Supplementary information for

Entropy-stabilized metal oxide nanoparticles supported on reduced graphene oxide as highly active heterogeneous catalyst for selective and solvent-free oxidation of toluene: A combined experimental and numerical investigation

Seyedsaeed Mehrabi-Kalajahi*[1,2], Ahmad Ostovari Moghaddam*[3], Fahimeh Hadavimoghaddam[4], Mikhail A. Varfolomeev*[1,2], Almaz L. Zinnatullin[5], Iskander Vakhitov[5], Kamil R. Minnebaev[6], Dmitrii A. Emelianov[2], Daniil Uchaev[3], Andreu Cabot[7,8], Il'dar R. Il'yasov[2], Rustem R. Davletshin[9], Evgeny Trofimov[3], Nailia M. Khasanova[6], Farit G. Vagizov[5]

[1] *Department of Petroleum Engineering, Kazan Federal University, Kazan 420008, Russia*

[2] *Department of Physical Chemistry, Kazan Federal University, Kazan 420008, Russia*

[3] *Department of Materials Science, Physical and Chemical Properties of Materials, South Ural State University, 76 Lenin Ave, Chelyabinsk, 454080, Russia*

[4] *Institute of Unconventional Oil & Gas, Northeast Petroleum University, Heilongjiang, Daqing, 163318, China*

[5] *Institute of Physics, Kazan Federal University, Kazan 420008, Russia*

[6] *Institute of Geology and Petroleum Technologies, Kazan Federal University, Kazan, 420008, Russia*

[7] *Catalonia Institute for Energy Research – IREC, 08930 Sant Adrià de Besòs, Spain*

[8] *ICREA, Pg. Lluís Companys 23, 08010 Barcelona, Spain*

[9] *Department of Organic Chemistry, Kazan Federal University, Kazan 420008, Russia*

*Corresponding authors

*(S.Mehrabi-Kalajahi) E-mail:ss.mehrabikalajahi@gmail.com, smehrabi.kaladzhahi@kpfu.ru

*(A. Ostovari Moghaddam) E-mail address: ostovarim@susu.ru, ostovary@aut.ac.ir

*(M. A. Varfolomeev) E-mail: vma.ksu@gmail.com, mikhail.varfolomeev@kpfu.ru

S1. Detailed description of the hybrid models

S1.1. K -Nearest Neighbour (KNN)

The K-nearest neighbour (KNN) is a non-parametric learning algorithm that is unique for analysis in segmentation studies [1]. Unlike parametric techniques, KNN does not assume the basic distribution of data. Non-parametric algorithms are better linked to real-world problemas where the information and data generally do not follow theoretical hypotheses. In addition, with KNN, there is no explicit training step, which means that the time necessary to train the model is much less. Unlike other machine learning techniques, calculations are based on a complete and wide data set. However, it is limited in terms of memory consumption and time. In the worst case, it may use all the data points to make a decision, or it may require a huge block of memory to store massive training data. Some of the assumptions that are commonly made while using KNN, are designed as follows:

S1.2. Adaptive boosting (AdaBoost)

Freund and Schapire for the first time suggested the AdaBoost system, which can assume the reweighted data for weak classifiers [2]. However, this system can reduce the focus of the inexperienced classifier [3]. The basic steps of the AdaBoost system are summarized as follows:

First, define the data of weights as: $w_j = \frac{1}{n}, j = 1,2,....,n$;

Second, in order to obtain the weighted error, the training data were settled to a weak classifier of $Wl_i(x)$ for each i as follows:

$$Err_i = \frac{\sum_{j=1}^{n} w_j I(t_j \neq wl_i(x))}{\sum_{j=1}^{n} w_j}, \quad I(x) = \begin{cases} 0 \; if \; x = false \\ 1 \; if \; x = true \end{cases};$$

Third, for every i, the weight determination is associated with each predictor as:

$$\beta_i = log\left(\frac{(1 - Err_i)}{Err_i}\right)$$

Forth, the modifying process of the data weights for every *i* based on N (the classifier's number) is calculated associated with the weak classifier data tests (x) adjusting as the output data.

