Abstract—While showing impressive performance on various kinds of learning tasks, it is yet unclear whether deep learning models have the ability to robustly tackle reasoning tasks. Measuring the robustness of reasoning in modern machine learning models such as Transformers is challenging as one needs to provide a task that cannot be easily shortcut by exploiting spurious statistical correlations in the data, while operating on complex objects and constraints. To address this issue, we propose CHEMALGEBRA, a benchmark for measuring the reasoning capabilities of Transformer models through the prediction of stoichiometrically-balanced chemical reactions. CHEMALGEBRA requires manipulating sets of complex discrete objects – molecules represented as formulas or graphs – under algebraic constraints such as the mass preservation principle. We believe that CHEMALGEBRA can serve as a useful test bed for the next generation of machine reasoning models and as a promoter of their development.

I. INTRODUCTION

Deep learning models, and Transformer architectures in particular, currently achieve the state-of-the-art for a number of application domains such as natural language and audio processing, computer vision, and computational chemistry [1]–[3]. Given enough data and enough parameters to fit, these models are able to learn intricate correlations [4].

Reasoning is the ability to manipulate a knowledge representation into a form that is more suitable to solve a new problem [5], [6]. In particular, algebraic reasoning includes a set of reasoning manipulations such as abstraction, arithmetic operations, and systematic composition over complex objects. Algebraic reasoning is related to the ability of a learning system to perform systematic generalization [7]–[9], i.e. to robustly make predictions beyond the data distribution it has been trained on. This is inherently more challenging than discovering correlations from data, as it requires the learning system to actually capture the true underlying mechanism for the specific task [10], [11].

Lately, much attention has been put on training Transformers to learn how to reason [12]–[15]. This is usually done by embedding an algebraic reasoning problem in a natural language formulation. Natural language, despite its flexibility, is imprecise and prone to shortcuts [16]. As a result, it is often difficult to determine whether the models’ performance on reasoning tasks is genuine or it is merely due to the exploitation of spurious statistical correlations in the data. Several works in this direction suggest [12], [17], [18] the latter is probably the case.

In order to effectively assess the reasoning capabilities of Transformers, we need to accurately design tasks that i) operate on complex objects ii) require algebraic reasoning to be carried out and iii) cannot be shortcut by exploiting latent correlations in the data. We identify chemical reaction prediction as a suitable candidate for these desiderata. First, chemical reactions can be naturally interpreted as transformations over bags of complex objects: reactant molecules are turned into product molecules by manipulating their graph structures while abiding certain constraints such as the law of mass conservation. Second, these transformations can be analysed as algebraic operations over (sub-)graphs (e.g., by observing bonds forming and dissolving [19]), and balancing them to preserve mass conservation can be formalised as solving a linear system of equations, as we will show in Section II. Third, the language of chemical molecules and reactions is much less ambiguous than natural language and by controlling the stoichiometric coefficients, i.e., the molecule multiplicities, at training and test time we can more precisely measure systematic generalization. Lastly, Transformers already excel at learning reaction predictions [20], [21]. Therefore, we think this can be a solid test bed to measure the current gap between learning and reasoning capabilities of modern deep learning models.

The main contributions of this paper are the following:

1) We cast chemical reaction prediction as a reasoning task where the learner has not only to predict a set of products but also correct stoichiometric coefficients variations (Section II).

2) We evaluate the current state-of-the-art Transformers for chemical reaction predictions, showing that they fail to robustly generalise when reasoning on simple variants

1An extended overview of the related works in chemical reaction prediction, as well as the dataset files, code, and additional experimental details and results, are given in the supplementary material (https://anonymous.4open.science/r/ChemAlgebra-7DDE).
of the chemical reaction dataset they have been trained on (Section III).

3) We introduce ChemAlgebra as a novel challenging benchmark for machine reasoning, in which we can more precisely measure the ability of deep learning models to algebraically reason over bags of graphs in in-, cross- and out-of-distribution settings (Section IV).

II. PREDICTING CHEMICAL REACTIONS AS ALGEBRAIC REASONING

To illustrate our point, let us consider the Sabatier reaction: it yields methane (CH\textsubscript{4}) and water (H\textsubscript{2}O) out of hydrogen (H\textsubscript{2}) and carbon dioxide (CO\textsubscript{2}), in the presence of nickel (Ni) as a catalyst. In chemical formulas:

\[
\text{CO}_2 + 4\text{H}_2 \xrightarrow{\text{Ni}} \text{CH}_4 + 2\text{H}_2\text{O}
\]  

(1)

where formulas encode complex graph structures where atoms are nodes and chemical bonds edges:

A reaction prediction learning task hence consists of outputting a bag of graphs, the products (right hand side), given the bag of graphs consisting of reactants (left hand side) and reagents (i.e., the catalysts, over the reaction’s arrow).

