	66.68%	**72.54%**	65.05%	67.90%	66.78%	72.04%	-	-	-	-	-	-	45.18%	61.37%	**96.29%**
	Bal. -H	67.40%	76.43%	65.64%	72.24%	67.32%	75.18%	-	-	-	-	-	-	47.12%	71.27%	**96.34%**
	SynAcc.	63.68%	64.67%	62.82%	64.41%	63.72%	64.78%	-	-	-	-	-	-	33.63%	39.32%	83.98%
	Unk.	-	0.02%	-	0.02%	-	0.02%	-	-	-	-	-	-	-	0.02%	-
Rhea	Bal.	38.76%	64.88%	29.03%	47.76%	36.79%	63.85%	52.37%	74.58%	31.82%	55.78%	52.85%	73.74%	83.29%	**92.69%**	41.50%
	Bal -H	43.93%	81.72%	32.85%	62.58%	40.55%	80.05%	54.50%	84.44%	33.08%	62.96%	54.92%	82.41%	84.52%	**96.38%**	41.52%
	Acc.	19.44%	23.42%	15.99%	19.06%	18.13%	23.10%	39.30%	52.91%	25.11%	39.00%	39.99%	53.00%	65.69%	**70.66%**	10.92%
	Unk.	-	0.88%	-	0.88%	-	0.88%	-	0.11%	-	0.11%	-	0.11%	-	0.88%	-
ECReact	Bal.	39.32%	**64.71%**	28.67%	48.19%	37.12%	64.13%	22.51%	39.44%	15.28%	29.40%	22.36%	37.44%	46.51%	63.66%	31.88%
	Bal. -H	43.56%	**80.84%**	32.33%	62.97%	41.18%	80.39%	24.79%	51.67%	16.90%	39.32%	24.65%	48.38%	48.00%	69.51%	31.88%
	Acc.	16.59%	20.05%	13.95%	16.56%	15.65%	20.24%	14.04%	20.79%	10.82%	16.18%	14.08%	20.48%	26.88%	**31.90%**	15.00%
	Unk.	-	2.49%	-	2.49%	-	2.49%	-	0.86%	-	0.86%	-	0.86%	-	2.49%	-

Table 3. Benchmarking top-1 results of the baseline (BL) and constrained (C) Chemical (CRB) Biochemical (BRB) and fine-tuned Biochemical (ft-BRB) Reaction Balancer with normal and constrained beam search (CBS) with a beam-size of 5 on the original validation set of SynRBL (4610), partialized validation set of SynRBL (41530), uncurated part of USPTO (146573), both USPTO 50k datasets (92205), partialized reactions of the Rhea test set (19806) and partialized reactions of the ECReact test set (204222). Results are divided into the percentage of balanced reactions (bal.), balanced reactions when hydrogen is ignored (bal. -H) and, if available, the percentage of reactions that exactly matched the set of ground truth reactants and products (acc.). For the original validation set of SynRBL and for the USPTO 50k data we report the SynRBL accuracy (SynAcc.), which is the percentage of reactions that exactly match the prediction of SynRBL in case SynRBL predicts a balanced reaction with a confidence of at least 90% for its MCS-based approach. Note that the SynRBL accuracy of the machine learning models cannot be higher than the SynRBL accuracy achieved by SynRBL itself. For experiments with constrained beam search the percentage of reactions that contains the `<unk>` token for which constrained beam search was disabled is reported (unk.).

molecule; for instance, [Br] is converted to Br, transforming the molecule from bromine (*Br*) to hydrogen bromide (*HBr*). Furthermore, SynRBL identifies a small subset of imbalanced reactions as solved—specifically, 30 out of 608,121 processed reactions. These imbalanced reactions involve carbon, sulfur, potassium and hydrogen imbalances. We also notice a peculiarity in the validation set of SynRBL.

In the reactions where the authors of SynRBL manually added molecular hydrogen to balance a reaction of their validation set, they used the SMILES notation [HH] instead of the conventional [H][H]. This notation remains unchanged during the canonicalization process, meaning it appears in both the input and ground truth data of the validation test sets. Since [HH] is never used in the training data, it is not part of the models' vocabulary and represented by the unknown token during translation. Unfortunately, this deviating notation was identified too late to be addressed in our experiments. It affects one input reaction and 308 target sequences (6.68% of the dataset) in the original validation set and 1,557 partialized input reactions (3.74%) and 1,247 partialized target sequences (3.0%) in the partialized validation set. Lastly, the process of partialization reveals several reactions that can be decomposed into two balanced reactions, which occur as balanced yet incomplete reactions within the partialized data. Specifically, we identify two reactions in the partialized validation set, five reactions in the Rhea test set, and 29 reactions in the ECReact data of which both the input and the output are non-empty and balanced.

C. Constrained loss function

Our results indicate that the constrained loss function does not improve the models ability to balance the reactions. This follows from the reported training performance on the validation sets presented in Table 4. Both the Chemical and Biochemical Reaction Balancer models are trained using the constrained loss function with weights (λ) of 1 and 0.1. When using a λ of 1, the model disproportionately weighs the atom-balance loss, leading to a higher cross-entropy loss and, consequently, lower accuracy. In contrast, when adjusting the weight to 0.1, both loss terms have a similar range of values, allowing the model to reduce the atom-balance loss while maintaining a low cross-entropy loss. However, despite this ability of the model with $\lambda = 0.1$, we do not observe significant improvements in accuracy. The reported accuracy and loss values during training are presented in Figures S8 and S9. From these plots we observe that the constrained models with $\lambda = 1$ have relatively unstable accuracy and cross-entropy loss values and the constrained models with $\lambda = 0.1$ have less stable values for the atom-balance loss.

When comparing the baseline models to their equivalents with the constrained loss function, we find that the constrained models with $\lambda = 1$ perform worse than their corresponding baseline models across all experiments, with the exception of the original validation set of SynRBL. As can be seen in Table 3, for the constrained models with $\lambda = 0.1$, the performance deviates from the baseline models by only about 1%. Overall, these constrained models tend to perform slightly better on datasets that are similar to the training data, such as the validation sets and USPTO 50k for the Chemical Reaction Balancer, as well as Rhea for the Biochemical Reaction Balancer. Conversely, their performance appears to be worse compared to baseline models on out-of-distribution datasets, including Uncurated USPTO and Rhea for the Chemical Reaction Balancer, and ECReact for both models.

Effect of the atom-balance loss on the cross-attention weights

We observe differences in cross-attention weights when examining a specific reaction where the output of the constrained Chemical Reaction Balancer with $\lambda = 0.1$ is imbalanced, while the baseline model produces a balanced output. An example of this is presented in the forward prediction task shown in Figure 4, accompanied by corresponding cross-attention plots in Figure 5. The cross-attention layers, as depicted in Figure 3, allow the model to focus on relevant parts of the input sequence when generating an output. The cross-attention weights represent how much attention the model pays to different parts of the input when making predictions for a specific output token.

While both the baseline and constrained Chemical Reaction Balancer with $\lambda = 1$ generate accurate balanced outputs, their cross-attention plots reveal significant differences. In the baseline model, the cross-attention weights for output tokens are often highest for input tokens or atoms of reactants that map closely to the tokens or atoms of the products, particularly those predicted next. In contrast, the constrained model with $\lambda = 1$ displays more fixed distributions of attention weights throughout the output sequence, with the highest attention weights assigned to a limited set of four input tokens: .,), > and >. Additionally, the constrained model with $\lambda = 0.1$ generated an incorrect output, predominantly focusing on the same fixed set of tokens: ., > and >. This loss of dynamic attention patterns

Data		BL CRB	C (λ=1) CRB	C (λ=0.1) CRB	BL BRB	C (λ=1) BRB	C (λ=0.1) BRB	BL ft-BRB
Rhea	Token Acc.	90.54%	88.10%	90.43%	95.98%	95.69%	96.05%	**97.22%**
	CE-Loss	0.381	0.597	0.392	0.295	0.431	0.305	**0.197**
	AB-Loss	-	-	-	0.867	**0.140**	0.498	-
USPTO	Token Acc.	**98.02%**	96.36%	98.01%	53.17%	53.93%	53.00%	88.91%
	CE-Loss	**0.049**	0.216	0.050	5.610	4.251	5.789	1.092
	AB-Loss	0.634	**0.105**	0.388	-	-	-	0.697

Table 4. Metrics of the trained models on the validation sets of the training data (both validation sets are 10% of the training data). Token accuracy (Token Acc.), cross-entropy loss (CE-Loss) and atom-balance loss (AB-Loss) for the baseline (BL) and constrained (C) Chemical Reaction Balancer (CRB), Biochemical Reaction Balancer (BRB) and fine-tuned Biochemical Reaction Balancer (ft-BRB). Since the vocabulary of the CRB and ft-BRB models does not include all tokens of the Rhea validation set and the vocabulary of the BRB models does not include all tokens of the USPTO validation set the atom-balance loss cannot be calculated, because it cannot interpret the <unk> tokens.

may explain the constrained model's less stable performance.

Potential improvements of the atom-balance loss

It is not straightforward to determine the exact contribution of the constrained loss function to the training of the model, as transformers are often considered black box models. This complexity also makes it challenging to identify potential improvements for the additional loss function. For future investigations, understanding how the backpropagation of the different loss terms affects the cross-attention weights might be helpful. Another area for improvement is better alignment of the objectives of the cross-entropy loss and the atom-balance loss. Currently, the atom-balance loss is calculated based on the difference between two probability functions, which means it reaches a minimal value only when the precomputed and predicted probabilities in $p_{context}$ and p_{pred} are perfectly aligned. As a result, achieving an atom-balance loss of zero often requires distributing probability scores across different tokens, while the cross-entropy loss is minimized only when the full probability is assigned to the ground truth token. A possible solution to this issue is to create a Boolean vector from the $p_{context}$ vector, with entry *True* for non-zero and *False* for zero values, and construct the loss using predicted probability scores that correspond to the *False* entries in the Boolean $p_{context}$ vector.

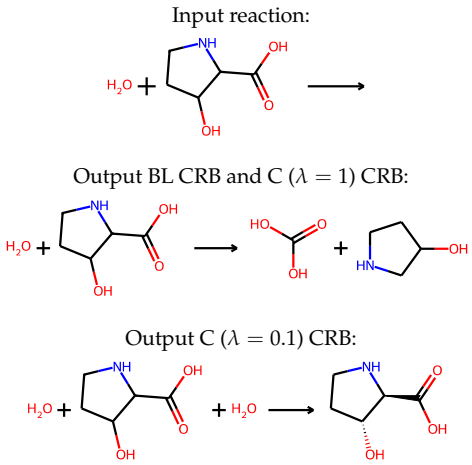


Fig. 4. Input and output reactions from the partialized validation set for which the baseline and constrained ($\lambda = 1$) Chemical Reaction Balancer produced a balanced output and constrained ($\lambda = 0.1$) Chemical Reaction Balancer produced an imbalanced output. The output of the baseline model matches the ground truth.

D. Constrained beam search

The percentage of balanced reactions and accuracy achieved through constrained beam search is consistently at least as high as that of normal beam search. This is because constrained beam search allows for the full generation of balanced reactions, meaning that tokens that do not violate atom-balance are not restricted. As shown in Table 3, using constrained beam search results in a higher number of balanced reactions and slightly improved accuracy in most experiments. Especially the percentage of balanced reactions when ignoring hydrogen is significantly higher in experiments with constrained beam search, because the model does not consider the hydrogen balance. In some types of reactions, such as reductions or oxidations, this can lead

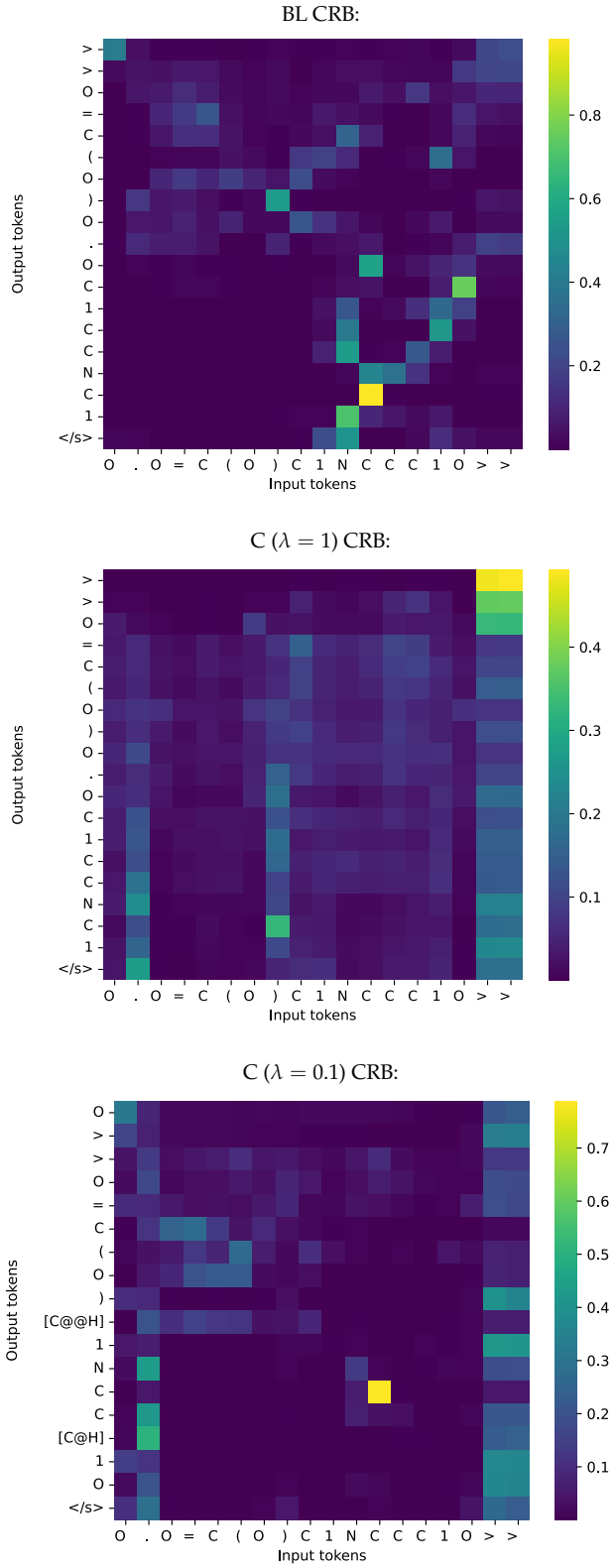


Fig. 5. Cross-attention plots of the baseline (BL) and constrained (C) Chemical Reaction Balancer models for the reaction depicted in Figure 4.