S1.3. Support vector machine regression (SVMR)

The most possible utilization for two main support vector clustering (SVC) and support vector regression (SVR) is the application of a series of machine learning algorithms called support vector machine (SVM) [4]. Usually, the support vector regression (SVR) can be used for the soft computation associated with the well-established mathematical relationship. Nowadays, the application of support vector regression (SVR) has attracted due to its capability to model and assessment of different complex structures [5]. The key aspects of the SVR concept are considered in this study as follows:

For example, for a series of dataset as $[(x_1, y_1),\ldots..,(x_n, y_n)]$ associated with $x \in R_d$, while the d is the one-dimensional input space and $y \in R$ considered as the output vector in order to estimate output data as follows:

$$y = f(x) = w.\phi(x_i) + b \qquad (1)$$

herein, the weight represented by *w,* the *b* presents the bias vectors, and the $\phi(x)$ is the kernel's function. In 1997, Vapnik and his coworkers [6] suggested the process in order to obtain the bias vectors and correct values of the weight as:

$$minimize\ \frac{1}{2}w^T w + C\sum_{j=1}^{N}\left(\zeta_j^- + \zeta_j^+\right)$$

$$\begin{cases}\left(w.\emptyset(x_i) + b\right) - y_i \le \varepsilon + \zeta_j^- \\ y_i - \left(w.\emptyset(x_i) + b\right) \le \varepsilon + \zeta_j^+ \\ \zeta_j^+.\zeta_j^- \ge 0\ .\ i = 1.2....m\end{cases} \qquad (2)$$

where, the transpose operator presented by *T*, the error tolerance represented by *ε*, C related to the positive regularization which is responsible for describing the variance of different parameters such as *ε*, ζ_j^+, and ζ_j^-. The latest parameters show the lower and higher additional variations, respectively. Construction of the Lagrange multipliers, the above-constrained

optimization issue becomes a dual function that leads to the final solution, which is presented as follows:

$$y = f(x) = \sum_{i=1}^{n} \left(a_i - a_i^*\right).K(x_i.\, x) + b \qquad (3)$$

herein, the kernel function represented by $K(x_k,x_l)$. In addition, the Lagrange multipliers indicated by a_k, and a_k^*.

S1.4. Random Forest (RF)

The decision tree is a helpful machine learning technique. However, it has two flaws. First, while the prediction bias of the decision tree is usually low, the prediction variance may be high because it is sensitive to minor perturbations in the training dataset; second, while the dividing rule for each node is suitable, as shown in the previous section, this greedy method cannot assure that the decision tree is preferable. By sequentially training numerous trees and transforming several weak classifiers to strong classifiers, ensemble techniques can alleviate these two difficulties. A random forest is made up of a collection of different decision trees that are all being trained at the same time. The algorithm determines the superiority and relevance of each decision tree. [7]. Furthermore, a built-in attribute of the RF classifier, which is used to pick distinct features, allows the RF to manage various inputs features without the requirement to remove a number of parameters for data preprocessing. [8]. The RF methodology uses an approach known as Bagging in the modelling to increase the diversity of trees in the forest (which stands for bootstrap aggregating). Typically, the model is provided with the population of trees as an input, and the model divides data points into distinct groupings as a result. Bagging is a form of random sampling approach that uses just a third of the data points in the training step of the subtree construction process, with the other data points getting regarded to as the out-of-bag data (OOB). Furthermore, cross-validation of findings is not necessary in the RF during model construction since the correctness of the model can be assessed using the OOB's mistakes 46. The RF technique strategy is depicted in Figure 2. If the model is provided with a training dataset as a prerequisite, the training procedure will be effective.

$D = [(x_1.y_1).(x_2.y_2).\cdots(x_n.y_n)]$ if in this form is a training data, D_t for the prescribed training dataset for tree h_t, and H^{oob} for the resultant estimate of the out-of-bag dataset of sample x, as follows:

$$H^{oob}(x) = argmax \sum_{t=1}^{T} I(h_t(x)) = y$$

The OOB dataset's inaccuracy is simplified as follows for modelling purposes:

$$\varepsilon^{oob}(x) = \frac{1}{|D|} \sum_{(x.y)\epsilon D} I(H^{oob}(x) \neq y)$$

The functioning of the RF should be random and unsystematic, and this aspect is regulated by the parameter $k = log_2 d$ [9]. The following phrase may be used to determine the relevance of a property of a variable X_i:

$$I(X_i) = \frac{1}{B} \sum_{t}^{B} O\tilde{O}Berr_{t^i} - OOBerr_t$$

Correspondingly, the i^{th} factor is denoted by ^{X_i} in the X vector, B represents the number of trees in the current RF, $^{O\tilde{O}Berr_{t^i}}$,which remains for the feature X_i of tree t, and the initial OOB data samples are given as ^{OOBerr_t}, which contains the subset variables, and the estimated error of the OOB samples is described by $^{O\tilde{O}Berr_{t^i}}$.