The multiplicities of the molecules, also called their stoichiometric coefficients, express the fractional proportions of reactants to yield a certain proportion of products. For example, one needs a 4:1 ratio of hydrogen molecules and carbon dioxide to produce a 1:2 ratio of methane and water. A reaction is (mass) balanced when its stoichiometric coefficients are well placed such that the sum of the number of atoms for every element across products shall be the same of that across reactants, i.e., it satisfies the principle of mass conservation [22]. Unbalanced reactions, on the other hand, would be chemically implausible.

This constraint over atoms of the molecules underpins the true chemical mechanism behind reactions: bonds between atoms break and form under certain conditions but atoms do not change. In reasoning terms, this is a symbol-manipulating process where bags of graphs are deconstructed into other bags of graphs. A machine reasoning system that would have learned this true chemical mechanism, would be able to perfectly solve the chemical reaction prediction task for all balanced reactions and for all possible variations of stoichiometric coefficients.

As humans, we can balance fairly complex chemical reactions quite easily.\textsuperscript{2} For machines, this process can be formalised as finding a solution of a potentially underdetermined system of linear equations. For example, we can write the Sabatier reaction as:

\[
r_1 \cdot \text{CO}_2 + r_2 \cdot \text{H}_2 + r_3 \cdot \text{Ni} = p_1 \cdot \text{CH}_4 + p_2 \cdot \text{H}_2\text{O} + p_3 \cdot \text{Ni}
\]

(2)

where the variables \(r_1, r_2, r_3, p_1, p_2, p_3\) represent the stoichiometric coefficients of the molecules they refer to. Molecules of reagents act as a special kind of confounders: since they are not changed during the reaction, they must appear on both sides of the equation with the same coefficient (i.e., \(r_3 = p_3\)). Under this rewriting, it becomes even more evident how a full chemical reaction can be interpreted as an algebraic equation where stoichiometric coefficients are the unknown variables.

Then, we can represent the molecule of CO\textsubscript{2} with the vector \([1, 0, 2, 0]\), indicating one atom of carbon, zero hydrogens, two oxygens, and zero nickles. Analogously, H\textsubscript{2} can be encoded as \([0, 2, 0, 0]\), and so on. Therefore, Eq. (1) can be encoded as the following linear system:

\[
\begin{bmatrix}
1 & 0 & 0 \\
0 & 2 & 0 \\
2 & 0 & 0
\end{bmatrix}
\begin{bmatrix}
r_1 \\
p_1 \\
p_3
\end{bmatrix}
=
\begin{bmatrix}
1 & 0 & 0 \\
4 & 2 & 0 \\
0 & 1 & 0
\end{bmatrix}
\begin{bmatrix}
p_2 \\
p_4 \\
0
\end{bmatrix}
\]

(3)

It is straightforward to verify that the minimum norm solution of Eq. (3) is \(r = [1, 4, 1]\), \(p = [1, 2, 1]\), thus conforming to Eq. (1). We now devise a set of reasoning tasks exploiting this perspective.

A. “Type 1” and “Type 2” chemical reaction reasoning tasks

We propose to cast chemical reaction prediction as a reasoning task, where the model has to predict both the graphs corresponding to the product molecules (i.e. the right hand side of the reaction) and the exact multiplicities of such molecules, given a particular input consisting of reagents and reactants equipped with varying stoichiometric coefficients. This is in stark contrast with the vanilla reaction prediction learning setting: in it, models are trained on reactions without stoichiometric information, largely on unbalanced reactions (see Section III-B); in our setting, stoichiometric coefficients are not only present but, at prediction time, can greatly differ from those the model has seen at training time. By doing so, we can better control and measure systematic generalization.

For example, given a reference reaction, we can multiply all its stoichiometric coefficients by a factor. By multiplying coefficients in Sabatier reaction by a factor of two, we would obtain the following input-output pair:

\[
2\text{CO}_2 + 8\text{H}_2 + 2\text{Ni} \quad \text{(INPUT)} \quad 2\text{CH}_4 + 4\text{H}_2\text{O} + 2\text{Ni} \quad \text{(OUTPUT)}.
\]

(4)

In the rest of this paper, we will refer to this type of reasoning task as Type 1 task. Alternatively, we can just add a certain number of molecules on the left hand side. Some of these additional molecules might not take part in the reaction, since there might not be enough reactants to bond with. For example, if we add two CO\textsubscript{2} and two Ni to Sabatier reaction