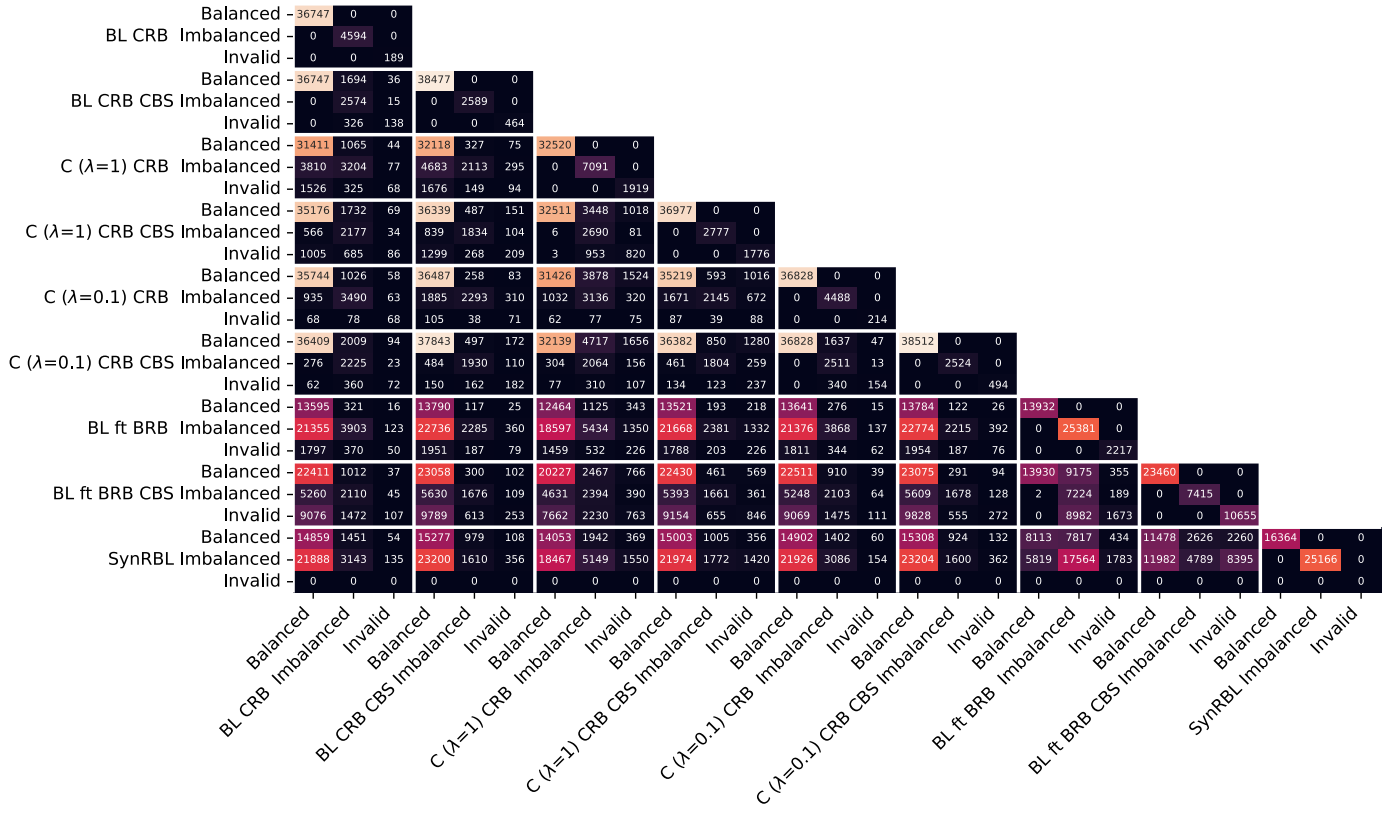


Fig. 6. Overlap in the performance of the Chemical Reaction Balancer and fine-tuned Biochemical Reaction Balancer models with and without constrained beam search and of SynRBL on the partialized validation set categorized into balanced, imbalanced and invalid predictions. Each matrix of 3×3 outlined by white borders contains the overlap between any two models adding up to the total of all 41530 reactions. The matrices on the main diagonal contain the categories within each method. The algorithmic approach SynRBL never produces invalid reactions.

to false outcomes. So, when constrained beam search fails to produce a balanced reaction, we typically observe either reactions with hydrogen imbalanced or invalid reactions, which, for example, reach the maximum sequence length.

In our analysis of the predictions, we occasionally observe violations in atom-balance or reaction validity by models using constrained beam search in reactions that were balanced using normal beam search. As illustrated in Figure 6, we categorize the predicted reactions of the partialized validation set into balanced, imbalanced, and invalid predictions and report the overlap in number of reactions per category between any two models. As expected, both the baseline and constrained Chemical Reaction Balancer with $\lambda = 0.1$ always generate balanced reactions in case the same model without constrained beam search generates a balanced reaction. In contrast, the constrained Chemical Reaction Balancer with $\lambda = 1$ generates six imbalanced and three invalid reactions in cases where this model without constrained beam search predicted balanced reactions. These specific reactions are shown in Figure S11. Notably, only two out of nine reactions resulted in a balanced outcome that matches the ground truth. This scenario can only occur when the generated imbalanced or invalid sequence achieves a higher probability score but remains undiscovered by the normal beam search due to limited beam size. These findings indicate that this model is insufficiently trained to consistently assign higher probability scores to balanced reactions.

Additionally, compared to normal beam search, we observe an increase in invalid reactions with constrained beam search. This happens because the constrained beam search restricts the prediction of imbalanced reactions. In cases where the model cannot predict the missing atoms, it can generate long sequences of nonsensical tokens, as the arrow or end-of-sequence token remains constrained. We hypothesized that the constrained loss function might mitigate the generation of invalid results; however, our findings indicate that constrained models using constrained beam search do not outperform their corresponding baseline models with constrained beam search.

Effect of the beam size on constrained beam search

Increasing the beam size in constrained beam search can be misleading, as it appears to increase the number of balanced reactions without improving accuracy on out-of-distribution data. Figure 7 illustrates the impact of beam size on the performance of the baseline Chemical Reaction Balancer on the biochemical Rhea dataset. While increasing the beam size with normal beam search maintains stable accuracy and percentage of balanced reactions, the percentage of balanced reactions in constrained beam search continues to rise with larger beam sizes, even as accuracy remains relatively stable. This discrepancy is particularly concerning in real-life scenarios where ground truth data is unavailable, as a high percentage of balanced but inaccurate reactions can lead to wrong conclusions.

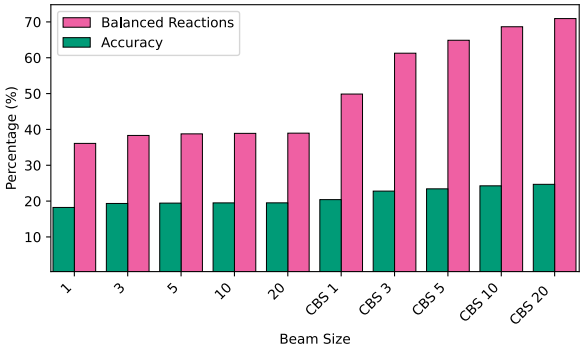


Fig. 7. Performance of the baseline Chemical Reaction Balancer with varying beam size on partialized Rhea test set with and without constrained beam search (CBS).

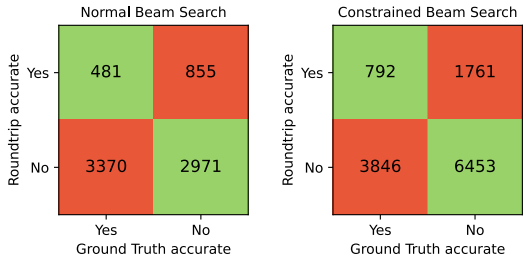


Fig. 8. Confusion matrices of the round-trip accuracy of the baseline Chemical Reaction Balancer on the Rhea test set, with and without constrained beam search. The balanced predictions are again partialized into modified reactions with a round-trip minimum of 50% and used as input for the model. The confusion matrices show the number of predictions that match the initial balanced output prediction as *round-trip accurate* and the number of prediction that match the ground truth reaction as *ground truth accurate*.

Round-trip accuracy proves unreliable for confidence estimation

To evaluate whether we can distinguish between inaccurate and accurate balanced predictions, we determine the round-trip accuracy on the Rhea test set, as described in Section 2G. We generate modified partialized reactions from the balanced predictions of the Chemical Reaction Balancer using normal and constrained beam search. With ground truth data available, we can construct a confusion matrix. If round-trip accuracy were a perfect indicator of the model's accuracy, all reactions would be classified as either both round-trip and ground truth accurate or as neither. However, as shown in Figure 8, the majority of the round-trip accurate predictions do not align with the ground truth. This indicates that using the round-trip accuracy is not informative for the actual accuracy of the predictions.

Probability scores for confidence estimation

To still distinguish inaccurate from accurate balanced predictions using an alternative method, we measure the confidence using the probability score of the model, which can provide a more reliable assessment of prediction quality. Figure 9 shows the distribution of probability scores for predictions that are accurate, balanced, but not accurate, and imbalanced (and not accurate). The plot shows that accurate predictions mostly have probability scores close to one, whereas inaccurate predictions have probability scores closer to zero. This makes the probability score a good estimate of the certainty of a prediction being accurate. The observed distributions for experiments with both normal and constrained beam search are similar. The indicated quartiles show that the probability scores of the inaccurate predictions made by models with constrained beam search are lower than for models with normal beam search. However, there are also inaccurate predictions with high probability scores and accurate predictions with low probability scores that risk being misclassified when only considering this score.

The heatmaps in Figure S10 show the relationship between the length of the prediction versus the probability score for accurate, inaccurate balanced and imbalanced predicted reactions. In general there are relatively few long predictions with high probability scores. For predictions with a score below 0.1 at least half is inaccurate across all length ranges. Balanced but inaccurate predictions appear mostly on the lower triangular matrix, which include short predictions with any probability value, mid-length predictions with low to medium probability values and long predictions with a low probabilities indicating some correlation between the prediction length and probability score within this group of predictions.

Runtime of constrained beam search

During our experiments we notice an increase in runtime when using constrained beam search. While running inference on in-distribution validation sets results in only a slight runtime increase (by a factor of 1-2), the runtime increases more for

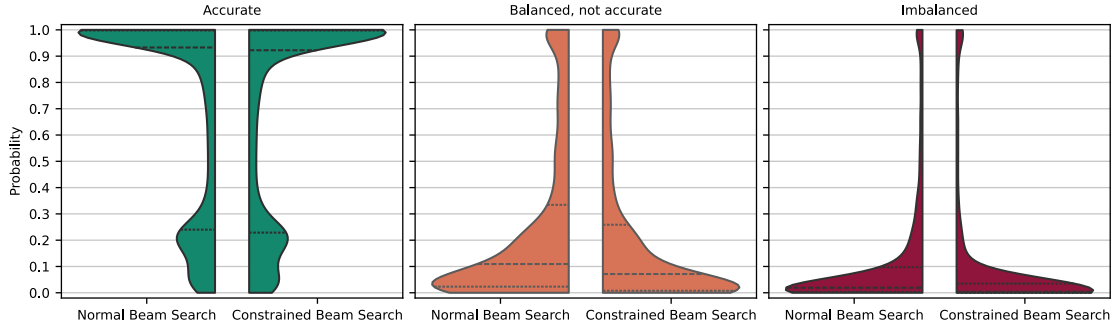


Fig. 9. Probability distributions of the probability scores associated with the predictions made by the baseline Chemical Reaction Balancer with normal and constrained beam search on the partialized validation set of SynRBL for accurate, balanced inaccurate and imbalanced inaccurate predictions. The quartiles of the distributions are indicated by dotted lines.

out-of-distribution datasets, with observed runtimes of up to four times slower. Due to the experiments being conducted on different GPUs within our cluster, precise comparisons are not feasible, and we do not report these runtimes.

E. Algorithmic versus Machine Learning approaches

While the algorithmic completion methods excel at predicting small missing molecules, our transformer models demonstrate superior performance across a variety of completion tasks. Our results, as shown in Table 3, indicate that SynRBL achieves the best performance on the original validation set, whereas our Chemical Reaction Balancer models perform better on the partialized validation set.

Performance on the different types of tasks

Figure 10 shows the performance of the models on the partialized validation set, for which the tasks is divided into different categories. The performance on tasks where only hydrogen is imbalanced is the same for all Chemical Reaction Balancer models, as would be expected since the constraints do not keep track of hydrogen imbalances. SynRBL generates a relatively large number of balanced, yet inaccurate reactions in this category of hydrogen-imbalanced reactions, by predicting molecular hydrogen (`[H][H]`, H_2) where the ground truth contains hydrogen atoms (`[H]`, H). Additionally, part of the inaccurately predicted reactions that are missing molecular hydrogen arise from the deviating notation `[HH]` in part of the target sequences, as mentioned in Section 3B. For other carbon-balanced reactions, SynRBL is not able to accurately solve most of the reactions where at least ten tokens are missing. For carbon-imbalanced reactions the performance of all methods seems relatively similar for a varying number of missing tokens. In cases of partialized reactions with carbon atoms present on only one side of the reaction (C-incomplete), SynRBL fails in nearly all instances due to its inability to construct a maximum common subgraph (MCS). Moreover, SynRBL is not able to solve any forward prediction or retrosynthesis tasks. In contrast, the Reaction Balancer models maintain relatively stable performance across these various tasks.

Synergy between algorithmic and machine learning approaches

The integration of algorithmic and machine learning approaches presents a synergistic potential. By training our models on reactions curated through an algorithmic method and extending the variety of tasks that these models can perform, algorithmic and machine learning approaches enhance each other's functionality. Additionally, both approaches can be used to cross-verify each other's predictions.

To cross-verify the predictions of our models and SynRBL we measure the percentage of reactions that exactly match the prediction of SynRBL in case this prediction is balanced and has a confidence of at least 90% for the MCS-based method. This SynRBL accuracy is reported in Table 3 (SynAcc.) for the original validation set and USPTO 50k. All Chemical Reaction Balancer models achieve a SynRBL accuracy of around 6% higher than their accuracy on the original validation data. All values are relatively similar across these models on both the original validation data and USPTO 50k indicating that the atom-balance constraints do not have a large impact on this value.

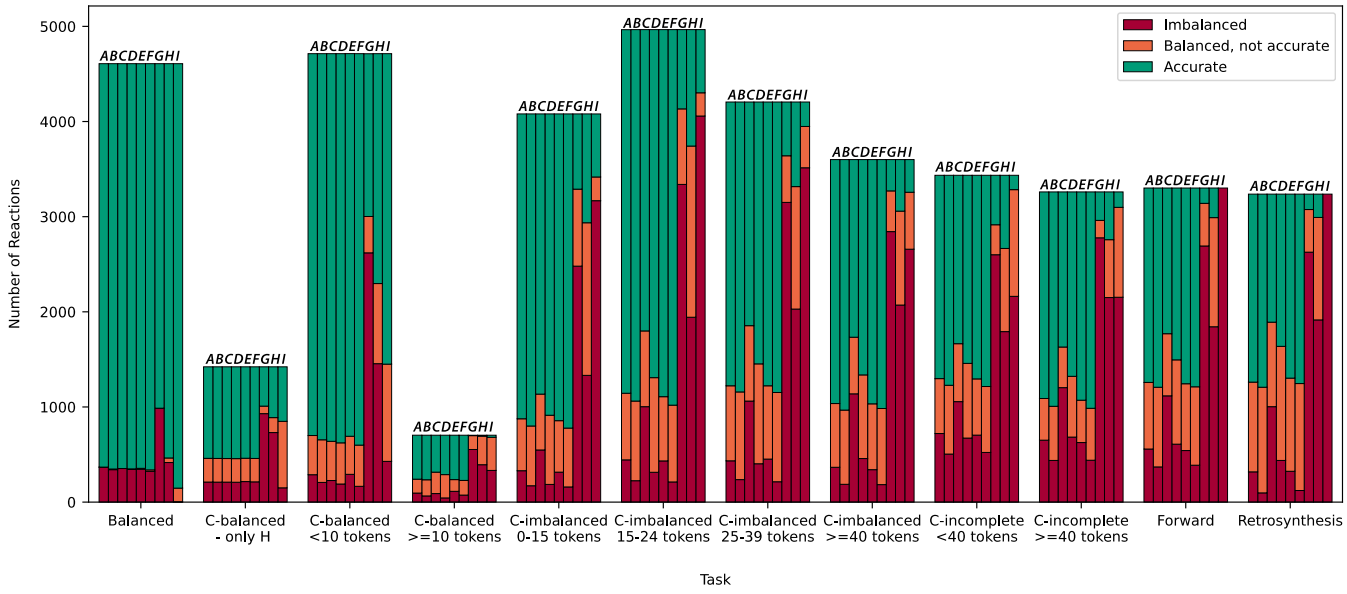


Fig. 10. Performance per task on the partialized validation set of SynRBL of ***A***: Baseline CRB, ***B***: Baseline CRB with CBS, ***C***: Constrained (λ=1) CRB, ***D***: Constrained (λ=1) CRB with CBS, ***E***: Constrained (λ=0.1) CRB, ***F***: Constrained (λ=0.1) CRB with CBS, ***G***: Baseline ft-BRB, ***H***: Baseline ft-BRB with CBS, ***I***: SynRBL. The tasks are divided into balanced reactions (first bar), imbalanced reactions with input molecules on both sides of the reaction (second to tenth bar), reactions with only the reactants as input (forward, eleventh bar) and reactions with only the products as input (retrosynthesis, twelfth bar). The imbalanced reactions are again divided into reactions with carbon balanced (C-balanced), carbon imbalanced, but with carbon atoms on both sides of the input reaction (C-imbalanced) and carbon imbalanced with carbon atoms on only one side of the input reaction (C-incomplete). The C-balanced reactions are subdivided into reactions where only hydrogen atoms are missing and reactions where other atom types are missing of less than ten missing tokens and of ten or more missing tokens. The C-imbalanced reactions are subdivided into reactions that are missing less than 15, 15-24, 25-39 and 40 or more tokens. The C-incomplete reactions are subdivided in reactions that are missing less than 40 and 40 or more tokens.

