Prediction of chemical reactions using statistical models of chemical knowledge

Philipp-Maximilian Jacob[1,2], Alexei A. Lapkin[1,2,*]

[1] Department of Chemical Engineering and Biotechnology, University of Cambridge, Philippa Fawcett Drive, Cambridge CB3 0AS, UK

[2] Cambridge Centre for Advanced Research and Education in Singapore Ltd. 1 Create Way, CREATE Tower #05-05, 138602 Singapore

*Correspondence to: aal35@cam.ac.uk

Abstract: Is chemistry discoverable or can it only be invented? – this is the question of a computer scientist and a philosopher of science when looking at application of artificial intelligence methods for developing new chemical entities and new chemical transformations. This study confirms that, at least today, chemistry is, in part, discoverable from past history of chemical research – the accumulated chemical data contains hidden rules of chemistry, which can be exploited to discover new reaction pathways. This is shown using a stochastic block model approach, trained on chemical reaction data obtained from Reaxys®.

Keywords: graph theory; network theory; network of chemical reactions; reaction prediction

Today a very large and rapidly growing amount of chemical knowledge is available through online databases. Reaxys (*1*) alone contains over 40 million reactions and in excess of 105 million compounds (*2*). Yet, this is only a fraction of the possible chemical space: it is estimated that 10^{60} drug-like molecules are synthetically accessible (*3*, *4*). Modern high-throughput screening can test "only" in the order of 10^6 molecules in the lab (*5*), leaving a large gap for which tools are required to guide experimentation (*4*). Without algorithmic use of chemical knowledge, capable of navigating within the whole of known chemical space, our efforts would continue to be limited as illustrated by the fact that in medicinal chemistry, as an example, few heavily used reactions have dominated the chemical landscape with no additions to this set in the past twenty years (*6*).

Machine learning and retrosynthetic algorithms so far have not provided the desired solution. 50 years after their development classical retrosynthetic methods have not been able to firmly establish themselves in use (*7*). Recent proliferation of electronic data and computational resources has seen a rapid growth in the application of machine learning and artificial intelligence to reaction prediction. At present this approach is, however, plagued with sparsity of chemical knowledge and poor data quality (*8–10*), potentially making accurate predictions of reaction outcomes mathematically impossible (*11*).

Abstracting complex data sets into networks is a well-established approach in trying to study trends underlying complex data. Chemical data has, for example, been converted into networks and used for planning of synthesis routes (*8*, *12–15*) or as an analytical tool to study the structure of chemical knowledge (*16–19*). This lead us to formulate a hypothesis for algorithmic use of chemical knowledge, which has not been explored in chemical science to date: the structure of chemical networks by itself contains information about chemistry 'rules' and appropriate mathematical treatment of the structure of chemical networks can give insights into yet un-discovered phenomena, such as new reactions without being adversely affected by poor or incomplete reaction data, such as errors in reporting, propagated into the databases or errors in abstracting the data from primary publications into the databases.

A way to circumvent the problem of inaccuracies present in reaction data is presented by Stochastic Block Models (SBMs), used in this study, which had been shown to be useful in community detection: understanding the community structure of a network allows interpretation

of the network, and binary interaction data has been shown to be sufficient to identify missing and spurious observations, as well as future dynamics (*20–23*).

Here we show that by taking a sub-set of observations of the chemical space we are able to predict reactions between species that have not been observed yet. In removing all chemical data other than the most fundamental interactions between species we are able to circumvent many of the problems around incomplete or erroneous reaction data plaguing other approaches. Using Markov chain Monte Carlo algorithms we then predict "missing" reactions, first in a test case, to demonstrate its performance before applying it to two case studies (30). In a previous study we investigated the use of network representations of chemical data obtained from Reaxys (termed "Network of Organic Chemistry" or "NOC") in planning efficient synthesis routes from limonene to paracetamol (*15*). This formed the basis of this study in a network containing 161,760 chemical species (30).

We trained the algorithm on the sub-set of reactions known by a specific date and checked whether the algorithm would correctly distinguish between randomly generated node connections and the reactions that were actually discovered in the following decades (30). Such time-split validation is not a fully random test but one that resembles the actual challenge more closely (*24*).