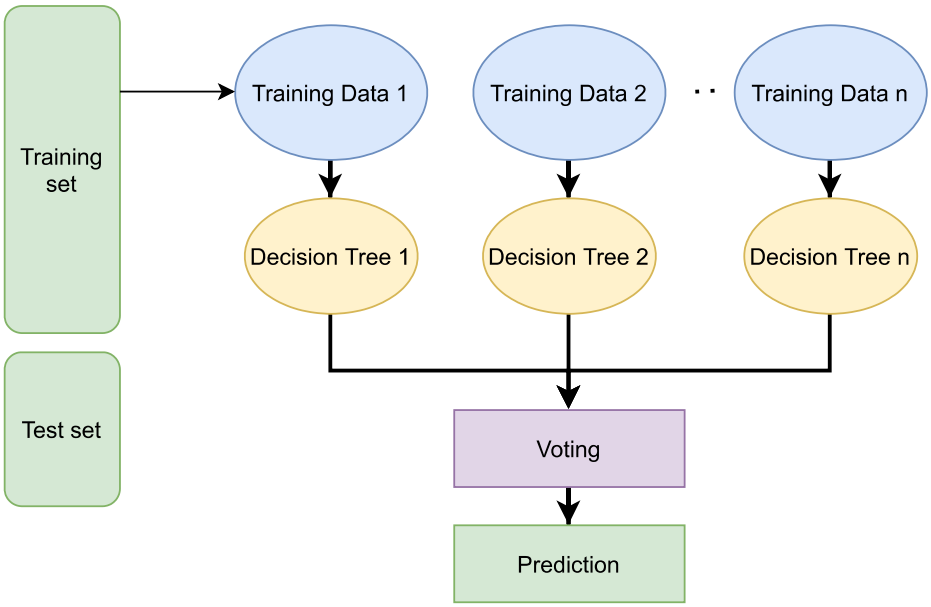

Figure S1. A schematic illustration of random forest algorithm.

S1.5. Extra Tree

Geurts *et al.* were the first for the suggestion of extra trees (ETs),[10] which consists of adding a random layer to decision trees. When the training procedure is over, the stated layer will be inserted into the node. As a result, the time spent looking for the optimal cut-points will be considerably reduced, as a collection of randomly determined threshold values will be used instead of cutting points. Furthermore, by employing this method, scientists may minimize the volume of data storage space necessary, resulting in a better and faster training process. A significant disadvantage of this strategy is the increase in the size of the forest. Geurts and his co-workers compared ET to other alternative techniques such as random decision forests and represented in a previous study.[10] Based on the results of the comparisons, it was determined that ETs can not only perform as well as other approaches but also, in certain situations, even outperform them. It's also worth mentioning that recognized ETs have lower final variance. As a result, it has been acknowledged that the ETs are impressively resilient to a collection of noisy training data.