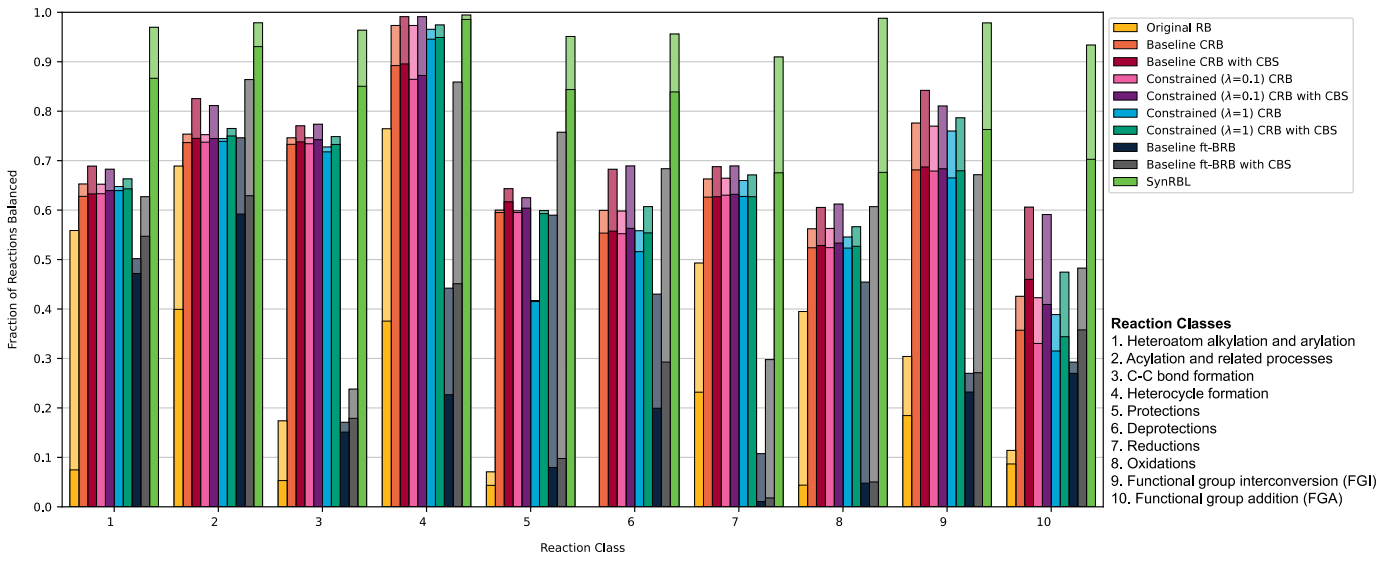


Fig. 11. Balancing performance of the Chemical Reaction Balancer and fine-tuned Biochemical Reaction Balancer models with and without constrained beam search and of the Original Reaction Balancer and SynRBL on USPTO 50k data per reaction class. The full bars show the total fraction of reactions that are balanced by each method within each reaction class. The brighter bars within each bar indicate the SynRBL accuracy, which is the fraction of reactions that exactly match the predictions made by SynRBL with a confidence of at least 90% for the MCS-based method. Note that this measure can never exceed the SynRBL accuracy of SynRBL itself.

Performance of machine learning approaches on reactions uncurated by algorithmic approaches

The uncurated part of the USPTO data is of particular interest, as it contains reactions that could not be solved by the algorithmic approach SynRBL. Since this dataset lacks ground truth, we report only the number of balanced reactions in Table 3. In our experiments with benchmarking datasets, we observe that the gap between the percentage of balanced reactions and accuracy can be large. Moreover, the transformer models are trained on curated data, which means the reactions from the uncurated part of the dataset are out-of-distribution.

To estimate the accuracy, we can use the probability scores that we determined to be a more reliable estimation than the round-trip accuracy, as described in Section 3D. Figure 12 shows the number of reactions categorized by the number of predicted tokens and the prediction probability scores of the balanced predictions generated by the baseline Chemical Reaction Balancer using normal beam search. The figure reveals that the majority of predictions have low probability scores, which indicates they are most likely inaccurate. Predictions made with constrained beam search demonstrate a similar distribution. However, given that the reactions are out-of-distribution, a lower probability score is not unexpected. To verify the true accuracy of these predictions, manual verification of the reactions using expert domain knowledge is required, but out of scope for this work.

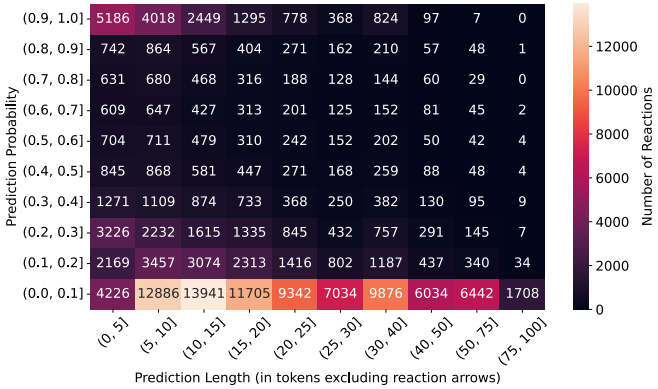
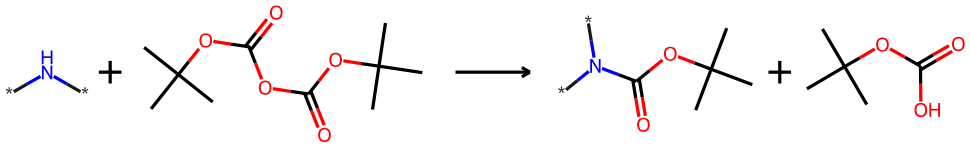


Fig. 12. Number of reactions binned by the number of predicted tokens and prediction probability scores of the balanced predictions made by the baseline Chemical Reaction Balancer with normal beam search on the uncurated USPTO reactions.

F. Reaction classes

The Reaction Balancer models developed in this study perform better across reaction classes compared to the original Reaction Balancer. Figure 11 shows the fraction of balanced reactions from the USPTO 50k datasets for each reaction class. The performance of SynRBL, along with all Chemical Reaction Balancer models, is relatively consistent across the reaction classes, suggesting that training on reactions from all reaction classes results in improved generalizability across reaction classes. The Original Reaction Balancer has lower and more variable performance across reaction classes, failing to predict any balanced reactions for reaction class 6. This discrepancy is likely due to the original model being trained on a dataset curated by an algorithm that did not perform well across imbalanced reactions of all reaction classes, as opposed to our models.

The SynRBL accuracy, which is indicated by the brighter part of the bars in the plot, is comparable among the Chemical



N.CC(C)(C)OC(=O)OC(=O)OC(C)(C)C > > CC(C)(C)OC(=O)N(*)*.CC(C)(C)OC(=O)O

Fig. 13. Molecular structures and SMILES representation of the general format of protection reactions from the USPTO 50k dataset in which di-tert-butyl dicarbonate ($C_{10}H_{18}O_5$) is converted into tert-butyl hydrogen carbonate ($C_5H_{10}O_3$). Asterisks represent varying side groups. In the reported reaction SMILES in the dataset tert-butyl hydrogen carbonate is missing.

Reaction Balancer models but shows some deviations for the fine-tuned Biochemical Reaction Balancer models. Especially on the reactions from class 7 (reductions) and 8 (oxidations) the models achieve a low SynRBL accuracy. Figure S2 shows that only a small fraction of the dataset belong to reaction class 8, of which most reactions originate from the 2015 dataset. To explore this further, we split the performance on this dataset based on the source years (2015 and 2016), as shown in Figure S13.

Performance on the 2015 dataset

Examining the performance on the 2015 dataset reveals overall lower results for all trained models. We observe that the reactions of the 2015 dataset have a higher average absolute atom-imbalance than reactions of the 2016 dataset as shown in Figure S14, which makes them more difficult to complete. Most likely this difference is caused by the preprocessing of the data. The 2016 dataset underwent preprocessing of reagents, while the 2015 dataset did not, resulting in reagents still being present on the reactant side. While our models were trained on data with these reagents added to part of the training instances, they still struggle with this dataset. However, SynRBL still demonstrates reasonable performance, indicating its ability to identify these reagents. However, the average confidence of SynRBL for these reactions is lower than for the reactions where the reagents are discarded from the reactants. Removing the non-atom mapped reactants, which are likely reagents, from the 2015 dataset is expected to improve the performance of our trained machine learning models.

Performance on the 2016 dataset

For the 2016 dataset, the performance of the Chemical Reaction Balancer models approaches 100%. An exception is noted in class 5 (protections), where the constrained Chemical Reaction Balancer with $\lambda = 1$ and normal beam search performs worse

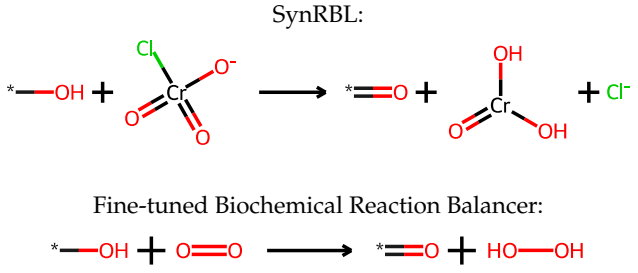


Fig. 14. Comparison of oxidation reactions of reaction class 8 missing hydrogen are completed by: SynRBL (top) and the fine-tuned Biochemical Reaction Balancer (bottom). The completion task is illustrated by the molecules marked with asterisks, which represent varying side groups.

than the other models. The reactions for which this model fails all involve di-tert-butyl dicarbonate ($C_{10}H_{18}O_5$) and appear to be missing only tert-butyl hydrogen carbonate ($C_5H_{10}O_3$) on the product side as is illustrated in Figure 13. The SMILES representation of this missing molecule is CC(C)(C)OC(=O)O. Interestingly, the constrained Chemical Reaction Balancer model with $\lambda = 1$ correctly predicts this molecule for more than half of the tasks involving this format. However, in 42% of these reactions it incorrectly predicts another molecule where oxygen appears earlier in the sequence, such as CC(C)(O)C(=O)O, CC(O)C(=O)O and CC(O)(C)OC(=O)O. This issue may arise because the atom-balance loss primarily focuses on the atom types that are most imbalanced. At the beginning of the sequence, carbon is the most imbalanced atom type, as the reaction is missing five carbon atoms. After predicting two carbon atoms, both the oxygen and carbon imbalances reduce to three atoms. Since this model is not using constrained beam search to prevent imbalanced reactions and was trained with a strong emphasis on atom-balance loss, it suggests that the atom-balance loss has not effectively taught the model to count the atoms accurately.

In the 2016 dataset the fine-tuned Biochemical Reaction Balancer still has the lowest SynRBL accuracy on the reactions in the reduction (7) and oxidation (8) classes. Further investigation reveals that most reduction reactions involve only hydrogen and possibly oxygen imbalances. SynRBL balances these reactions by adding molecular hydrogen (H_2) and, if necessary, water (H_2O) molecules. In contrast, the fine-tuned Biochemical Reaction Balancer models are often not able to solve these reactions. When only hydrogen is missing in many predictions no molecules are added. If oxygen is also imbalanced, the predictions contain a variety of different molecules.

In the case of oxidations, the fine-tuned Biochemical Reaction Balancer appears to have adopted a different strategy. Most oxidation reactions in the 2016 dataset involve a molecule with an alcohol group being converted to a carbonyl group, resulting in a two-hydrogen atom imbalance on the product side. Figure 14 illustrates how the methods solve these completion tasks. SynRBL adds chlorochromate (CrO_3Cl^-) to the reactant side (along with pyridinium as a reagent) and chromium compound ($Cr(OH)_2O$) and chloride (Cl_2) to the product side. Pyridinium chlorochromate is commonly used as a reagent in oxidations [38]. However, the fine-tuned Biochemical Reaction Balancer adds molecular oxygen (O_2) as a reactant and water peroxide (H_2O_2) as a product. This difference likely occurs because the biochemical training data contains similar oxidation reactions where molecular oxygen and water peroxide are part of the complete reaction.

This difference also reveals a potential limitation in the curation process of SynRBL. Since oxidation reactions can occur using a variety of molecules, including but not limited to chloro-

mate, the curation by SynRBL may be overly specific. This overspecification could hinder the generalization capabilities of the machine learning models that are trained on these curated reactions. This issue might also arise for reactions of other reaction classes.

G. Fine-tuning with new data

From Table 3, we observe that the fine-tuned Biochemical Reaction Balancer performs better on biochemical reactions than the Biochemical Reaction Balancer, which is trained from scratch on a relatively small dataset. However, the performance of the fine-tuned model on chemical datasets is lower than that of the Chemical Reaction Balancer, suggesting that it may have unlearned or adapted its ability to complete chemical reactions. We observed this adaptation in the oxidation reactions of the USPTO 50k dataset (Figure 14). While we designed the fine-tuned Biochemical Reaction Balancer to perform well on biochemical reaction data, we believe that to achieve strong performance on both datasets, the training sets should be iterated more frequently.

4. CONCLUSION

In this work, we demonstrate the application of soft atom-balance constraints in the loss function during the training phase and hard constraints by restricting the prediction of specified tokens during the inference phase in transformer models. For effective training and evaluation of machine learning models on reaction completion tasks, it is crucial to have extensive and diverse ground truth data. This data should cover all reaction classes to ensure the model's ability to generalize to new, unseen reactions and, specifically for benchmarking purposes, be validated by experts.

Our results indicate that models using soft constraints in the loss function do not lead to improved balancing performance and require further tuning. In contrast, the implementation of hard atom-balance constraints within constrained beam search proves to be the most effective approach for improving the performance of transformer-based models in reaction completion tasks. However, this method carries the risk of inaccurately balancing reactions, a challenge that is difficult to identify without chemical expertise, particularly in the absence of reliable ground truth data.

Recommendations

Based on our findings, we propose the following recommendations to improve the performance and applicability of reaction balancing and prediction models.

First, it is essential to minimize or effectively identify the gap between accuracy and balancing performance. We investigated potential solutions to this issue by using both round-trip accuracy and probability scores of the predictions. Our analysis revealed that probability scores serve as a more reliable metric for evaluation.

Secondly, the models should be benchmarked against state-of-the-art forward prediction and retrosynthesis methods in addition to the completion method SynRBL [39, 40]. The forward prediction model mentioned in this work, the Molecular Transformer, was not designed to output a balanced product side, but solely the main product of the reaction, which makes it unsuitable for benchmarking against our proposed method.

Furthermore, the generalizability could potentially be enhanced by data augmentation methods such as including both directionalities of the training reactions in both directions. Especially in biochemical reactions, both directionalities of the reaction are often of equal interest. Moreover, in inference experiments, the same input reaction could be given in both directionalities to compare and verify their outcomes.

Most importantly, high-quality benchmarking data remains essential for verifying the performance of our proposed methods. Robust datasets are needed for accurate assessments and comparisons of various approaches.

This work is primarily focused on the completion of the reactants and products in valid partial reactions. Our models are not able to repair invalid molecules or reactions, nor predict missing solvents, reagents and catalysts. These limitations highlight areas for potential improvement. Future work could enhance the model's capabilities by incorporating features that allow for the identification and correction of invalid components. Additionally, providing the model with more contextual information, such as details about reported reaction conditions, solvents, reagents, and catalysts, could improve its predictive accuracy and overall performance.

Lastly, we want to highlight applications of constrained beam search. The constrained beam search approach has potential applications beyond reaction balancing. It can be used in related fields where certain tokens must be restricted at predefined points in the prediction process, based on the input and the previously generated part of the sequence, such as in automated chemical flowsheets [41].