\textsuperscript{2}We learn to do it from very few examples, e.g. a handful of reactions taken from chemistry textbooks in high school. Without following an explicit algorithm, we can usually perform balancing in an intuitive way, by leveraging our quick arithmetic skills to count the atoms for an element and match the numbers on both sides of the equation, iteratively changing the stoichiometric coefficients until all elements are balanced.
we expect a model to predict them in the right hand side in addition to its usual outputs:

\[
3 \text{CO}_2 + 4 \text{H}_2 + 3 \text{Ni} \quad \text{(INPUT)} \\
\text{CH}_4 + 2 \text{H}_2\text{O} + 3 \text{Ni} + 2 \text{CO}_2 \quad \text{(OUTPUT)}.
\] (5)

We refer to this as a Type 2 task. Type 2 reactions are harder to reason with than Type 1, but should still be easy for a machine learning model that has learned the true underlying chemical mechanism. There is one final aspect to consider: if the test coefficients are sampled from the same data distribution as the training coefficients, there is still a chance that the model would be able to predict them just by pattern matching. Conversely, learning the actual algebraic reasoning behind the stoichiometry of chemical reactions, e.g., by solving the linear system of Eq. (3), empowers the model to solve any stoichiometry problem, regardless of the actual values of the coefficients observed during training.

To control for these aspects, in addition to the usual in-distribution scenario where both training and test coefficients are sampled from the same set of integer numbers \(S_{in}\), we consider also an out-of-distribution setting where the training and test reactions are instantiated with coefficients coming from the disjoint sets of integers \(S_{in}\) and \(S_{out}\), respectively. Analogously, we can test a cross-distribution scenario, where the training set is divided into two halves. The reactions of the first half are instantiated with coefficients selected from \(S_{in}\), while the second half’s coefficients are selected from \(S_{out}\). The cross-distribution test set will contain the same reactions of the training set but with “swapped stoichiometry”: that is, the first half of the test set will have coefficients from \(S_{out}\), the second half from \(S_{in}\). We argue that a well-trained reasoning model should have no problem to semantically disentangle the stoichiometric coefficients from the molecules, thus achieving the same performance in both the in-distribution, cross-distribution, and out-of-distribution scenarios.

III. CAN TRANSFORMERS PERFORM ALGEBRAIC REASONING?

In this section, we question whether current deep learning models can effectively solve the algebraic reasoning tasks induced by the stoichiometry of chemical reactions, as discussed in Section II. In order to do so, we first need to build a suitable dataset of chemical reactions that can be processed by state-of-the-art models for reaction predictions. We then evaluate these models on some meaningful variations of the task and discuss their limitations.

A. How to build a benchmark of balanced reactions

A natural candidate is the USPTO-MIT dataset [23], a large collection of chemical reactions extracted from US patents from 1976 to 2016 and represented in the SMILES format [24]. SMILES reactions are text strings encoding a linearization of the molecular graphs of each molecule participating in the reaction.\(^3\) We employ the version popularized by [25], often referred to as USPTO-MIT. This version contains a subset of polished reactions, obtained by removing duplicate and incorrect ones. For simplicity, in the rest of this paper we will refer to “USPTO-MIT” as simply “USPTO”. This dataset is composed of 409k training reactions, 30k validation reactions and 40k test reactions (see Table I).

Unfortunately, reactions in the USPTO dataset are mostly unbalanced: we found that less than 4.6% of the reactions in the dataset are mass balanced and readily usable for our tasks. For the remaining reactions, in many cases only the major product of the reaction was recorded, while disregarding minor byproducts, such as \(\text{H}_2\text{O}\) or \(\text{HCl}\), probably not deemed interesting from a patent perspective. Examples of these reactions are illustrated in Fig. 1. While this does not impact patents, it can affect the generalization capability of the learned models, as we will show next.

Fortunately, the missing byproducts can be sometimes deduced from the unbalanced reaction: if the set of missing atoms of one side of the reaction is enough to unambiguously reconstruct one (or more) valid molecules on the other side, we add such reconstructed molecules to that side of the reaction. The rightmost column in Table I shows the number of reactions contained in the balanced subset of USPTO that we used as the basis for the augmentation procedures described in Section IV. Using this strategy, we were able to re-balance about 44% of the original reactions for the training, validation and test sets. We denote this re-balanced version of USPTO as USPTO-BAL and use it next.

B. Type 1 and Type 2 variants of USPTO

We build two variants of the original USPTO dataset, called USPTO-T1 and USPTO-T2, as instances of the Type 1 and Type 2 reasoning tasks introduced in Section II-A. For T1, we multiply all coefficients by two, in practice duplicating the molecule representations in the SMILES string of a reaction.