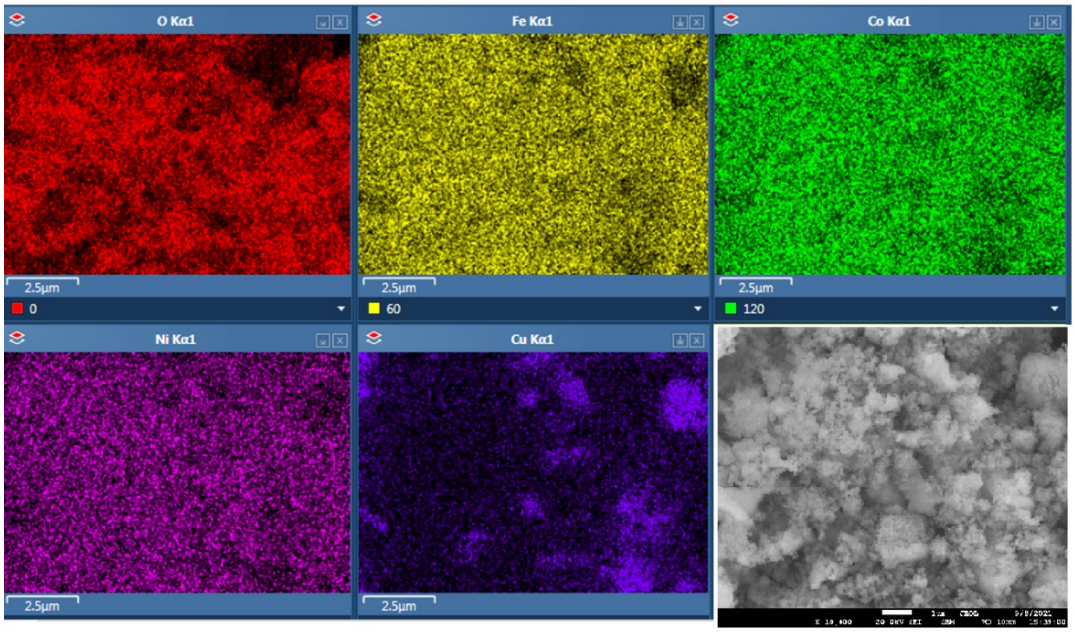

Figure S2. SEM image and the corresponding EDS maps of FeCoNiCu-MEO.

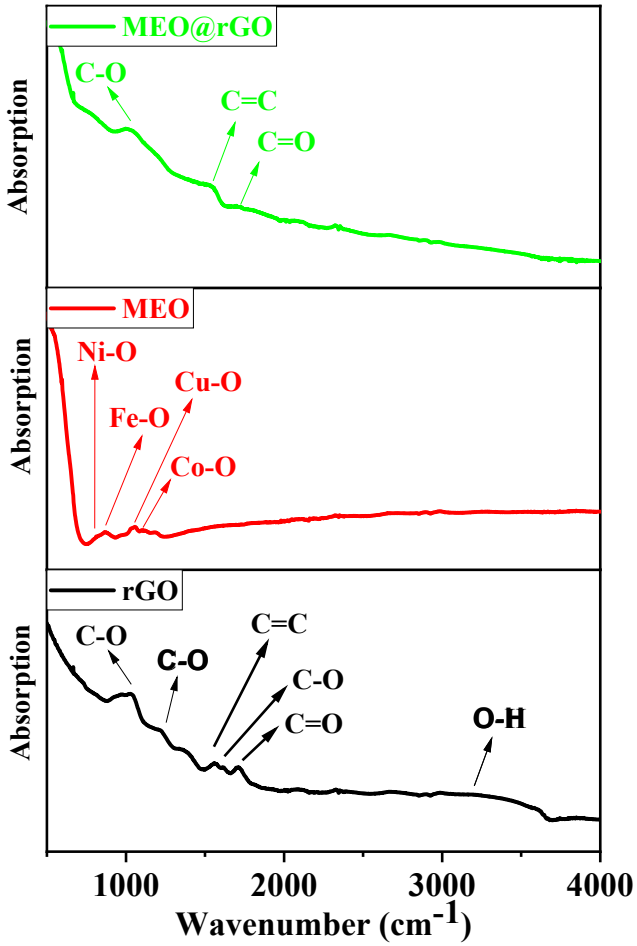

Figure S3. FTIR spectra of rGO, FeCoNiCu MEO, and MEO@rGO;

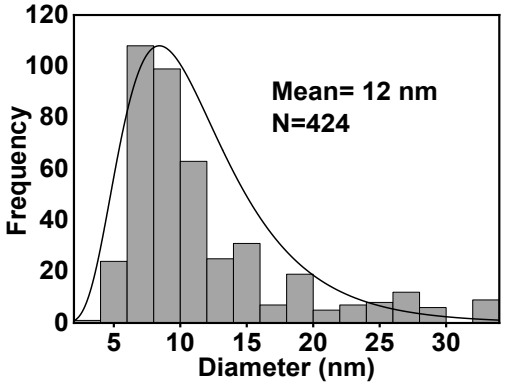

Figure S4. Particle size distribution of FeCoNiCu-MEO.

Table S1. Aerobic and solvent-free oxidation of toluene at different reaction conditions.