ACKNOWLEGDEMENTS

Research reported in this work was partially or completely facilitated by computational resources and support of the Delft AI Cluster (DAIC) at TU Delft (RRID: SCR_025091), but remains the sole responsibility of the authors, not the DAIC team [42].

CODE AVAILABILITY

The code that was used to conduct the preliminary experiments is available at https://github.com/mnoordsij/USPTO-analysis. The code that was used to train and evaluate the machine learning models presented in this work is available at https://github.com/Intelligent-molecular-systems/constrained-reaction-balancer.

REFERENCES

1. D. M. Lowe, "Extraction of chemical structures and reactions from the literature," Ph.D. thesis, Apollo - University of Cambridge Repository (2012).
2. P. Schwaller, T. Laino, T. Gaudin, *et al.*, "Molecular transformer: A model for uncertainty-calibrated chemical reaction prediction," ACS Cent. Sci. **5**, 1572–1583 (2019).
3. H. Gao, T. J. Struble, C. W. Coley, *et al.*, "Using machine learning to predict suitable conditions for organic reactions," ACS Cent. Sci. **4**, 1465–1476 (2018).
4. A. Puliyanda, K. Srinivasan, K. Sivaramakrishnan, and V. Prasad, "A review of automated and data-driven approaches for pathway determination and reaction monitoring in complex chemical systems," Digit. Chem. Eng. **2**, 100009 (2022).
5. J. M. Weber, Z. Guo, C. Zhang, *et al.*, "Chemical data intelligence for sustainable chemistry," Chem. Soc. Rev. **50**, 12013–12036 (2021).
6. J. M. Weber, Z. Guo, and A. A. Lapkin, "Discovering circular process solutions through automated reaction network optimization," ACS Eng. Au **2**, 333–349 (2022).
7. A. C. Vaucher, P. Schwaller, and T. Laino, "Completion of partial reaction equations," in *Machine Learning for Molecules Workshop at NeurIPS*, (2020).

8. T. L. Phan, K. Weinbauer, T. Gärtner, *et al.*, "Reaction rebalancing: a novel approach to curating reaction databases," J. Cheminformatics **16** (2024).
9. C. Zhang, A. Arun, and A. A. Lapkin, "Completing and balancing database excerpted chemical reactions with a hybrid mechanistic-machine learning approach," ACS Omega **9**, 18385–18399 (2024).
10. M. van Wijngaarden, G. Vogel, and J. M. Weber, "Completing partial reaction equations with rule and language model-based methods," Comput. Aided Chem. Eng. **53**, 3139–3144 (2024).
11. F. Zipoli, Z. Ayadi, P. Schwaller, *et al.*, "Completion of partial chemical equations," Mach. Learn. Sci. Technol. **5** (2024).
12. Y. Liu, M. Ott, N. Goyal, *et al.*, "Roberta: A robustly optimized BERT pretraining approach," CoRR **abs/1907.11692** (2019).
13. H. Chen, G. E. Flores, and C. Li, "Physics-informed neural networks with hard linear equality constraints," Comput. Chem. Eng. **189** (2024).
14. G. E. Karniadakis, I. G. Kevrekidis, L. Lu, *et al.*, "Physics-informed machine learning," Nat. Rev. Phys. 2021 3:6 **3**, 422–440 (2021).
15. J. Ren, C. Hu, Z. Shang, *et al.*, "Learning interpretable and transferable representations via wavelet-constrained transformer for industrial acoustic diagnosis," IEEE Trans. on Instrum. Meas. (2025).
16. F. Liu, Y. Cao, X. Cheng, and X. Wu, "Transformer-based local-to-global lidar-camera targetless calibration with multiple constraints," IEEE Trans. on Instrum. Meas. **73** (2024).
17. P. Anderson, B. Fernando, M. Johnson, and S. Gould, "Guided open vocabulary image captioning with constrained beam search," in *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing,* M. Palmer, R. Hwa, and S. Riedel, eds. (Association for Computational Linguistics, Copenhagen, Denmark, 2017), pp. 936–945.
18. M. Post and D. Vilar, "Fast lexically constrained decoding with dynamic beam allocation for neural machine translation," CoRR **abs/1804.06609** (2018).
19. J. E. Hu, H. Khayrallah, R. Culkin, *et al.*, "Improved lexically constrained decoding for translation and monolingual rewriting," in *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers),* J. Burstein, C. Doran, and T. Solorio, eds. (Association for Computational Linguistics, Minneapolis, Minnesota, 2019), pp. 839–850.
20. Z. Li, X. Ding, T. Liu, *et al.*, "Guided generation of cause and effect," in *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence,* (International Joint Conferences on Artificial Intelligence Organization, 2020), IJCAI-PRICAI-2020, p. 3629–3636.
21. T. Wolf, L. Debut, V. Sanh, *et al.*, "Huggingface's transformers: State-of-the-art natural language processing," CoRR **abs/1910.03771** (2019).
22. D. Weininger, "Smiles, a chemical language and information system: 1: Introduction to methodology and encoding rules," J. Chem. Inf. Comput. Sci. **28**, 31–36 (1988).
23. D. Weininger, A. Weininger, and J. L. Weininger, "Smiles. 2. algorithm for generation of unique smiles notation," J. chemical information computer sciences **29**, 97–101 (1989).
24. P. Schwaller, T. Gaudin, D. Lányi, *et al.*, ""found in translation": predicting outcomes of complex organic chemistry reactions using neural sequence-to-sequence models," Chem. Sci. **9**, 6091–6098 (2018).
25. N. Schneider, D. M. Lowe, R. A. Sayle, and G. A. Landrum, "Development of a novel fingerprint for chemical reactions and its application to large-scale reaction classification and similarity," J. Chem. Inf. Model. **55**, 39–53 (2015).
26. N. Schneider, N. Stiefl, and G. A. Landrum, "What's what: The (nearly) definitive guide to reaction role assignment," J. Chem. Inf. Model. **56**, 2336–2346 (2016).
27. N. S. Limited, "Namerxn," https://www.nextmovesoftware.com/namerxn.html (2015). Accessed on 27/01/2025.
28. A. Vaswani, N. Shazeer, N. Parmar, *et al.*, "Attention is all you need," Adv. Neural Inf. Process. Syst. **2017-December**, 5999–6009 (2017).
29. G. Klein, Y. Kim, Y. Deng, *et al.*, "OpenNMT: Open-source toolkit for neural machine translation," in *Proceedings of ACL 2017, System Demonstrations,* (Association for Computational Linguistics, Vancouver, Canada, 2017), pp. 67–72.
30. A. Paszke, S. Gross, F. Massa, *et al.*, "Pytorch: An imperative style, high-performance deep learning library," in *Advances in Neural Information Processing Systems 32,* H. Wallach, H. Larochelle, A. Beygelzimer, *et al.*, eds. (Curran Associates, Inc., 2019), pp. 8024–8035.
31. D. M. Lowe (2017). Chemical reactions from US patents (1976 - Sep 2016) https://figshare.com/articles/Chemical_reactions_from_US_patents_1976-sep2016_/5104873.
32. G. Landrum, "Rdkit: Open-source cheminformatics." .
33. J. Bradshaw, A. Zhang, B. Mahjour, *et al.*, "Challenging reaction prediction models to generalize to novel chemistry," ACS Cent. Sci. **11**, 539–549 (2025).
34. P. Bansal, A. Morgat, K. B. Axelsen, *et al.*, "Rhea, the reaction knowledgebase in 2022," Nucleic acids research **50**, D693–D700 (2022).
35. W. Jaworski, S. Szymkuć, B. Mikulak-Klucznik, *et al.*, "Automatic mapping of atoms across both simple and complex chemical reactions," Nat. Commun. 2019 10:1 **10**, 1–11 (2019).
36. A. Lin, N. Dyubankova, T. I. Madzhidov, *et al.*, "Atom-to-atom mapping: A benchmarking study of popular mapping algorithms and consensus strategies," Mol. Informatics **41**, 2100138 (2022).
37. D. Probst, M. Manica, Y. G. N. Teukam, *et al.*, "Biocatalysed synthesis planning using data-driven learning," Nat. Commun. 2022 13:1 **13**, 1–11 (2022).
38. E. Corey and J. Suggs, "Pyridinium chlorochromate. an efficient reagent for oxidation of primary and secondary alcohols to carbonyl compounds," Tetrahedron Lett. **16**, 2647–2650 (1975).
39. B. Delépine, T. Duigou, P. Carbonell, and J. L. Faulon, "Retropath2.0: A retrosynthesis workflow for metabolic engineers," Metab. Eng. **45**, 158–170 (2018).
40. S. Zheng, T. Zeng, C. Li, *et al.*, "Bionavi-np: Biosynthesis navigator for natural products," (2021).
41. G. Vogel, L. S. Balhorn, and A. M. Schweidtmann, "Learning from flowsheets: A generative transformer model for autocompletion of flowsheets," Comput. Chem. Eng. **171**, 108162 (2023).
42. Delft AI Cluster (DAIC), "The delft ai cluster (daic), rrid:scr_025091," (2024).

SUPPLEMENTARY INFORMATION

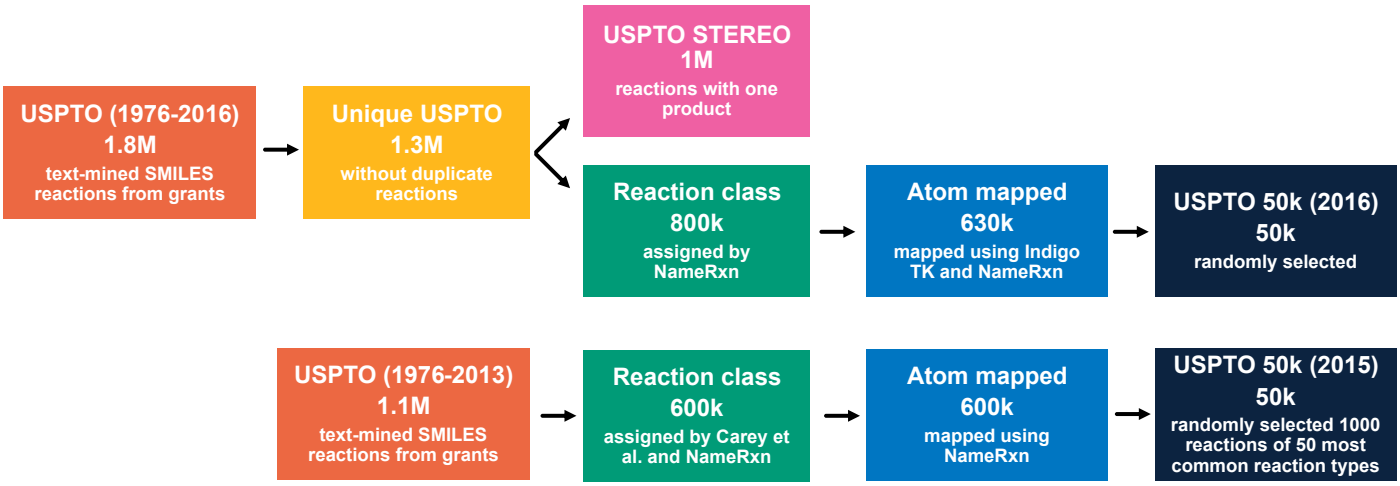
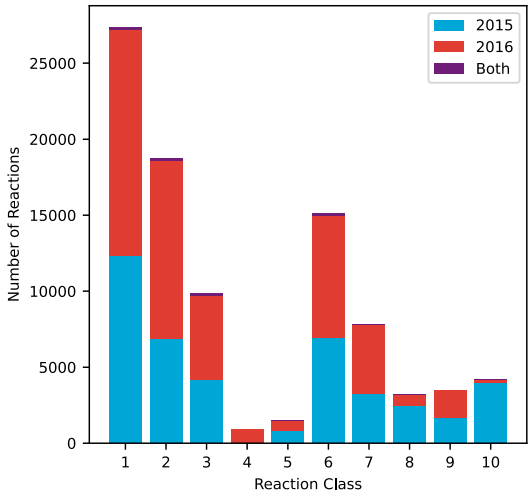


Fig. S1. Subsets of the United States Patent Trademark Office (USPTO) reaction database [1]. The number of reactions in each dataset is denoted under the name of the datasets. The full dataset of 1.8M reactions, text-mined from grants between 1976 and 2016, contains 1.3M unique reactions of which 1M reactions have a single product (USPTO STEREO) and of which a subset is publicly available of 50,000 randomly selected reactions with class labels (USPTO 50k) [24, 26]. From a dataset with 1.1M reactions, text-mined from grants between 1976 and 2013, exists a publicly available subset of 1000 reactions of the 50 most common (three-level) reaction classes [25]



Reaction Classes
1. Heteroatom alkylation and arylation
2. Acylation and related processes
3. C-C bond formation
4. Heterocycle formation
5. Protections
6. Deprotections
7. Reductions
8. Oxidations
9. Functional group interconversion (FGI)
10. Functional group addition (FGA)

Fig. S2. The distribution of reaction superclasses of the reactions in the USPTO 50k dataset of 2015 [25] and 2016 [26].

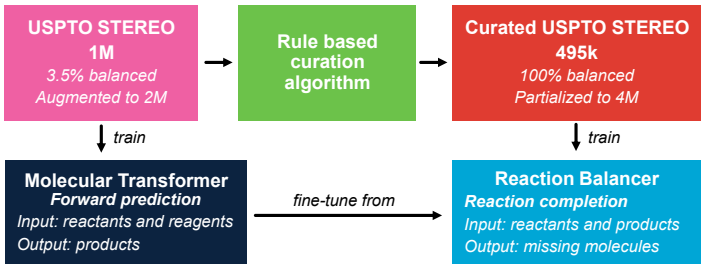


Fig. S3. Overview of the data and models that were used to train the Reaction Balancer [10]. The Molecular Transformer was trained on forward prediction tasks using the 1M reactions of USPTO STEREO as training dataset [2, 24]. The USPTO STEREO data was partly curated to 495,197 atom-balanced reactions, which were used to train the Reaction Balancer on reaction completion tasks.

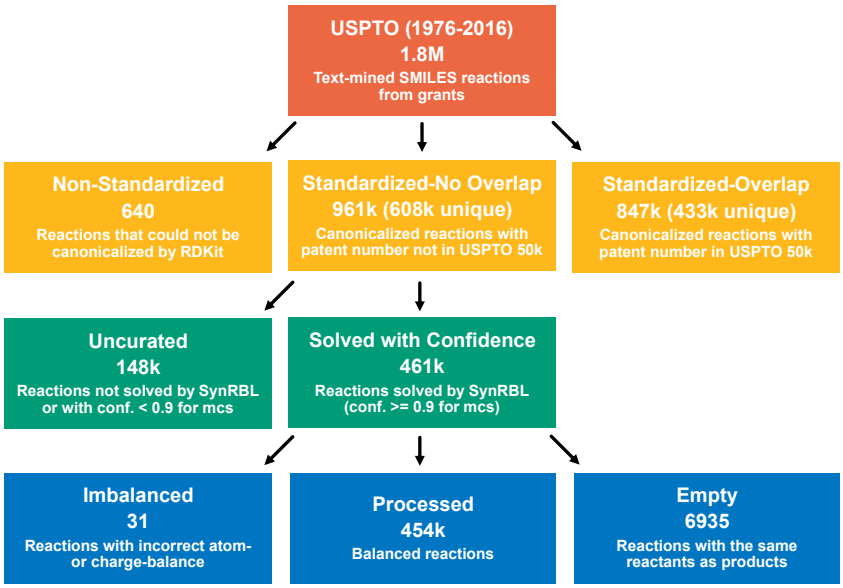


Fig. S4. Preprocessing steps applied to the chemical reactions in the USPTO database [1]. Dataset sizes are denoted below the name of each subset. The preprocessing steps are described in Section 2D.