\(^3\)Note that SMILES strings represent hydrogen implicitly, when the resulting molecule is not ambiguous.
TABLE I
Statistics of the training, validation and test splits of the USPTO dataset before and after our rebalancing. Percentages in parenthesis are w.r.t. the original dataset.

|                     | Total | Balanced (%) | Re-balanced (%) | USPTO-BAL (%) |
|---------------------|-------|--------------|-----------------|---------------|
| Training set        | 409035| 18815 (4.59) | 162220 (39.66)  | 181035 (44.26) |
| Validation set      | 30000 | 1292 (4.31)  | 11686 (39.56)   | 13160 (43.87)  |
| Test set            | 40000 | 1809 (4.52)  | 15973 (39.93)   | 17782 (44.45)  |

(as the SMILES format does not allow to represent stoichiometric coefficients). For T2 we add a randomly-selected molecule to the reactants. The detailed procedure is discussed in the supplementary material.

These two variants should not, in theory, pose significant challenges for chemical reaction models that learned the true reaction mechanisms of USPTO, since despite shifting from the original distribution, they still use the same molecules and reaction mechanisms of USPTO.

In the case of USPTO-T1, the reactants contain all the required additional molecules to perform the reaction twice, so the output should correspond to the same multiple of the original products. This is even easier considering that the original reactions in USPTO generally involve a single molecule per type. On the other hand, in the case of USPTO-T2, the added molecule is not sufficient to trigger multiple reactions, so it should just be copied in output by the model. As we will show in the next section, in practice this is not the case.

C. Current chemical models are actually alchemical models

We now evaluate the systematic generalization of state-of-the-art models for reaction prediction trained on USPTO on the chemically-sound variation and the reasoning variants introduced in Section III-A and Section III-B. We focus on Transformer-based language models that operate on SMILES representations [21], [26] as well Transformers that employ graph neural networks (GNNs) to parse the molecules structures and be permutation invariant [27]. For additional details about these models, we refer the reader to the supplementary material.

For all models, we report the top-1 prediction accuracy (ACC) on the BAL, T1 and T2 dataset variants as well as the original USPTO. On the T1 and T2 variants, a correct prediction would involve the model output both the right molecular graph and the right multiplicity of each product molecule. This proved extremely challenging for all models. As such, we also report the “at-least-one” accuracy (ALO) score which consider the prediction correct if at least one of the ground truth molecules is being predicted, despite its multiplicity being inaccurate.

The results in Table II are striking: while all models are able to reach over 90% of top-1 accuracy over the standard unbalanced USPTO test set, the performance drops sharply below 2% on USPTO-T1 and USPTO-T2 but also on USPTO-BAL, despite this variant only adds a small number of byproduct molecules. This fall in performance is also evident for the ALO metric. The distribution shifts in the original training distribution introduced in our variants highlight that these state-of-the-art models are not learning the true chemical mechanism, but a brittle function approximation that leverages statistical correlations present in the data. This is somehow expected as these models have been trained on unbalanced reactions and their predicted products often do not satisfy the mass preservation constraint. In fact, predicted product molecules might contain not only an unbalanced number of atoms per element, but also introduce atoms of elements that were not present in the reactants and reagents or making other elements disappear from the products. We show some qualitative examples of these “alchemical” predictions reported on the original USPTO dataset,4 in Fig. 2.

4In nuclear reactions the appearance or disappearance of elements would be allowed, but this kind of reactions is out of the scope of USPTO and our work.
where each coefficient is encoded as an integer surrounded stoichiometrically-augmented SMILES string needed for a challenging benchmark). To this end, we define long molecules and large coefficients numbers, which are needed for a challenging benchmark. Starting from the balanced variant of USPTO, described in Section III (USPTO-BAL), and overcome these challenges in the next section.

We devise eight tasks for assessing the reasoning capabilities of deep learning models. First of all, we need to find a way to systematically train and evaluate models over several variations of these two. Designing a suitable benchmark suite to systematically train and evaluate models over several variations of these manipulations poses certain challenges in terms of input specification and controlling potential shortcutting. We discuss and overcome these challenges in the next section.