Entry†	Catalyst	T (˚C)	P (atm)	Conversion (%)	Selectivity (%)			
					BAl	BAd	BAc	BBz
1	None	100	5	0.01	5.1	80.5	14.4	0.0
2	None	125	5	0.01	10.2	73.3	15.1	1.4
3	None	150	5	0.02	18.4	67.8	12.2	1.6
4	None	100	10	0.05	15.3	81.2	3.5	0.0
5	None None [a, b]	125	10	0.11 0.13[a], 0.13[b]	12.3 10.6 [a], 11.3 [b]	80.4 78.9 [a], 76.3 [b]	6.4 9.1 [a], 12.1 [b]	0.9 1.4 [a], 0.3 [b]
6	None	150	10	0.17	16.4	75.2	7.9	0.5
7	None	100	15	0.13	13.5	77.6	8.9	0.0
8	None None [a, b]	125	15	0.14 0.18[a], 0.21[b]	11.2, 10.2[a], 9.3[b]	79.6 78.3[a], 71.2[b]	9.2 9.1[a], 17.6[b]	0.0 2.4[a], 1.9[b]
9	None	150	15	0.21	15.2	81.5	2.9	0.4
10	MEO [e]	100	5	0.21	25.3	73.4	1.3	0.0
11	MEO [e]	125	5	0.27	12.7	84.6	2.5	0.2
11	MEO [e]	150	5	0.35	15.3	69.7	12.4	2.6
12	MEO [e]	100	10	1.24	21.1	75.3	3.6	0.0
13	MEO [e] MEO [a, b]	125	10	4.15 4.50[a], 5.12 [b]	16.3 15.3 [a], 14.0 [b]	78.3 75.8 [a], 71.3[b]	4.5 4.3 [a], 8.6 [b]	0.9 4.6[a], 6.1 [b]
	MEO	125	10	13.1	13.5	66.4	16.7	3.4
14	MEO [e]	150	10	5.51	15.6	72.5	8.6	3.3
15	MEO [e]	100	15	2.31	21.3	70.3	5.6	2.8
16	MEO [e]	125	15	4.89	15.2	75.8	6.1	2.9
17	MEO [e]	150	15	5.77	14.2	78.6	5.4	1.8
18	rGO [f]	125	10	2.04	9.3	77.2	11.4	2.1
	rGO	125	10	2.81	8.7	76.7	8.4	6.2
19	rGO [f]	150	10	2.23	13.5	75.8	9.3	1.4
20	rGO [f]	125	15	2.27	10.4	81.3	6.8	1.5
21	rGO [f]	150	15	2.56	8.1	78.5	12.2	1.2
22	MEO@rGO	100	10	6.7	12.6	87.2	0.2	0.0
23	MEO@rGO MEO@rGO [a, b, c, d]	125	10	12.6 14.2[a], 17.5 [b], 15.6 [c], 18.2 [d]	3.1 3.6[a], 4.2 [b], 2.9[c], 3.4 [d]	94.7 94.0 [a], 92.5 [b], 93.5 [c], 92.1 [d]	1.7 1.5 [a], 2.7[b], 2.5 [c], 3.3 [d]	0.5 0.9 [a],0.6 [b], 1.1 [c], 1.2 [d]
24	MEO@rGO MEO@rGO [a, b, c, d]	150	10	13.8 15.4[a], 18.7 [b], 14.6 [c], 16.7 [d]	4.0 4.3[a], 4.8[b], 4.5[c], 5.1 [d]	93.4 91.5 [a], 90.3 [b], 91.9 [c], .89.2 [d]	1.5 3.5 [a], 4.1[b], 2.6 [c], 4.2 [d]	1.1 0.7 [a],0.8 [b], 1.0 [c], 1.5 [d]
25	MEO@rGO	100	15	7.8	8.7	86.2	4.5	0.6
26	MEO@rGO	125	15	13.4	3.7	92.1	3.8	0.4
27	MEO@rGO	150	15	15.8	6.2	87.4	5.9	0.5

†Typical oxidation reaction conditions: 3 ml of toluene, 0.04 g catalyst (actual weight ratio of rGO to MEO in the resultant nanocomposite is 10 to 4), stirring rate of 500 rpm, 4 h under specific air pressure. [a] 6h , [b] 8h , [c] 0.06 g, [d] 0.1 g, [e] the amount of MEO used in this case is equal to the amount of ratio used in the rGO@MEO (0.016 g), [f] the amount of rGO used in this case is equal to the amount of ratio used in the rGO@MEO (0.024 g). BAl: benzyl alcohol, BAd: benzaldehyde, BAc: benzoic acid, and BBz: benzyl benzoate.

Table S2. Comparison of the catalytic performance of MEO@rGO catalyst with reported studies in the literature.

Catalyst	Temperature (°C)	Pressure (atm)	Time (h)	Conversion (%)	Selectivity to benzaldehyde (%)	Reference
Au@Silicalite-1	160	10	24	10.7	90.5	[11]
Fe_2O_3/HZSM-5	90	H_2O_2	4	15.6	52.2	[12]
Carbon	160	10	7	0.1	72.6	[13]
Au/C	160	10	7	0.2	81.9	[13]
Au-Pd/C	160	10	7	0.3	57.6	[13]
Pd/C	160	10	7	1.6	56.4	[13]
mpg130	160	10	16	8.5	66	[14]
SBA-15	180	10	1	2.4	56.8	[15]
T(p-Cl)PPMnCl	100	1	8	6.44	51.9	[16]

Table S3. Atomic composition of the used MEO@rGO catalyst after 5 times recycling experiments asd etermined by ICP-MS.