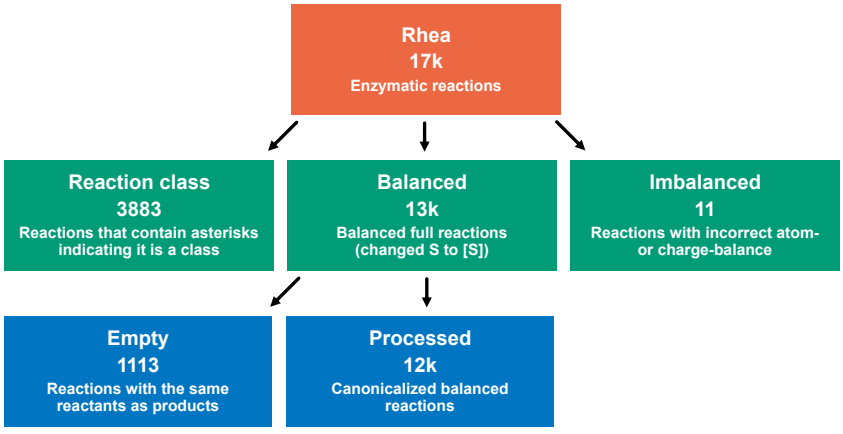


Fig. S5. Preprocessing steps applied to the chemical reactions in the Rhea database [34]. Dataset sizes are denoted below the name of each subset. The preprocessing steps are described in Section 2D.

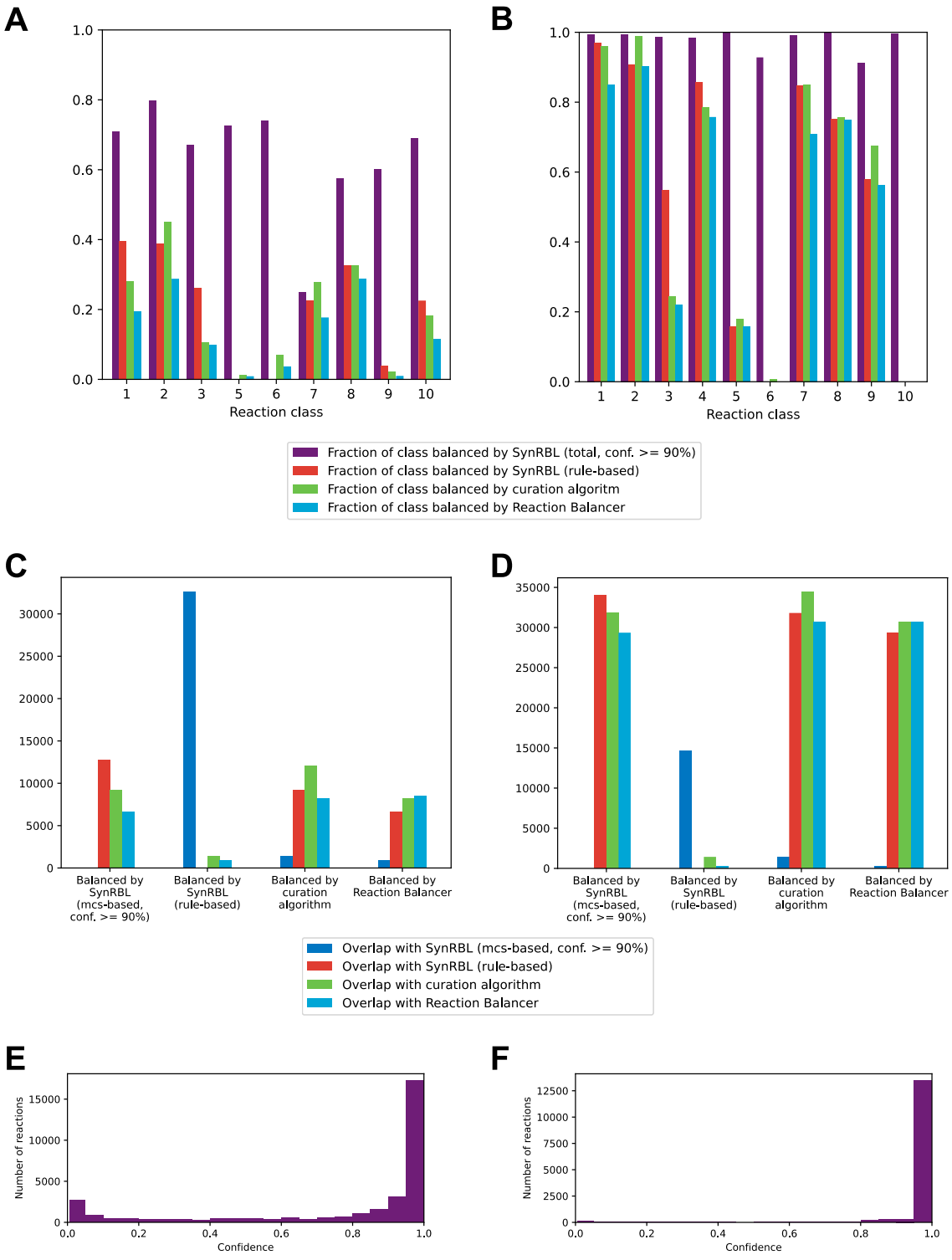


Fig. S6. **A)** Fraction of the 49,230 imbalanced reactions of the USPTO 50k dataset of 2015 balanced by existing reaction balancing approaches [8, 10, 25]. **B)** Fraction of the 48,809 imbalanced reactions of the USPTO 50k dataset of 2016 balanced by existing reaction balancing approaches [8, 10, 26]. **C)** Overlap of the reactions of USPTO 50k (2015) solved by SynRBL, the Reaction Balancer and its curation algorithm [8, 10, 25]. The reactions solved by SynRBL are split in rule-based or MCS-based, so they show no overlap [8]. **D)** Overlap of the reactions of USPTO 50k (2016) solved by SynRBL, the Reaction Balancer and its curation algorithm [8, 10, 26]. The reactions solved by SynRBL are split in rule-based or MCS-based, so they show no overlap [8]. **E)** Confidence distribution of the USPTO 50k (2015) reactions solved by SynRBL with MCS-based approach. **F)** Confidence distribution of the USPTO 50k (2016) reactions solved by SynRBL with MCS-based approach.

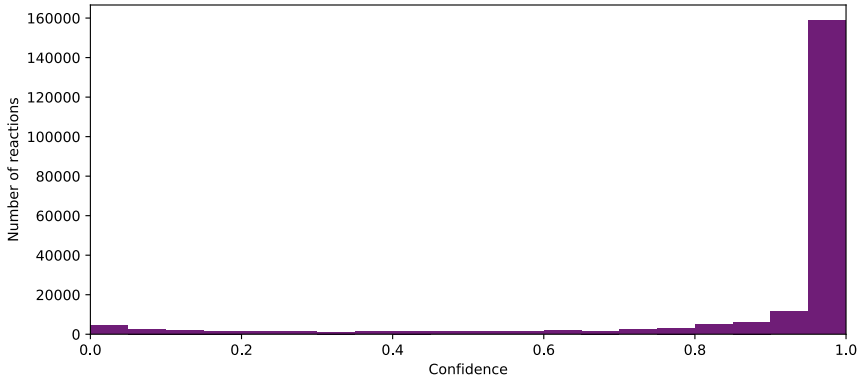


Fig. S7. Confidence distribution of the 502,761 USPTO reactions solved by SynRBL with MCS-based approach. The confidence values have a mean of 0.893, median of 0.994 and stanndard deviation of 0.238. The 170,253 reactions with confidence ≥ 0.9 were included in our training dataset. In this selected set 158,681 reactions have confidence ≥ 0.95 and 125,057 confidence ≥ 0.99.

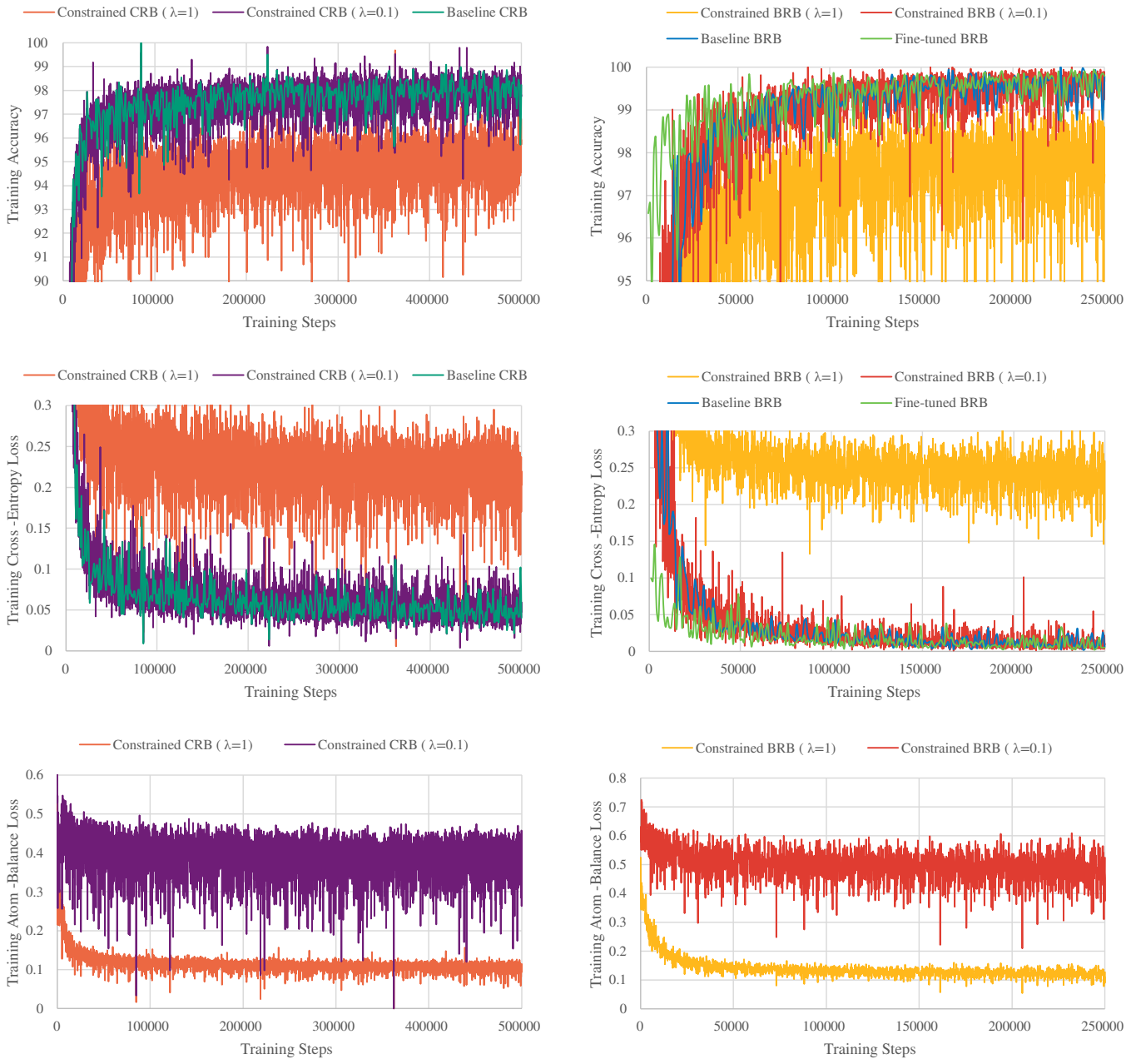


Fig. S8. Accuracy, cross-entropy loss and atom-values loss values reported during training our Chemical Reaction Balancer (CRB) and Biochemical Reaction Balancer (BRB) models. For the baseline and fine-tuned models the values are reported every 1,000 steps and for the constrained models the values are reported every 100 steps.

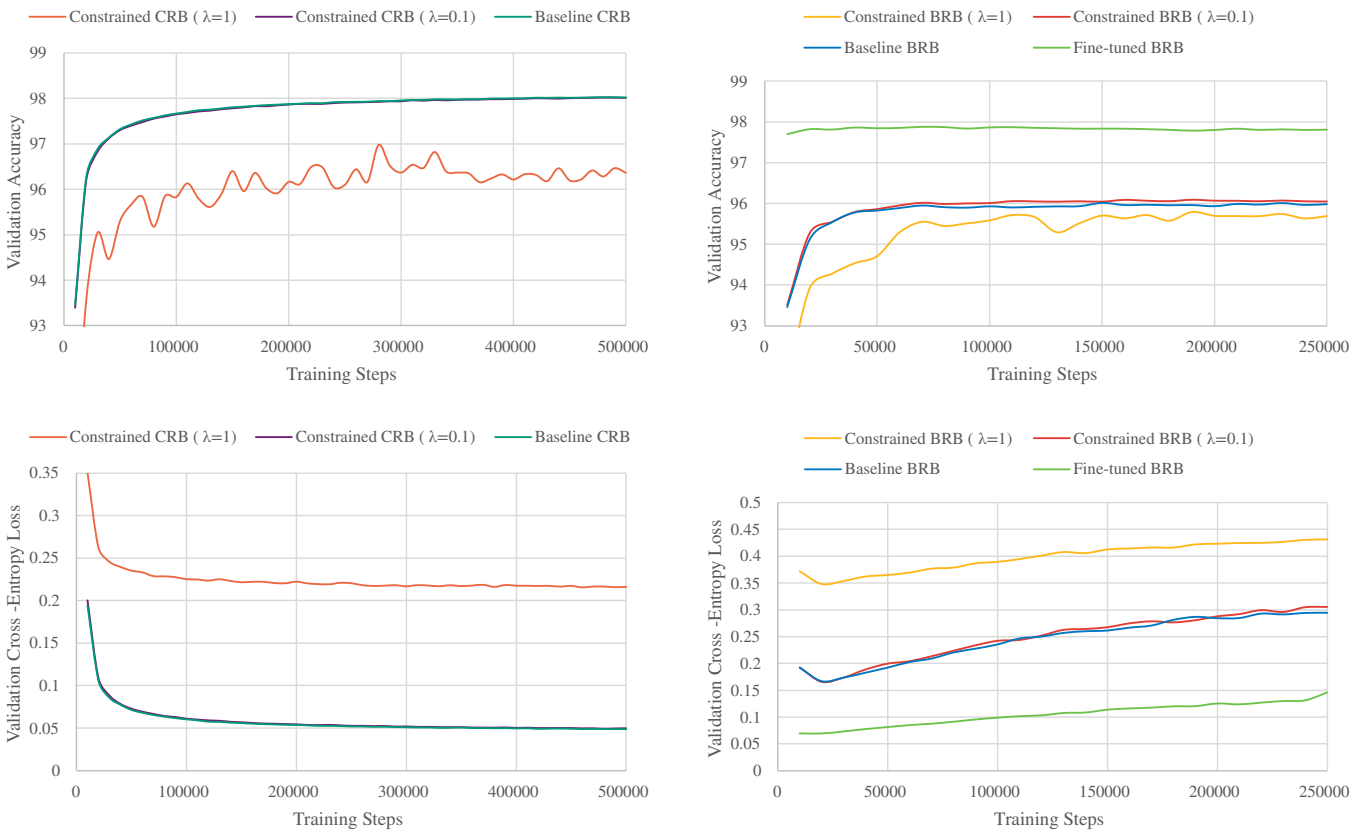


Fig. S9. Accuracy, cross-entropy loss and atom-values loss values reported during training our Chemical Reaction Balancer (CRB) and Biochemical Reaction Balancer (BRB) models. The values are reported every 10,000 steps.