A standard metric in the machine learning community to classify the performance of prediction algorithms is the area under the receiver operating characteristic (ROC) curve (AUC). It can be interpreted as the probability that a randomly chosen missing reaction (a true positive) is assigned a higher score than a randomly chosen pair of unconnected vertices (a true negative) (*22*). Upon successful model selection (see Fig. S1, S2, and supplementary text) all AUC values after the year 1890 take values around 0.9, indicating excellent performance of the algorithm, Figure 1. Thus, the probability that a randomly chosen positive will rank higher than a randomly chosen negative is 90%.

When comparing the algorithm to the results obtained for benchmark graph theoretical algorithms the SBM-based approach continues to perform remarkably well (Figure 1). The SBM-based approach clearly outperforms all tested metrics. It is worth pointing out that amongst the benchmark algorithms only the preferential attachment approach manages to exceed an AUC

of 0.5, thus all other metrics fare, at times significantly, worse than using chance alone in predicting missing reactions.

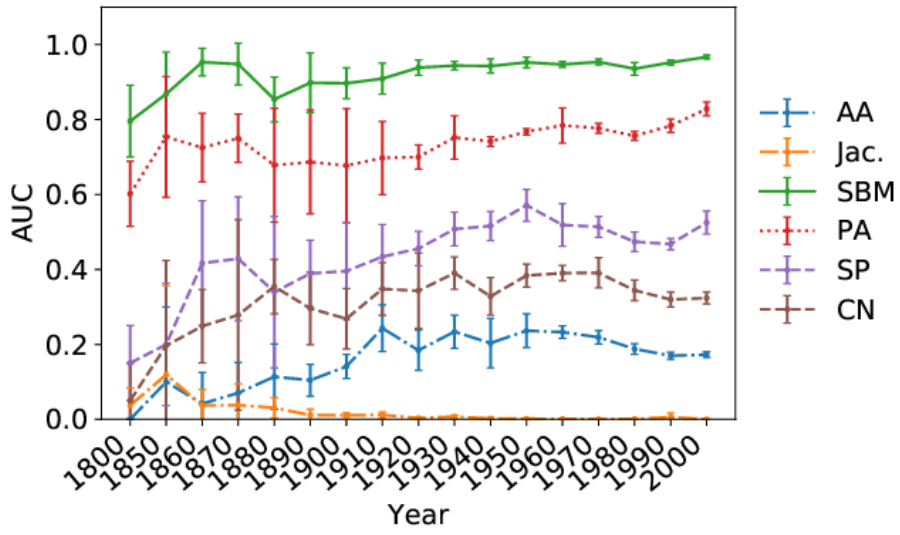

Figure 1: The AUC for the SBM-based link prediction approach compared against a number of local similarity-based approaches. AA stands for Adamic-Adar, Jac. for Jaccard index, SBM for stochastic block model, PA for preferential attachment, SP for shortest path, and CN for common neighbours.

In a significant number of publications the reported data is unreliable, meaning the data cannot easily be used for prediction, unless rigorous statistical analysis is employed (*25*). A consequence of the lack of statistical treatment of the data by, for example, the "conventional" metrics, is that they are liable to over fitting. If the data used contains erroneous entries this is a problem when using it to predict new, unknown reactions. One of the big advantages of the SBM-based approach in comparison is its ability to trade off noise versus genuine statistical features of the data (*26–28*).

Looking at the results for the SBM-based approach shown in Figure 1 we conclude that the link prediction algorithm performs very well on the network investigated, being able to recover a remarkably large share of true positives very quickly with a very low rate of false positives. Performance of a certain SBM is always dependent on the network it has been fit to and whether

it can describe that class of networks well. Thus, comparison of it to other methodologies fitted to other networks in other publications is not straightforward. Comparing it to the AUC curves reported in (*22*) it seems to outperform the results shown. In the absence of AUC curves, comparing accuracy of the algorithms, the algorithm used here outperforms the results shown in (*21*) (plots of the accuracy of the algorithm here tested can be found in Fig. S3).