Element	Fe(%)	Co(%)	Ni(%)	Cu(%)
MEO@rGO	4.58	4.15	4.52	3.92

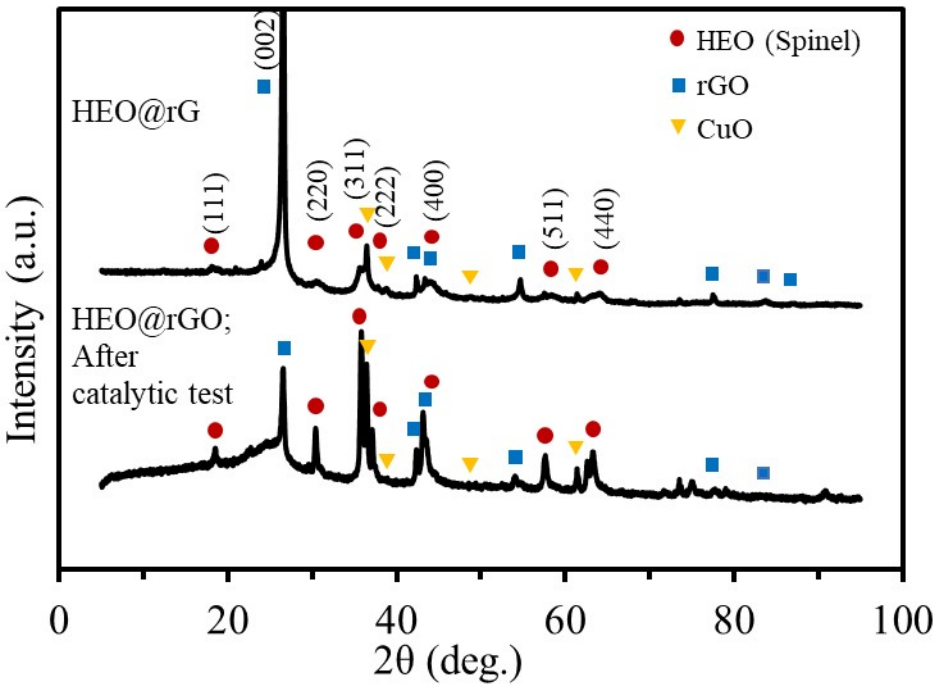

Figure S5. XRD patterns of MEO@rGO after the catalytic test.