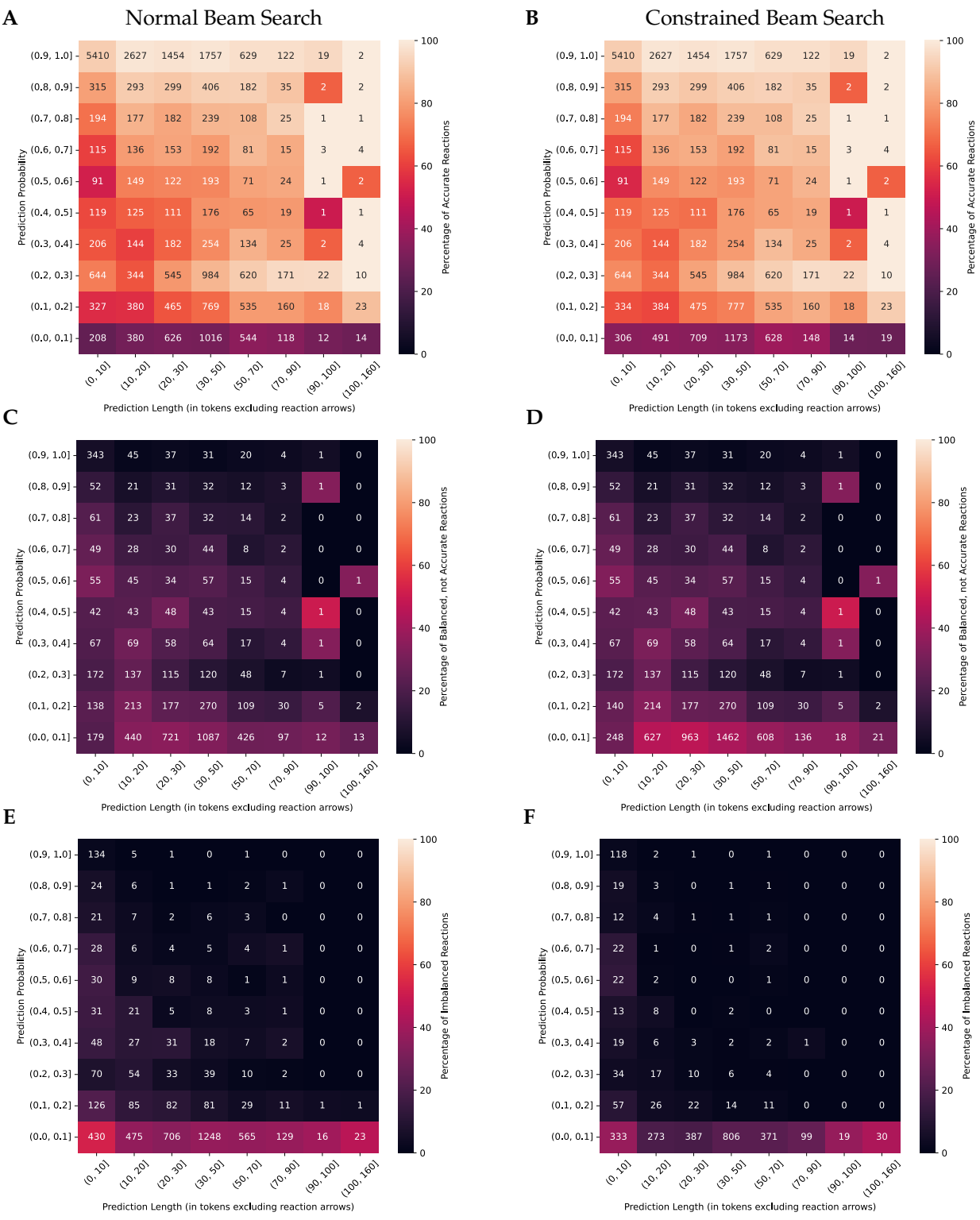
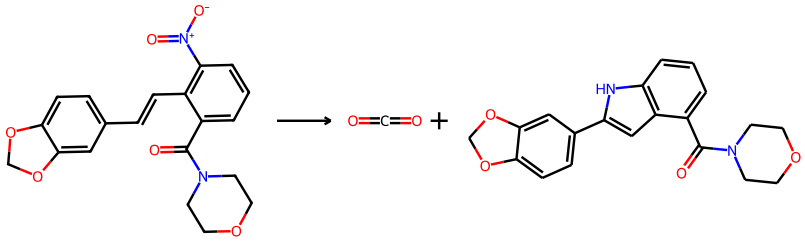


Fig. S10. Heatmaps of the number of reactions binned by number of predicted tokens and prediction probability scores of the baseline Chemical Reaction Balancer with normal (A, C, E) and constrained (B, D, F) beam search on the partialized validation set of SynRBL for accurate, balanced inaccurate and imbalanced inaccurate predictions. The heatmaps are coloured based in the percentage of accurate (A, B), balanced inaccurate (C, D) or imbalanced inaccurate (E, F) reactions with the same probability and length range.

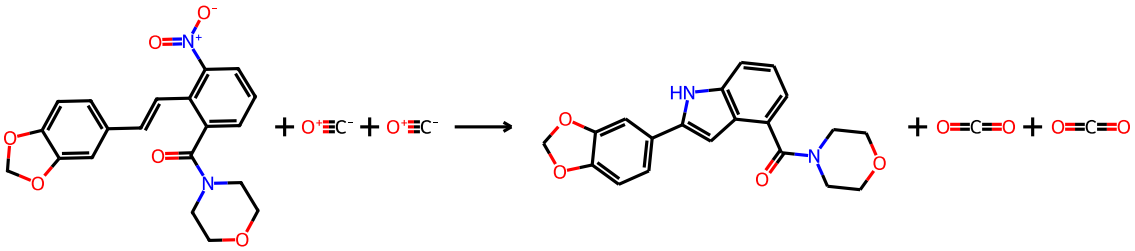
Input reaction:

O=C(c1cccc([N+](=O)[O-])c1/C=C/c1ccc2c(c1)OCO2)N1CCOCC1 > >
O=C=O.O=C(c1cccc2[nH]c(-c3ccc4c(c3)OCO4)cc12)N1CCOCC1



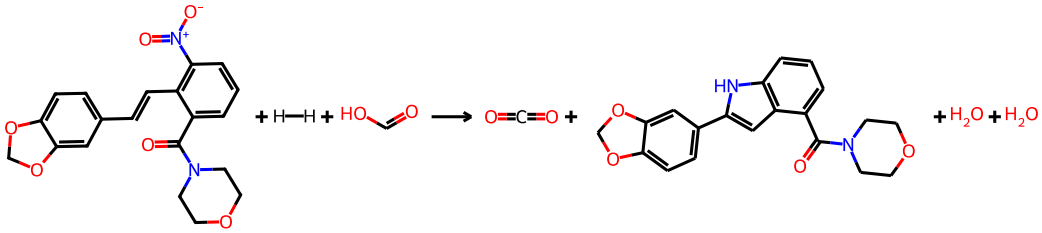
Ground truth reaction:

O=C(c1cccc([N+](=O)[O-])c1/C=C/c1ccc2c(c1)OCO2)N1CCOCC1.[C-]#[O+].[C-]#[O+] > >
O=C(c1cccc2[nH]c(-c3ccc4c(c3)OCO4)cc12)N1CCOCC1.O=C=O.O=C=O



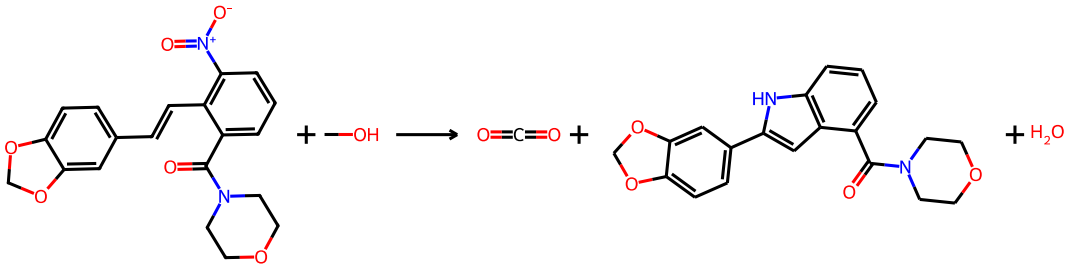
Output C ($\lambda = 1$) CRB:

O=C(c1cccc([N+](=O)[O-])c1/C=C/c1ccc2c(c1)OCO2)N1CCOCC1.[H][H].O=CO > >
O=C=O.O=C(c1cccc2[nH]c(-c3ccc4c(c3)OCO4)cc12)N1CCOCC1.O.O



Output C ($\lambda = 1$) CRB with CBS:

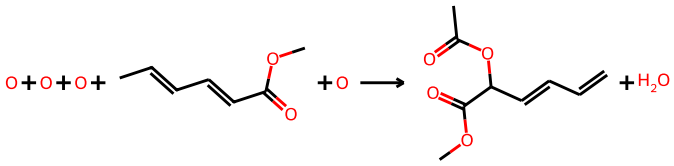
O=C(c1cccc([N+](=O)[O-])c1/C=C/c1ccc2c(c1)OCO2)N1CCOCC1.CO > >
O=C=O.O=C(c1cccc2[nH]c(-c3ccc4c(c3)OCO4)cc12)N1CCOCC1.O



(a) Input, ground truth and output SMILES and reaction from the partialized validation set for which the constrained ($\lambda = 1$) Chemical Reaction Balancer without constrained beam search produced a balanced output and with constrained beam search produced an imbalanced output. Both outputs do not match the ground truth.

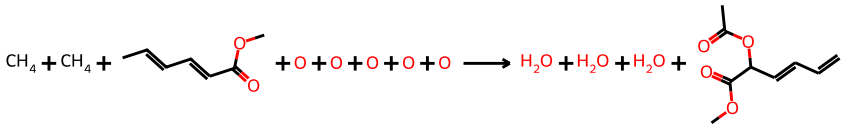
Input reaction:

`[O].[O].[O].CC=CC=CC(=O)OC.[O] > > C=CC=CC(OC(C)=O)C(=O)OC.O`



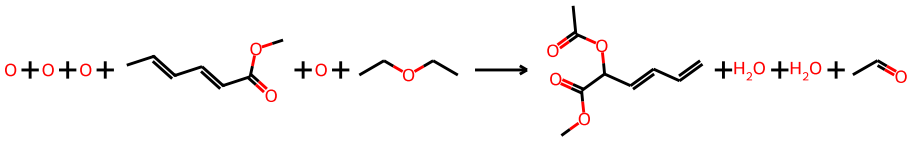
Ground truth reaction:

`C.C.CC=CC=CC(=O)OC.[O].[O].[O].[O].[O] > > O.O.O.C=CC=CC(OC(C)=O)C(=O)OC`



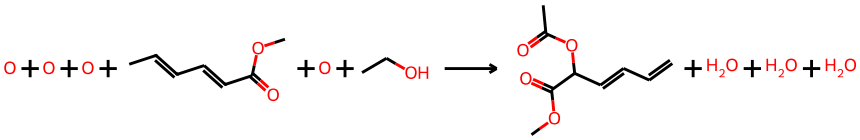
Output C ($\lambda = 1$) CRB:

`[O].[O].[O].CC=CC=CC(=O)OC.[O].CCOCC > > C=CC=CC(OC(C)=O)C(=O)OC.O.O.CC=O`



Output C ($\lambda = 1$) CRB with CBS:

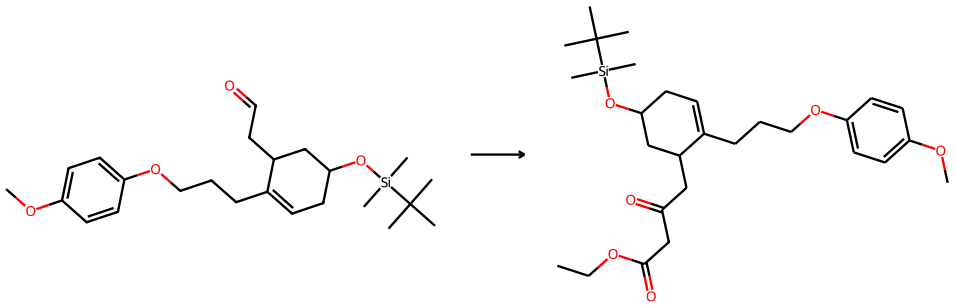
`[O].[O].[O].CC=CC=CC(=O)OC.[O].CCO > > C=CC=CC(OC(C)=O)C(=O)OC.O.O.O`



(b) Input, ground truth and output SMILES and reaction from the partialized validation set for which the constrained ($\lambda = 1$) Chemical Reaction Balancer without constrained beam search produced a balanced output and with constrained beam search produced an imbalanced output. Both outputs do not match the ground truth.

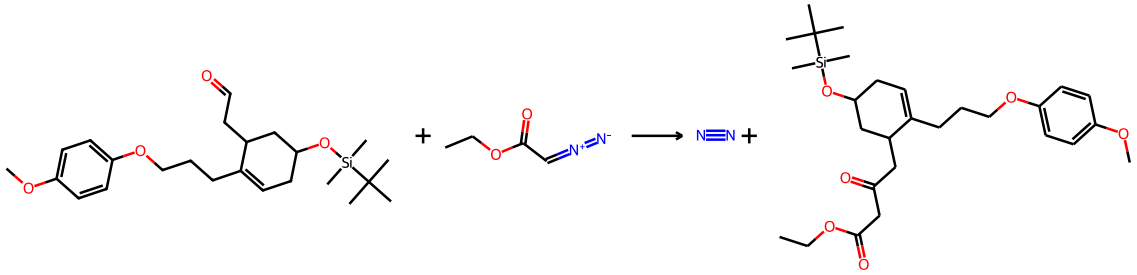
Input reaction:

COc1ccc(OCCCC2=CCC(O[Si](C)(C)C(C)(C)C)CC2CC=O)cc1 > >
CCOC(=O)CC(=O)CC1CC(O[Si](C)(C)C(C)(C)C)CC=C1CCCOc1ccc(OC)cc1



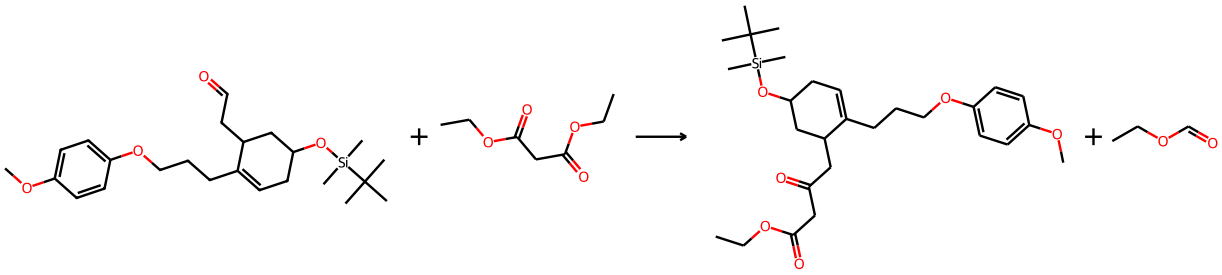
Ground truth reaction:

COc1ccc(OCCCC2=CCC(O[Si](C)(C)C(C)(C)C)CC2CC=O)cc1.CCOC(=O)C=[N+]=[N-] > >
N#N.CCOC(=O)CC(=O)CC1CC(O[Si](C)(C)C(C)(C)C)CC=C1CCCOc1ccc(OC)cc1



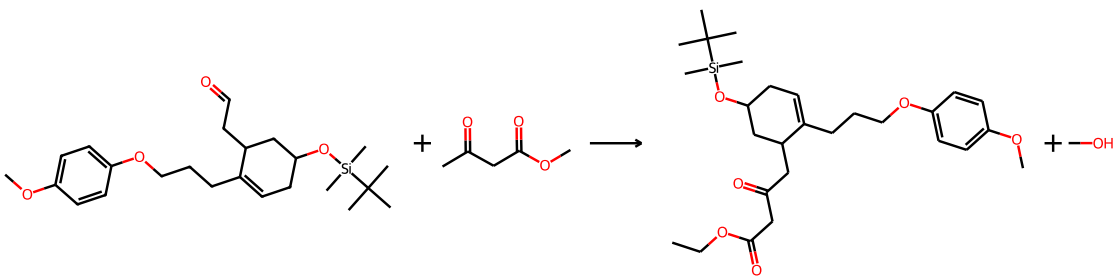
Output C ($\lambda = 1$) CRB:

COc1ccc(OCCCC2=CCC(O[Si](C)(C)C(C)(C)C)CC2CC=O)cc1.CCOC(=O)CC(=O)OCC > >
CCOC(=O)CC(=O)CC1CC(O[Si](C)(C)C(C)(C)C)CC=C1CCCOc1ccc(OC)cc1.CCOC=O



Output C ($\lambda = 1$) CRB with CBS:

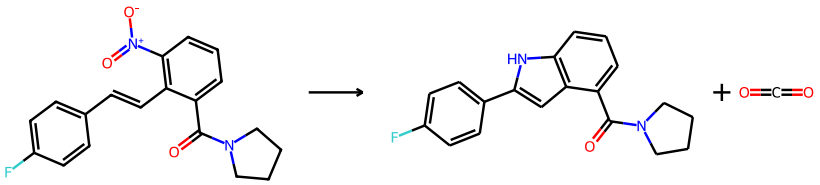
COc1ccc(OCCCC2=CCC(O[Si](C)(C)C(C)(C)C)CC2CC=O)cc1.COC(=O)CC(C)=O > >
CCOC(=O)CC(=O)CC1CC(O[Si](C)(C)C(C)(C)C)CC=C1CCCOc1ccc(OC)cc1.CO



(c) Input, ground truth and output SMILES and reaction from the partialized validation set for which the constrained ($\lambda = 1$) Chemical Reaction Balancer without constrained beam search produced a balanced output and with constrained beam search produced an imbalanced output. Both outputs do not match the ground truth.