IV. THE CHEM ALGEBRA BENCHMARK

Building on the previous considerations, we now design the CHEM ALGEBRA benchmark. Starting from the balanced variant of USPTO, described in Section III (USPTO-BAL), we devise eight tasks for assessing the reasoning capabilities of deep learning models. First of all, we need to find a way to explicitly represent the stoichiometric coefficients directly (as duplicating SMILES sub-strings becomes impractical for long molecules and large coefficients numbers, which are needed for a challenging benchmark). To this end, we define a stoichiometrically-augmented SMILES string language, where each coefficient is encoded as an integer surrounded by curly braces preceding each molecule in the reaction. For example, Sabatier reaction from Eq. (1) becomes

\[ \{\text{H}_2\} \rightarrow \{\text{Ni}\} \]  \hspace{1cm} (6)

in this new SMILES dialect. To better understand how the reasoning task of predicting the correct reaction mechanisms affects the performance, we devise a second, simpler, representation where molecules in the reactions are encoded as raw chemical formulas. Sabatier reaction encoded in this way is:

\[ \{\text{CO}_2\} \cdot \{\text{H}_2\} \cdot \{\text{Ni}\} \]  \hspace{1cm} (7)

Note that in both cases the reagent Ni is copied explicitly on both sides of the reaction. These alternative notations in CHEM ALGEBRA pose different challenges for deep learning models. The SMILES representation is more verbose and lower-level, allowing for a complete reconstruction of the original molecular structure. However, SMILES strings can be ambiguous [28], with more than one string representing the same molecule.\(^5\) They also do not explicitly represent Hydrogen atoms, so the models would need to extract that implicit information by its own and properly balance it. The formula notation, on the other hand, is more compact and more abstract. It represents all the atoms, and their multiplicities, in an explicit way, but the correct reaction mechanism is harder to predict, as the formula does not contain structural information.

We can now instantiate the two reasoning tasks defined in Section II-A with the new notation for stoichiometric coefficients in CHEM ALGEBRA. For Type 1 reactions, we randomly select a single multiplication factor from a set and apply it to all molecules in the reaction. Since the reactions in USPTO-BAL are balanced, but rarely contain molecules with coefficients greater than one, Type 1 reactions will likely contain molecules sharing the same random stoichiometric coefficient. We employ this variant to precisely measure how much the learning model can shortcut the reaction reasoning task by just predicting the same coefficient of the input reagents for the output products.

Instead, for Type 2 reactions, each molecule in the reaction reagents, reactants and products is randomly assigned a different coefficient and allowed to appear on the left or right side in addition to its original placement, thus creating confounder molecules that might not react. This process requires taking some precautions in order to ensure that the resulting reaction is still balanced. We detail the complete procedure to create Type 2 reactions in the supplementary material. As this would make it difficult for a model to shortcut coefficient prediction,

\(^5\)We use the canonized molecular string representation [28].
Type 2 reactions pose a greater challenge than Type 1. Table III illustrates examples of Type 1 and 2 derivations from a reference reaction, encoded in stoichiometric SMILES and raw formula representations.

We now have to decide how many times shall we create Type 1 and 2 variants of a reaction from USPTO-BAL at training time. We control this factor of variation by devising two augmentation strategies to measure how different training sizes could impact generalization:

**x1 augmentations**
Each reaction of USPTO-BAL is used to create a single stoichiometric reaction in CHEMAlgebra.

**x5 augmentations**
Each reaction of USPTO-BAL is used to create five stoichiometric reactions in CHEMAlgebra. All the stoichiometric reactions contain the same input/output molecules, but differ by the sampled values of the stoichiometric coefficients.

This leaves us with a total of 8 datasets variants, one for each combination of the 2 reasoning types, 2 molecule encodings and 2 augmentation amounts. A comprehensive overview of the eight CHEMAlgebra datasets is included in the supplementary material. Lastly, we need to determine the different sets of coefficients to sample from across train and test distributions. We thus instantiate the chemical reactions in CHEMAlgebra by sampling the stoichiometric coefficients in the interval $S_{in} = [1, 5]$. We retain the same train, validation and test split as the original USPTO-BAL dataset. As outlined in Section II-A, in order to test systematic generalisation, we build three different test sets for each dataset variant:

**In-distribution**
The test reactions are instantiated with stoichiometric coefficients drawn in the interval $S_{in}$. This is the “easy” scenario where the model should be expected to use the shortcuts for Type 1 reasoning.

**Out-of-distribution**
The test reactions are instantiated with stoichiometric coefficients drawn in the $S_{out} = [6, 10]$ interval. This is the “hard” prediction task where to succeed a model would also require the arithmetic knowledge to derive coefficients in $S_{out}$ from those in $S_{in}$. This challenging setting is introduced explicitly to stimulate future research to inject external knowledge or ways to bind symbols to concepts.