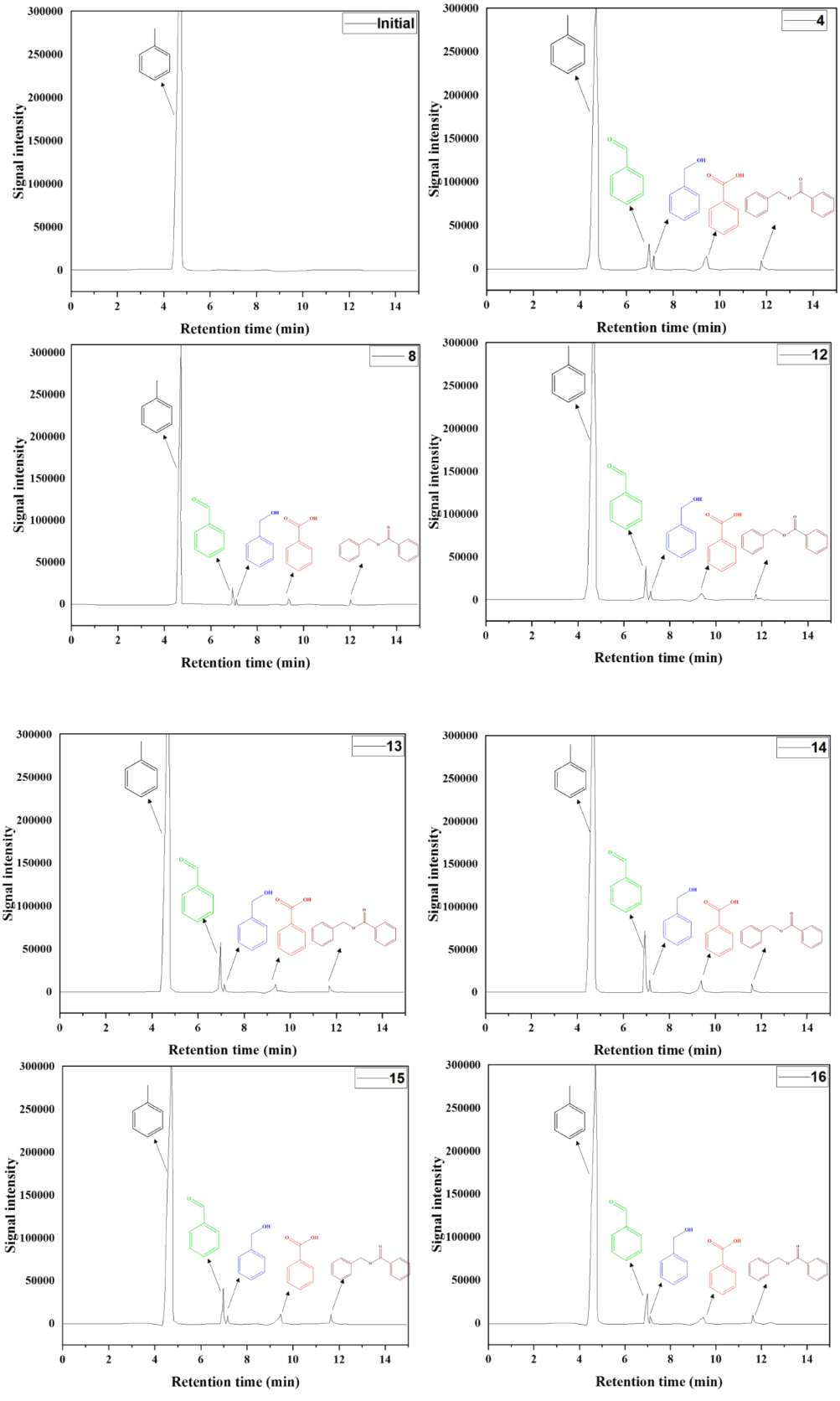
Initial
4
8
12
13
14
15
16
Signal intensity
Retention time (min)

Figure S6. Gas chromatography spectra of the pure toluene (Initial) and selected samples related to Table 3: MEO_4 h_0.04 g (4), rGO_4 h_0.04 g (8), MEO@rGO_4 h_0.04 g (12), MEO@rGO_6 h_0.04 g (13), MEO@rGO_8 h_0.04 g (14), MEO@rGO_4 h_ 0.06 g (15), MEO@rGO_8 h_0.1 g (16).