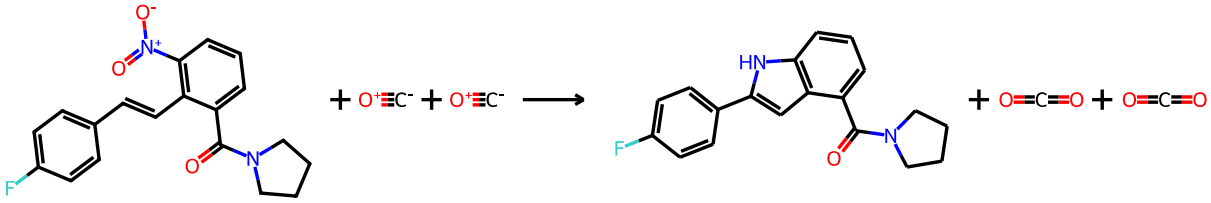
Input reaction:

O=C(c1cccc([N+](=O)[O-])c1/C=C/c1ccc(F)cc1)N1CCCC1 > > O=C(c1cccc2[nH]c(-c3ccc(F)cc3)cc12)N1CCCC1.O=C=O



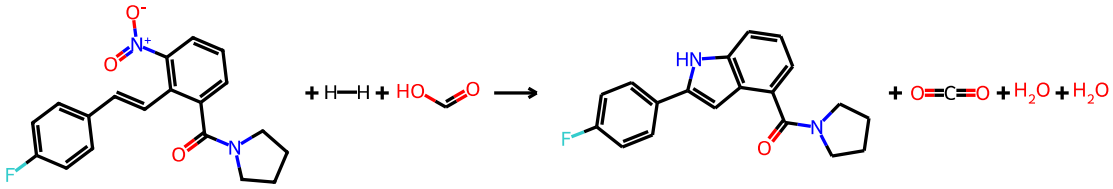
Ground truth reaction:

O=C(c1cccc([N+](=O)[O-])c1/C=C/c1ccc(F)cc1)N1CCCC1.[C-]#[O+].[C-]#[O+] > >
O=C(c1cccc2[nH]c(-c3ccc(F)cc3)cc12)N1CCCC1.O=C=O.O=C=O



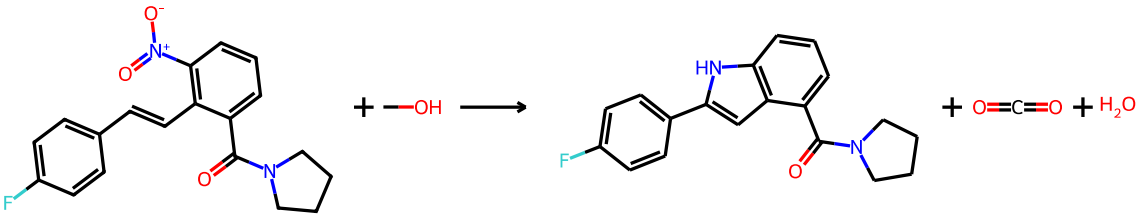
Output C ($\lambda = 1$) CRB:

O=C(c1cccc([N+](=O)[O-])c1/C=C/c1ccc(F)cc1)N1CCCC1.[H][H].O=CO > >
O=C(c1cccc2[nH]c(-c3ccc(F)cc3)cc12)N1CCCC1.O=C=O.O.O



Output C ($\lambda = 1$) CRB with CBS:

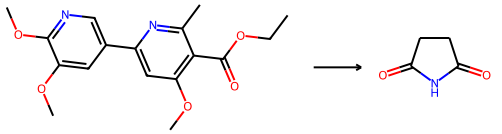
O=C(c1cccc([N+](=O)[O-])c1/C=C/c1ccc(F)cc1)N1CCCC1.CO > > O=C(c1cccc2[nH]c(-c3ccc(F)cc3)cc12)N1CCCC1.O=C=O.O



(d) Input, ground truth and output SMILES and reaction from the partialized validation set for which the constrained ($\lambda = 1$) Chemical Reaction Balancer without constrained beam search produced a balanced output and with constrained beam search produced an imbalanced output. Both outputs do not match the ground truth.

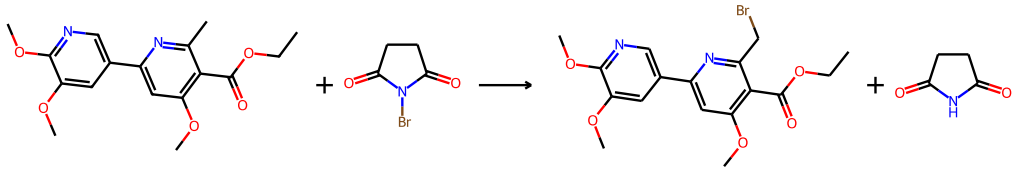
Input reaction:

CCOC(=O)c1c(OC)cc(-c2cnc(OC)c(OC)c2)nc1C > > O=C1CCC(=O)N1



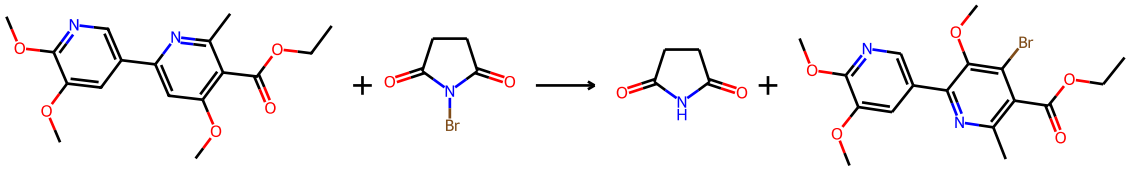
Ground truth reaction:

CCOC(=O)c1c(OC)cc(-c2cnc(OC)c(OC)c2)nc1C.O=C1CCC(=O)N1Br > >
CCOC(=O)c1c(OC)cc(-c2cnc(OC)c(OC)c2)nc1CBr.O=C1CCC(=O)N1



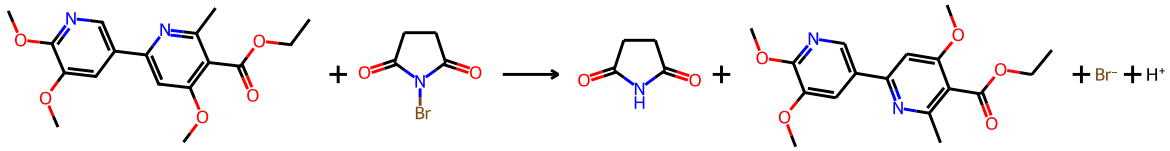
Output C ($\lambda = 1$) CRB:

CCOC(=O)c1c(OC)cc(-c2cnc(OC)c(OC)c2)nc1C.O=C1CCC(=O)N1Br > >
O=C1CCC(=O)N1.CCOC(=O)c1c(C)nc(-c2cnc(OC)c(OC)c2)c(OC)c1Br



Output C ($\lambda = 1$) CRB with CBS:

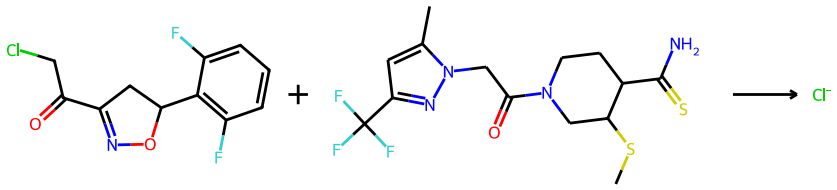
CCOC(=O)c1c(OC)cc(-c2cnc(OC)c(OC)c2)nc1C.O=C1CCC(=O)N1Br > >
O=C1CCC(=O)N1.CCOC(=O)c1c(C)nc(-c2cnc(OC)c(OC)c2)cc1OC.[Br-].[H+]



(e) Input, ground truth and output SMILES and reaction from the partialized validation set for which the constrained ($\lambda = 1$) Chemical Reaction Balancer without constrained beam search produced a balanced output and with constrained beam search produced an imbalanced output. Both outputs do not match the ground truth.

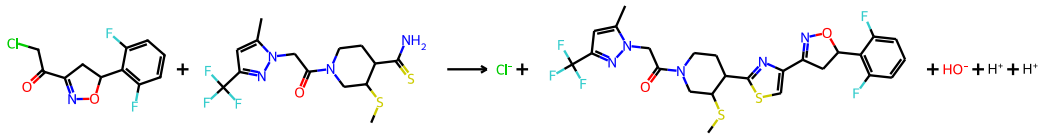
Input reaction:

O=C(CCl)C1=NOC(c2c(F)cccc2F)C1.CSC1CN(C(=O)Cn2nc(C(F)(F)F)cc2C)CCC1C(N)=S > > [Cl-]



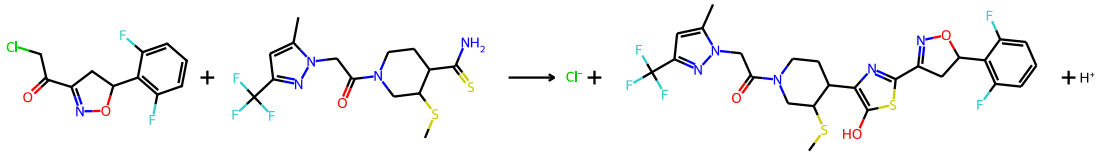
Ground truth and output C ($\lambda = 1$) CRB:

O=C(CCl)C1=NOC(c2c(F)cccc2F)C1.CSC1CN(C(=O)Cn2nc(C(F)(F)F)cc2C)CCC1C(N)=S > >
[Cl-].CSC1CN(C(=O)Cn2nc(C(F)(F)F)cc2C)CCC1c1nc(C2=NOC(c3c(F)cccc3F)C2)cs1.[OH-].[H+].[H+]



Output C ($\lambda = 1$) CRB with CBS:

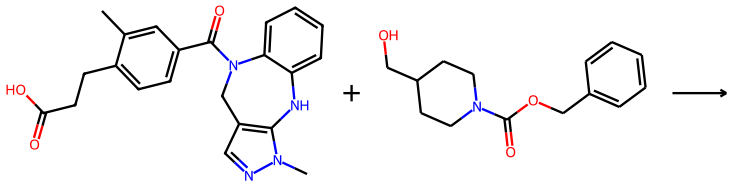
O=C(CCl)C1=NOC(c2c(F)cccc2F)C1.CSC1CN(C(=O)Cn2nc(C(F)(F)F)cc2C)CCC1C(N)=S > >
[Cl-].CSC1CN(C(=O)Cn2nc(C(F)(F)F)cc2C)CCC1c1nc(C2=NOC(c3c(F)cccc3F)C2)sc1O.[H+]



(f) Input and output SMILES and reaction from the partialized validation set for which the constrained ($\lambda = 1$) Chemical Reaction Balancer without constrained beam search produced a balanced output and with constrained beam search produced an imbalanced output. The output of the model without constrained beam search matches the ground truth.

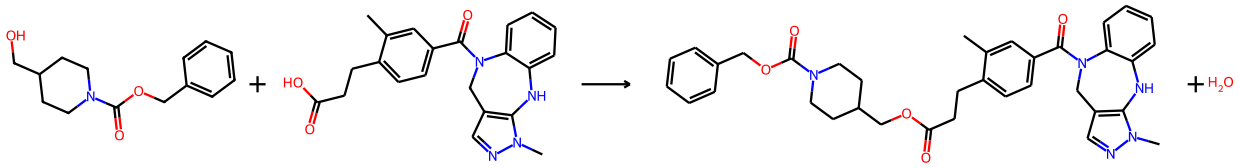
Input reaction:

Cc1cc(C(=O)N2Cc3cnn(C)c3Nc3ccccc32)ccc1CCC(=O)O.O=C(OCc1ccccc1)N1CCC(CO)CC1 > >



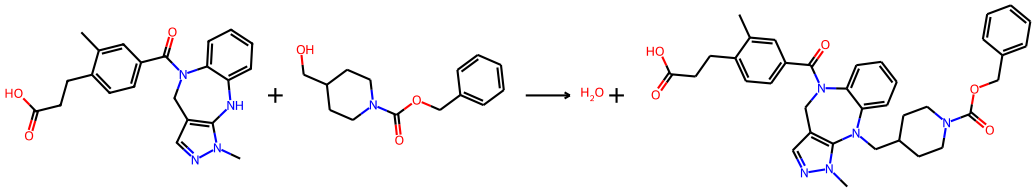
Ground truth:

O=C(OCc1ccccc1)N1CCC(CO)CC1.Cc1cc(C(=O)N2Cc3cnn(C)c3Nc3ccccc32)ccc1CCC(=O)O > >
Cc1cc(C(=O)N2Cc3cnn(C)c3Nc3ccccc32)ccc1CCC(=O)OCC1CCN(C(=O)OCc2ccccc2)CC1.O



Output C ($\lambda = 1$) CRB:

Cc1cc(C(=O)N2Cc3cnn(C)c3Nc3ccccc32)ccc1CCC(=O)O.O=C(OCc1ccccc1)N1CCC(CO)CC1 > >
O.Cc1cc(C(=O)N2Cc3cnn(C)c3N(CC3CCN(C(=O)OCc4ccccc4)CC3)c3ccccc32)ccc1CCC(=O)O



Output C ($\lambda = 1$) CRB with CBS:

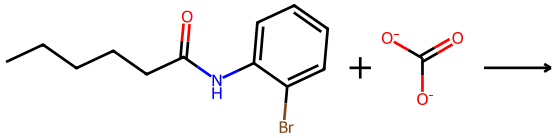
Cc1cc(C(=O)N2Cc3cnn(C)c3Nc3ccccc32)ccc1CCC(=O)O.O=C(OCc1ccccc1)N1CCC(CO)CC1 > >
O.Cc1cc(C(=O)N2Cc3cnn(C)c3N(CC3CCN(C(=O)OCc4ccccc4)CC3)c3ccccc32)oc1CCC(=O)CC1

Invalid

(g) Input, ground truth and output SMILES and reaction from the partialized validation set for which the constrained ($\lambda = 1$) Chemical Reaction Balancer without constrained beam search produced a balanced output and with constrained beam search produced an invalid output. The output of the model with constrained beam search contains an invalid molecule, so it cannot be plotted.

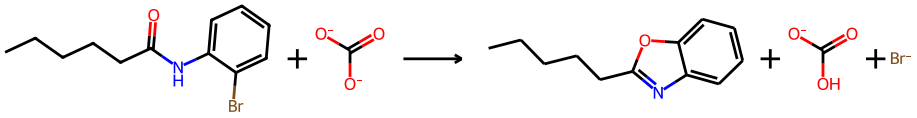
Input reaction:

CCCCCC(=O)Nc1ccccc1Br.O=C([O-])[O-] > >



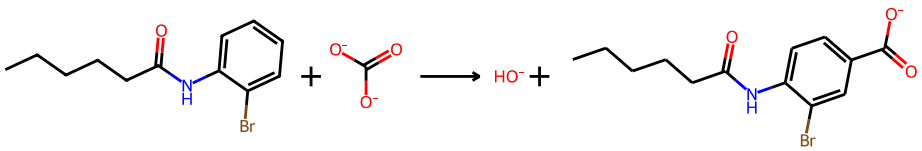
Ground truth:

CCCCCC(=O)Nc1ccccc1Br.O=C([O-])[O-] > > CCCCCc1nc2ccccc2o1.O=C([O-])O.[Br-]



Output C ($\lambda = 1$) CRB:

CCCCCC(=O)Nc1ccccc1Br.O=C([O-])[O-] > > [OH-].CCCCCC(=O)Nc1ccc(C(=O)[O-])cc1Br



Output C ($\lambda = 1$) CRB with CBS:

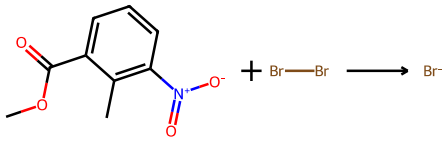
CCCCCC(=O)Nc1ccccc1Br.O=C([O-])[O-] > > [OH-].CCCCCC(=O)Nc1ccc(C(=O)[O-])Br)cc1

Invalid

(h) Input, ground truth and output SMILES and reaction from the partialized validation set for which the constrained ($\lambda = 1$) Chemical Reaction Balancer without constrained beam search produced a balanced output and with constrained beam search produced an invalid output. The output of the model with constrained beam search contains an invalid molecule, so it cannot be plotted.