It can be concluded that, given the data available, the algorithm employed here seems to not only perform very well but also to outperform the results shown in literature and is indeed able to predict reactions correctly. This has been verified using time split validation on historical networks.

We then applied this methodology to the conversion of limonene to paracetamol as our group is interested in converting terpenes into useful products in order to valorize paper waste through a circular economy approach. The first task was to design a set of potential edges that, if correct, would provide more efficient synthesis routes from limonene to paracetamol. Testing all combinatorially-possible transformations in this network would require ranking of more than 26 billion transformations, many of which would be irrelevant for this system. From our previous study (*15*) we know that the synthesis can be carried out in five steps. Taking the set of molecules involved in the four- and five-step routes allowed us to ensure that any discovered transformation was relevant.

Since our goal was to rely on a minimum of chemical insight, all molecules lying on a four-step route were recombined to generate the list of all mathematically possible new transformations. Our network contained 27 four-step paths involving a total of 25 distinct chemical species. This meant that there were a total of 519 edges possibly connecting a given pair of the 25 species that had not been discovered yet. These were ranked according to their likelihood ratios (30). The earlier study applied heuristics to determine which reaction products were chemically desirable. Applying these for consistency left a total of 355 edges. Finally excluding all transformations not resulting in a path executable in four steps or less left a total of 116 edges of interest. The 16 most highly scoring of these can be found in Figure 2 and 3. Closer investigation revealed that at least 15 of the 116 transformations (and 52 of the 355 transformations) were correct predictions, known to Reaxys but not contained in this dataset. A selection can be found in Figure S4.