**Cross-distribution**
Half of training and test reactions is instantiated with stoichiometric coefficients in $S_{in}$, the other half in $S_{out}$. This is a “medium” difficulty scenario where the model can infer arithmetic knowledge from the training data and still shortcut predictions for Type 1 reactions.

### A. Baseline experiments

As part of our benchmark, we provide reference baseline results for the eight variations of CHEMAlgebra described in Section IV. We employ a Transformer architecture\(^6\) akin to the state-of-the-art models of Table II (see the supplementary material for the details about the architecture and the training procedure). Our aim with these baselines is to analyse the performance variations with respect to different molecule representations, coefficient assignment types, and dataset sizes. We measure the performance of the model as in a multilabel classification, where the task is to predict not only the exact bag of product molecules but also their stoichiometric coefficients. To do so, we consider the bag of labels to be the bag of product molecules, in which a molecule appears as many times as its correct coefficient states. We therefore employ classical multi-label metrics [30] for the top-1 generated bags of products. We use the exact match (EM) score, the multi-label accuracy, also called Jaccard (JAC) score, and the F1 score metrics (details in supplementary material). These metrics are differently forgiving, with the EM score being the most strict and F1 the most permissive.

\(^6\)We chose to focus on the Transformer for our baseline because i) the models with the highest accuracy on forward reaction prediction are all based on Transformers, ii) Transformers are already currently used for many types of reasoning tasks [12] and iii) while other specialized approaches, such as MEGAN [29], can bake simple chemical rules into the model by design, they cannot be used to solve different kinds of reasoning tasks.
The results in Table IV show some interesting trends. First, as expected, EM scores drop sharply on the out-of-distribution setting. Not only the model is not able to recover the exact multiplicity of product molecules, but also their presence, as the F1 and JAC scores drop to less than half the in-distribution scores. Type 2 tasks are confirmed to be much harder than Type 1, as highlighted especially by EM scores. Second, the baseline can actually achieve better out-of-distribution JAC and F1 scores Type 2 than Type 1 tasks. This can be explained by considering that JAC and F1 scores are more permissive than EM and can increase if the prediction contains more instances of one molecule as labels, even if the complete configuration does not exactly match. Third, augmenting the training data fivefold (x5) does not yield any breakthrough performance improvement. On the contrary, for Type 1 reactions, it seems that additional augmentations can slightly confuse the model on which shortcut to use. Overall, this suggests that the CHEMAGLERA tasks are challenging and robust to the usual data augmentation techniques that are usually used to improve the performance of Transformer-based models [21], [26]. We recommend running CHEMAGLERA x1 if one needs to save computation. Finally, balancing raw chemical formulas is reported to be easier than doing that on SMILES. It seems that predicting the stoichiometric coefficients is easier when using formulas, while even Type 1 in-distribution reaction predictions are challenging when using graph structures, encoded as SMILES, as inputs.

V. CONCLUSION

Our capacity to measure the reasoning capabilities for current machine learning models is bounded by the availability of benchmark tasks. Many datasets in use for this purpose cast reasoning as a subroutine of more complex problems, often expressed in natural language [12], [31]. By doing so, it becomes hard to disentangle the systematic generalization ability of a model from a number of confounding factors such as mere statistical correlations in the data [32], [33]. We introduced CHEMAGLERA as a solid benchmark for robustly measuring the reasoning abilities of deep learning models over complex objects, such as bags of graphs. CHEMAGLERA offers a controlled and versatile experimental setting — including different reasoning tasks and graph encodings — while remaining highly challenging in both the in- and out-of-distribution settings. In fact, it requires to combine statistical learning over graphs with algebraic reasoning over the (sub-)graph structures and their multiplicity. As such, we think that the CHEMAGLERA task can serve as a useful test bed for the upcoming generation of machine reasoning models.

A. Limitations and Future Directions

While completely disentangling learning and reasoning might not be possible (as it is not in many real-world applications), we stress that the intended use of CHEMAGLERA is mainly for the evaluation of the reasoning abilities of neural models. Therefore, it should not be considered to be a dataset to train real-world models for pure chemical reaction prediction. We remark that neural reaction prediction models injecting prior knowledge about chemical reactions (e.g. MEGAN) might have a more favourable inductive bias to solve the reasoning task. Our focus is however on Transformers, which are emerging as general-purpose learning and reasoning machines. We believe CHEMAGLERA can be a solid benchmark to measure their capabilities and offer a more controlled environment where shortcutting can be limited. We recognize the limitations of our re-balancing technique, which is a simple heuristic and may abstracts-away chemical nuances relevant for pure chemical reaction predictions. At the same time, we believe that can it be a simple initial solution to propel more rigorous and challenging algebraic reasoning benchmarks. Type 2 augmentation, while sound from an algebraic point of view, might also introduce noise in the chemical reaction prediction task.