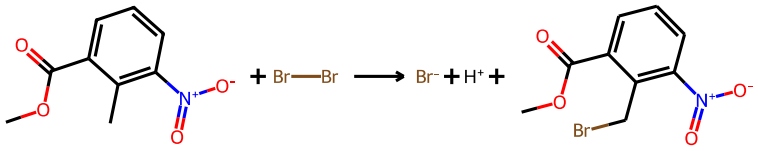
Input reaction:

COC(=O)c1cccc([N+](=O)[O-])c1C.BrBr > > [Br-]



Output C ($\lambda = 1$) CRB and ground truth:

COC(=O)c1cccc([N+](=O)[O-])c1C.BrBr > > [Br-].[H+].COC(=O)c1cccc([N+](=O)[O-])c1CBr



Output C ($\lambda = 1$) CRB with CBS:

COC(=O)c1cccc([N+](=O)[O-])c1C.BrBr > > [Br-].[H+].COC(=O)c1cccc([N+](=O)[O-])Br)c1C

Invalid

(i) Input and output SMILES and reaction from the partialized validation set for which the constrained ($\lambda = 1$) Chemical Reaction Balancer without constrained beam search produced a balanced output and with constrained beam search produced an invalid output. The output of the model without constrained beam search matches the ground truth. The output of the model with constrained beam search contains an invalid molecule, so it cannot be plotted.

Fig. S11. All input, ground truth and output SMILES and reactions from the partialized validation set for which the constrained ($\lambda = 1$) Chemical Reaction Balancer without constrained beam search produced a balanced output and with constrained beam search produced an imbalanced (a, b, c, d, e, f) or invalid (g, h, i) output.

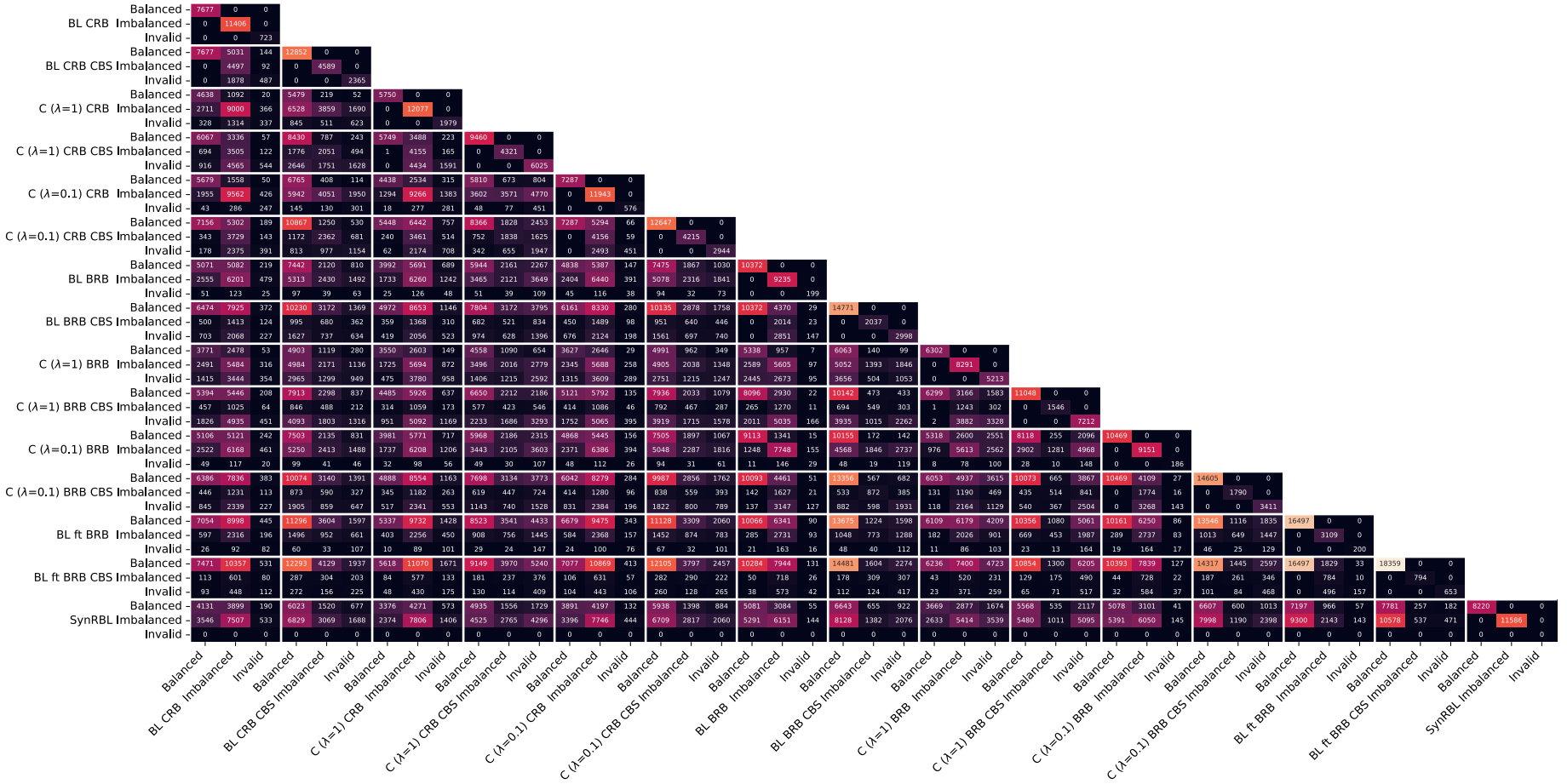
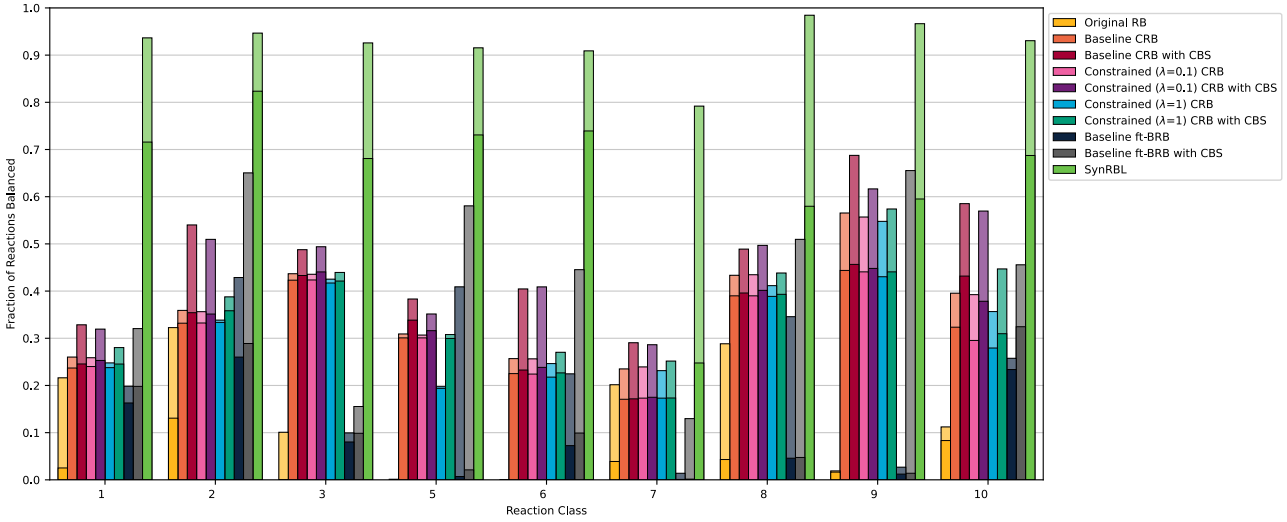


Fig. S12. Overlap in the performance of all trained Reaction Balancer models with and without constrained beam search and of SynRBL on the partialized Rhea test set categorized into balanced, imbalanced and invalid predictions. Each matrix of 3×3 outlined by white borders contains the overlap between any two models adding up to the total of all 19806 reactions. The matrices on the main diagonal are the categories within each method.

A



B

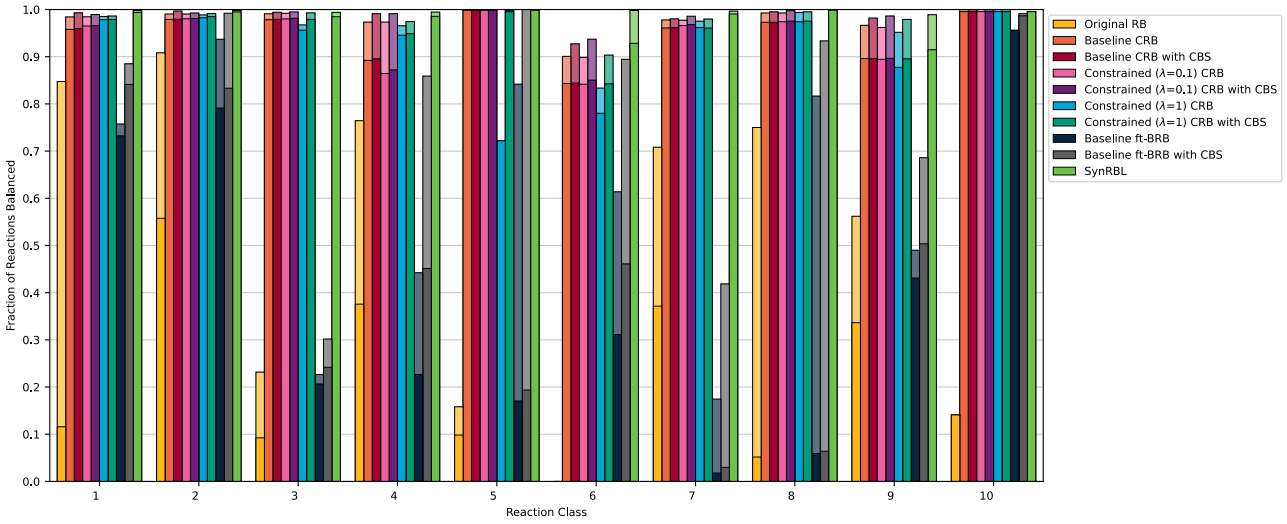


Fig. S13. Balancing performance of the Chemical Reaction Balancer and fine-tuned Biochemical Reaction Balancer models with and without constrained beam search and of the Original Reaction Balancer and SynRBL on USPTO 50k data of 2015 (**A**) and 2016 (**B**) per reaction class. The full bars show the total fraction of reactions that are balanced by each method within each reaction class. The brighter bars within each bar indicate the SynRBL-accuracy, which is the fraction of reactions that exactly match the predictions made by SynRBL with a confidence of at least 90% for the MCS-based method. Note that this measure can never exceed the SynRBL-accuracy of SynRBL itself.

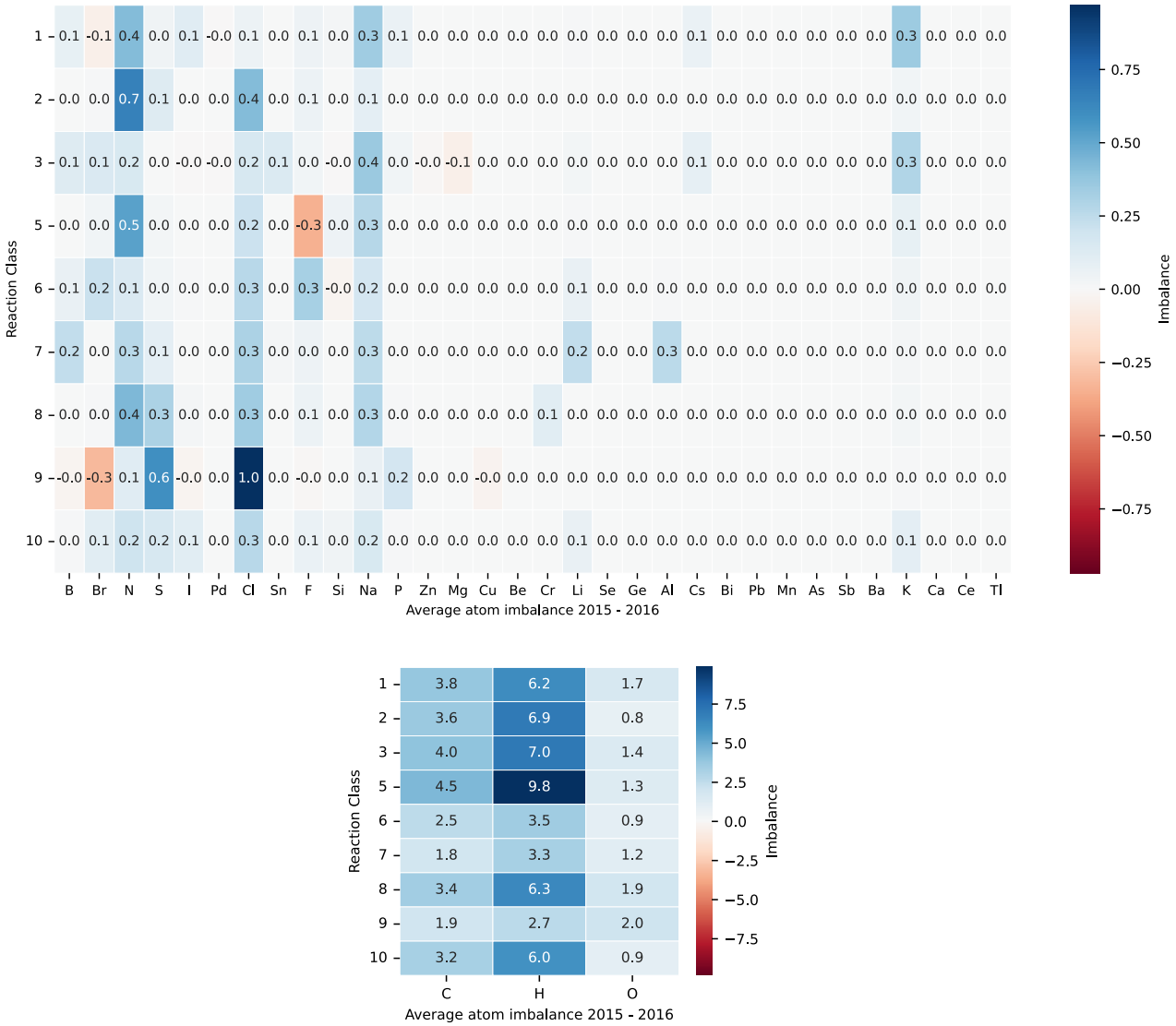


Fig. S14. Difference in absolute average atom imbalances between the USPTO 50k datasets for 2015 and 2016, categorized by reaction class. For each atom type present in either dataset, the absolute average atom imbalance is calculated per reaction class. This is done by first determining the absolute atom imbalance for each reaction and then averaging these values. If an atom type is absent in a reaction, its balance is considered zero. The figure displays the difference in absolute average atom imbalances, calculated as the value for the 2016 dataset subtracted from that of the 2015 dataset. Positive values (in blue) indicate that atom types are, on average, more imbalanced in the 2015 dataset for that reaction class, while negative values (in red) indicate greater imbalance in the 2016 dataset. Values are reported separately for carbon, hydrogen, and oxygen due to their larger scale ranges.