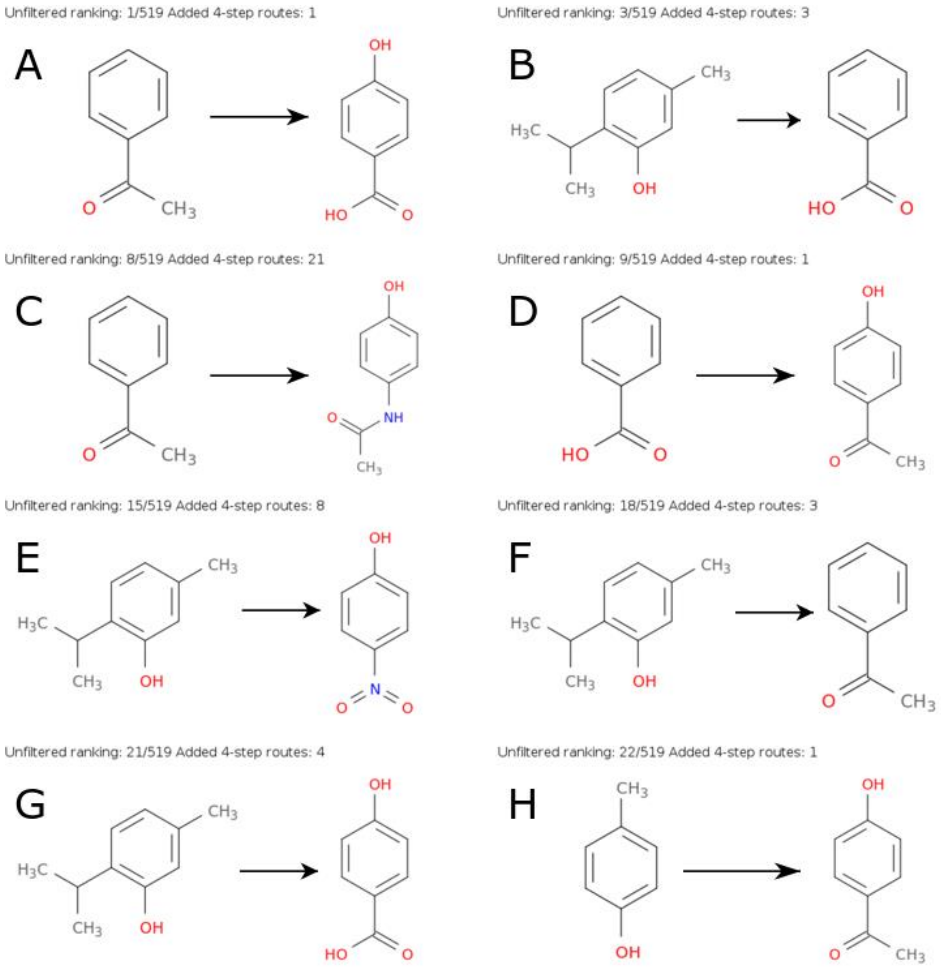

Figure 2: First part of reaction predictions, showing filtered ranks 1-8. Only the main reactant and product are shown.

Expanding the search set to five-step paths will require evaluation of more than 20,000 transformations. Clearly further heuristics are required to decide which transformations might merit a closer investigation. A first question is which transformations would be of value chemically. Obviously of interest here would be a transformation that turns a less reactive functional group into a reactive one as it increases the molecule's "synthetic value" by opening it up for further transformations. In our case study a transformation pattern found repeatedly is the conversion of a less reactive group into a highly reactive amine, as is the case in the reactions 40, 62, or 110 shown in Figure 3.