We plan to design additional variations of CHEMAGLERA to take into account the different facets of machine and human reasoning as well as more chemically and physically plausible constraints such as modeling redox reactions.

ACKNOWLEDGMENTS

We wish to thank Adrià'n Javaloy and Frank Mollica for their precious discussions and suggestions, which were crucial to improve the quality of the final manuscript. We would also like to thank iGenius for supporting Andrea Valenti with conference registration fees, travel and accommodation expenses. This work has been supported by H2020 TAILOR (Grant

| TABLE IV | EXACT MATCH (EM), JACCARD (JAC) AND F1 SCORES EVALUATING THE PREDICTION OF BOTH MOLECULES AND COEFFICIENTS ON THE 8 VARIATIONS OF CHEMAGLERA FOR IN-, CROSS- AND OUT-OF-DISTRIBUTION GENERALIZATION (METRICS REPORTED AS PERCENTAGES). |
|-----------------|-------------------|-------------------|-------------------|-------------------|-------------------|-------------------|-------------------|
|                | IN                |                  | CROSS             |                  | OUT               |                  |
| TYPE            | ACG.              | EM    | JAC  | F1    | EM    | JAC  | F1    | EM    | JAC  | F1    | EM    | JAC  | F1    | EM    | JAC  | F1    |
| FORMULA         |                   |       |      |       |       |      |       |       |      |       |       |      |       |       |      |       |       |
| T1              | x1                | 29.99 | 63.03 | 72.42 | 37.81 | 67.40 | 75.72 | 0.36  | 3.89  | 6.48  |
|                 | x5                | 28.87 | 62.10 | 71.69 | 25.85 | 60.64 | 70.59 | 0.38  | 3.70  | 12.15 |
| T2              | x1                | 1.10  | 39.87 | 53.53 | 0.03  | 31.43 | 44.84 | 0.00  | 13.71 | 23.05 |
|                 | x5                | 1.30  | 41.01 | 54.71 | 0.06  | 32.57 | 45.97 | 0.08  | 1.87  | 3.38  |
| SMILES          |                   |       |      |       |       |      |       |       |      |       |       |      |       |       |      |       |       |
| T1              | x1                | 4.17  | 44.43 | 57.49 | 5.96  | 45.74 | 58.59 | 0.04  | 1.88  | 3.18  |
|                 | x5                | 2.62  | 41.16 | 54.99 | 2.77  | 42.01 | 55.18 | 0.08  | 1.87  | 3.38  |
| T2              | x1                | 0.03  | 23.37 | 35.37 | <10−2 | 22.73 | 34.57 | 0.00  | 8.90  | 15.61 |
|                 | x5                | 0.04  | 24.00 | 36.18 | <10−2 | 22.73 | 34.67 | 0.00  | 7.44  | 13.08 |

This work has been supported by H2020 TAILOR (Grant Agreement No. 870742).
REFERENCES

[1] T. Lin, Y. Wang, X. Liu, and X. Qiu, “A survey of transformers,” arXiv preprint arXiv:2106.04554, 2021.

[2] S. Khan, M. Naseer, M. Hayat, S. W. Zamir, F. S. Khan, and M. Shah, “Transformers in vision: A survey,” ACM Computing Surveys (CSUR), 2021.

[3] A. M. Brașoveanu and R. Andonie, “Visualizing transformers for nlp: a brief survey,” in 2020 24th International Conference Information Visualisation (IV), pp. 270–279, IEEE, 2020.

[4] T. Brown, B. Mann, N. Ryder, M. Subbiah, J. D. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, et al., “Language models are few-shot learners,” Advances in neural information processing systems, vol. 33, pp. 1877–1901, 2020.

[5] L. Bottou, “From machine learning to machine reasoning,” Machine learning, vol. 94, no. 2, pp. 133–149, 2014.

[6] A. d. Garcez, M. Gori, L. C. Lamb, L. Serafini, M. Spranger, and S. N. Tran, “Neural-symbolic computing: An effective methodology for principled integration of machine learning and reasoning,” arXiv preprint arXiv:1905.06085, 2019.

[7] G. F. Marcus, The algebraic mind: Integrating connectionism and cognitive science. MIT press, 2003.

[8] D. Bahdanau, M. Levy, Y. Bengio, J. Pennington, and S. Chopra, “Memory and attention in sequence transduction,” in Proceedings of the 33rd Conference on Neural Information Processing Systems, 2019.