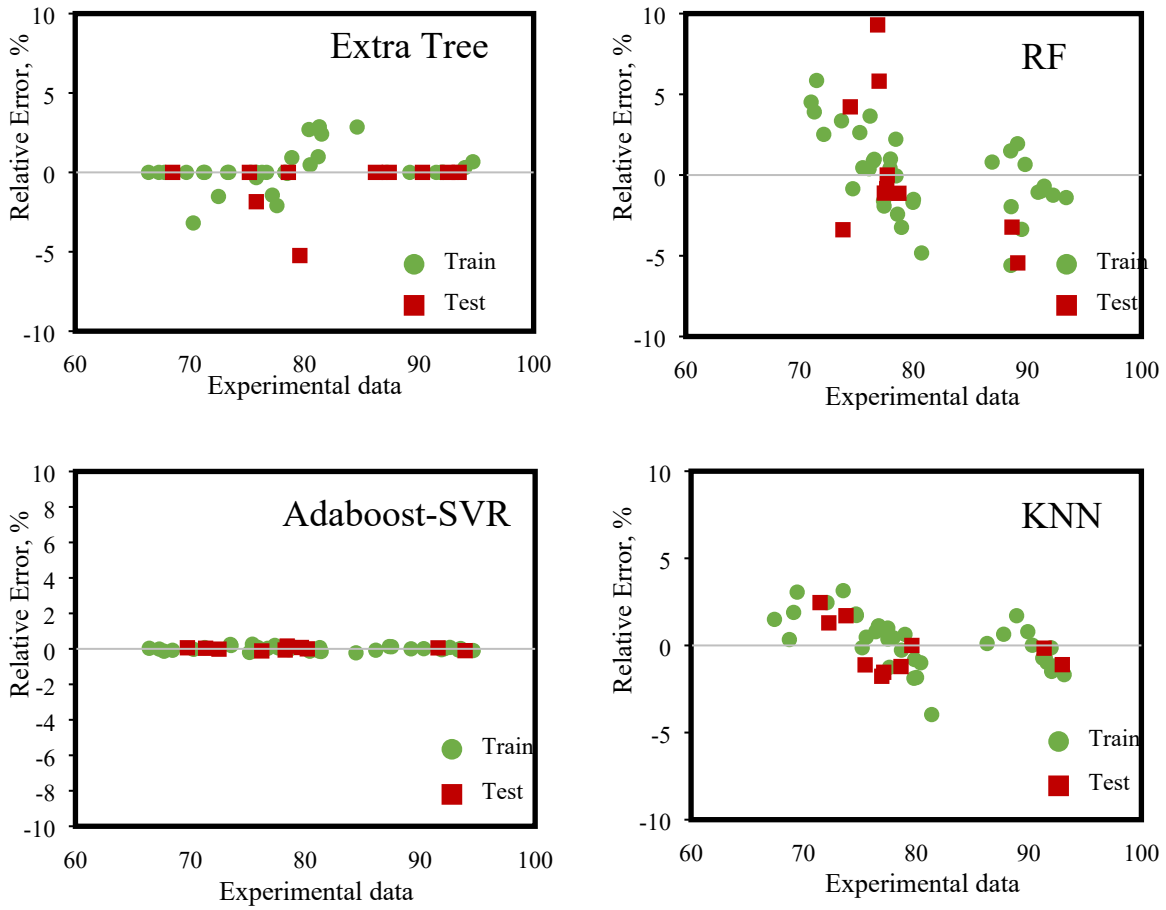

Figure S7. Error distribution plots obtained from each method

References

1 A. Jhamtani, R. Mehta and S. Singh, *IIMB Manag. Rev.*, 2021, **33**, 184–190.

2 Y. Freund and R. E. Schapire, *J. Comput. Syst. Sci.*, 1997, **55**, 119–139.

3 M. R. Mohammadi, F. Hadavimoghaddam, M. Pourmahdi, S. Atashrouz, M. T. Munir, A. Hemmati-Sarapardeh, A. H. Mosavi and A. Mohaddespour, *Sci. Reports 2021 111*, 2021, **11**, 1–20.

4 A. J. Smola and B. Schölkopf, *Stat. Comput. 2004 143*, 2004, **14**, 199–222.

5 B. Schölkopf, A. J. Smola, R. C. Williamson and P. L. Bartlett, *Neural Comput.*, 2000, **12**, 1207–1245.

6 V. Vapnik, S. Golowich and A. S. Neural, in *books.google.com*, 1997, p. 281.

7 Y. Wu and S. Misra, *IEEE Geosci. Remote Sens. Lett.*, 2020, **17**, 1144–1147.

8 T. Shaikhina, D. Lowe, S. Daga, D. Briggs, R. Higgins and N. Khovanova, *Biomed. Signal Process. Control*, 2019, **52**, 456–462.

9 L. Yang, H. Wu, X. Jin, P. Zheng, S. Hu, X. Xu, W. Yu and J. Yan, *Sci. Reports 2020 101*, 2020, **10**, 1–8.

10 P. Geurts, D. Ernst and L. Wehenkel, *Mach. Learn. 2006 631*, 2006, **63**, 3–42.

11 H. Huang, W. Ye, C. Song, Y. Liu, X. Zhang, Y. Shan, Y. Ge, S. Zhang and R. Lu, *J. Mater. Chem. A*, 2021, **9**, 14710–14721.

12 X. Li, B. Lu, J. Sun, X. Wang, J. Zhao and Q. Cai, *Catal. Commun.*, 2013, **39**, 115–118.

13 G. J. H. L. Kesavan, R. Tiruvalam, M. H.i A. Rahim, M. I. Saiman, D. I. Enache, R. L. Jenkins, N. Dimitratos, J. A. Lopez-Sanchez, S.t H. Taylor, D. W. Knight, C. J. Kiely, *Science*, 2011, **331**, 195–199.

14 X.-H. Li, X. Wang and M. Antonietti, *ACS Catal.*, 2012, **2**, 2082–2086.

15 W. Zhong, S. R. Kirk, D. Yin, Y. Li, R. Zou, L. Mao and G. Zou, *Chem. Eng. J.*, 2015,

280, 737–747.

16 W. Deng, Y. Wan, H. Jiang, W.-P. Luo, Z. Tan, Q. Jiang and C.-C. Guo, *Catal. Lett. 2013 1442*, 2013, **144**, 333–339.