[9] G. Tsoumakas and I. Katakis, “Multi-label classification: An overview,” International Journal of Data Warehousing and Mining (IJDWM), vol. 3, no. 3, pp. 1–13, 2007.

[10] J. Pearl, Causality. Cambridge university press, 2009.

[11] J. Pearl, “Causality. Cambridge university press, 2009.

[12] C. Helwe, C. Clavel, and F. M. Suchanek, “Reasoning with transformation-based models: Deep learning, but shallow reasoning,” in 3rd Conference on Automated Knowledge Base Construction, 2021.

[13] H. Al-Negheimish, P. Madhyastha, and A. Russo, “Numerical reasoning in machine reading comprehension tasks: are we there yet?,” arXiv preprint arXiv:2109.08207, 2021.

[14] S. Storks, Q. Gao, and J. Y. Chai, “Commonsense reasoning for natural language understanding: A survey of benchmarks, resources, and approaches,” arXiv preprint arXiv:1904.01172, pp. 1–60, 2019.

[15] N. Gontier, K. Sinha, S. Reddy, and C. Pal, “Measuring systematic generalization in neural proof generation with transformers,” in Advances in Neural Information Processing Systems (H. Larochelle, M. Ranzato, R. Hadsell, M. Balcan, and H. Lin, eds.), vol. 33, pp. 22231–22242, Curran Associates, Inc., 2020.

[16] R. Geirhos, J.-H. Jacobsen, C. Michaelis, R. Zemel, W. Brendel, M. Bethge, and F. A. Wichmann, “Shortcut learning in deep neural networks,” Nature Machine Intelligence, vol. 2, no. 11, pp. 665–673, 2020.

[17] A. Agrawal, A. Katsev, and E. Hovy, “Analyzing the behavior of visual question answering models,” arXiv preprint arXiv:1606.07356, 2016.

[18] R. Jia and P. Liang, “Adversarial examples for evaluating reading comprehension systems,” arXiv preprint arXiv:1904.01172, pp. 1–60, 2019.

[19] B. Su, V. Van Deursen, G. Godin, and A. Courville, “Systematic generalization: what is required and can it be learned?,” Nature communications, vol. 11, no. 1, pp. 1–13, 2020.

[20] A. Neelakantan, P. Shyam, G. Sastry, A. Askell, et al., “Language models are few-shot learners,” Advances in neural information processing systems, vol. 33, pp. 1877–1901, 2020.

[21] Z. Tu and C. W. Coley, “Permutation invariant graph-to-sequence model for template-free retrosynthesis and reaction prediction,” arXiv preprint arXiv:2110.09681, 2021.

[22] J. Aris-Pous, S. V. Johansson, O. Prykhodko, E. J. Bjerrum, C. Tyrväinen, J.-L. Reynold, H. Chen, and O. Engkvist, “Randomized smiles strings improve the quality of molecular generative models,” Journal of chemical information and computer sciences, vol. 50, no. 7, pp. 3273–3284, 2021.

[23] T. Brown, B. Mann, N. Ryder, M. Subbiah, J. D. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, et al., “Language models are few-shot learners,” Advances in neural information processing systems, vol. 33, pp. 1877–1901, 2020.

[24] G. Tsoumakas and I. Katakis, “Multi-label classification: An overview,” International Journal of Data Warehousing and Mining (IJDWM), vol. 3, no. 3, pp. 1–13, 2007.

[25] J. Wei, X. Wang, D. Schuurmans, M. Bosma, E. Chi, Q. Le, and D. Zhou, “Chain of thought prompting elicits reasoning in large language models,” arXiv preprint arXiv:2201.11903, 2022.

[26] H. Zhang, L. H. Li, T. Meng, K.-W. Chang, and G. V. d. Broeck, “On the paradox of learning to reason from data,” arXiv preprint arXiv:2205.11502, 2022.

[27] R. Branco, A. Branco, J. Rodrigues, and J. Silva, “Shortcutted commonsense: Data spuriousness in deep learning of commonsense reasoning,” in Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, pp. 1504–1521, 2021.

[28] R. Hadsell, M. Balcan, and H. Lin, eds.), vol. 33, pp. 22231–22242, Curran Associates, Inc., 2020.

[29] R. Branco, A. Branco, J. Rodrigues, and J. Silva, “Shortcutted commonsense: Data spuriousness in deep learning of commonsense reasoning,” in Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, pp. 1504–1521, 2021.

[30] H. Zhang, L. H. Li, T. Meng, K.-W. Chang, and G. V. d. Broeck, “On the paradox of learning to reason from data,” arXiv preprint arXiv:2205.11502, 2022.