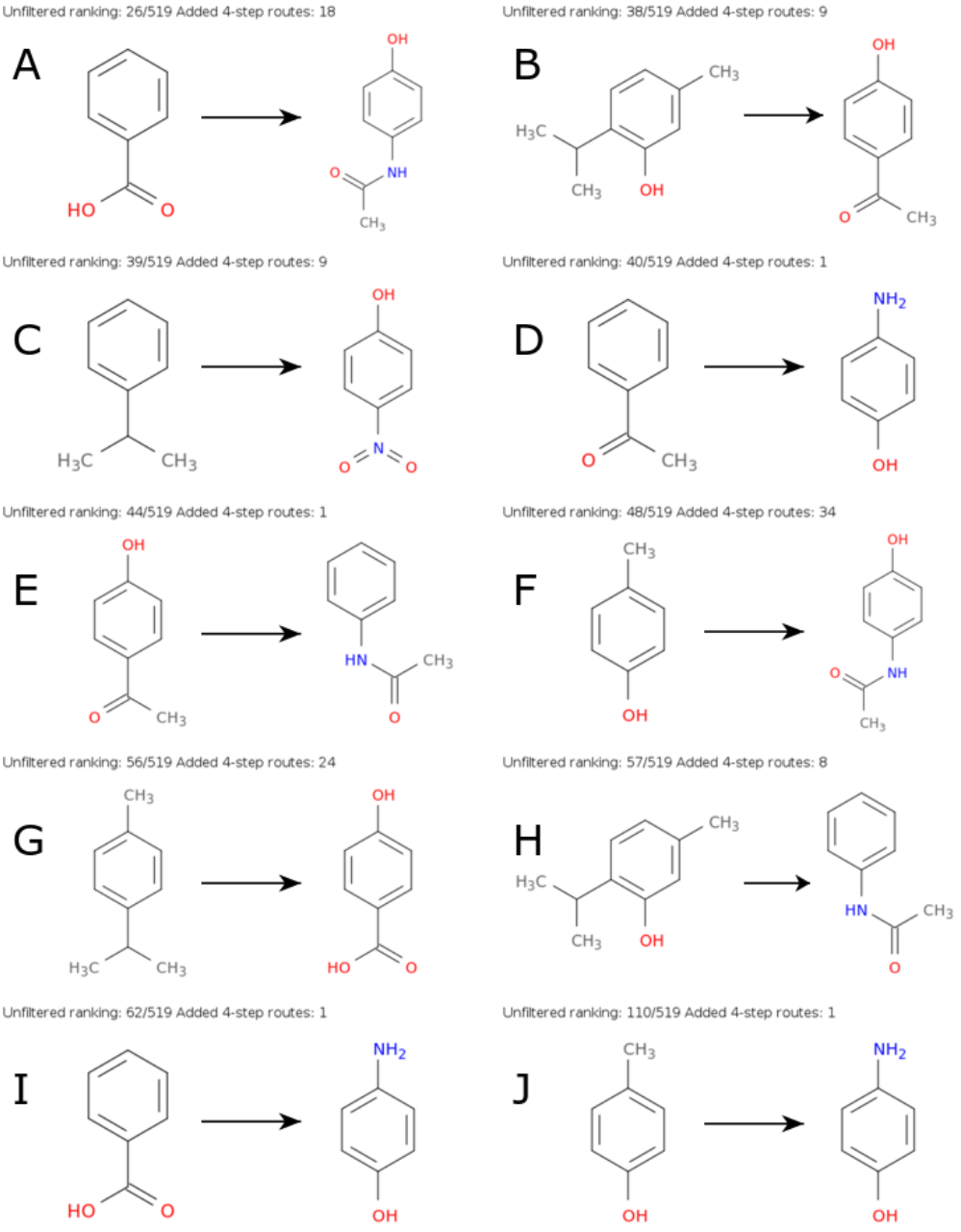

Figure 3: Second part of reaction predictions, showing filtered ranks 9-16, 18, and 31. Only the main reactant and product are shown.

Of similar value would be a transformation making a promising, new feedstock accessible. A number of proposed transformations are suggesting the use of thymol, which can be sourced sustainably, as reactant. Transformations that use thymol are reactions 3, 18, 21, 38 and 57 (the

first three can be found in Figure 2 and the last two in Figure 3), which also represent transformations deemed most probable by the algorithm.

Finally, we can ask if a transformation significantly increases the size of the accessible chemical space. In (*15*) we have shown that addition of a single key transformation can dramatically increase the number of synthesis route options. Clearly, such a transformation could be a highly valuable target to pursue. Here the insertion of a nitrogen atom between benzene ring and carbonyl group (though in some cases a further transformation is required to obtain a carbonyl group) is a transformation that is repeatedly found (*c.f.* ranks 8, 26, 44, and 48) and would result in 74 additional routes of interest (30), rendering it a potential target for closer investigation.

The results of this algorithm are dependent on the edges that are being tested. If the initial set contained no viable transformations, then even the most highly-ranked transformation will still not be feasible. Thus, the test set was expanded considering all molecules involved in the five-step synthesis routes. This yielded 21,080 different transformations that were ranked (30). Applying the ring-count heuristic drops this to 14,474 transformations and applying all the same initial heuristics reduces the count to 2,585 transformations. At least 38 of these (and 540 of the 14,474 transformations) were already known to Reaxys. Results can be found in Figures S5-42. Comparing these results to those in Figure 2 and 3, the highly ranking transformations from the four-step set continue to score highly in the five-step set (Table S4), showing that the method efficiently separates likely from unlikely transformations.

Finally, we consider whether a given transformation appears thermodynamically feasible. A very large positive change in Gibbs free energy of reaction (ΔG_R°) might be used to discount a suggested transformation initially. Doing so for the top three-ranking transformations yields ΔG_R° of 124 kcal mol^{-1}, -139 kcal mol^{-1}, and -148 kcal mol^{-1} (30). The first transformation thus has a very large positive ΔG_R°making it highly non-spontaneous under the assumptions used perhaps providing reason to discount it. The other two reactions do not have this problem.

Clearly the energy analysis is heavily impacted by the assumptions around reaction stoichiometry, solvent choices, or temperature, possibly impacting the calculation significantly. What this thus cannot provide is a hard and fast classification into feasible and unfeasible reactions. What it does provide are data points for initial scoring of predictions. Given the large amount of data returned by the link prediction, metrics are required which can be used to decide

which reactions to focus on and which to perhaps save for a later stage. Undoubtedly there will be approaches yielding greater fidelity or insight into the feasibility or required processes. These will, however, require a greater investment of time and thus may not be suitable for this early stage of analysis.

In conclusion, by 2011 Bayer had halted two-thirds of its target validation programs because in-house experimental findings did not correspond with published literature (*10*). Unreliable data and sparsity of chemical data are serious problems which, at this stage, almost all data-driven applications in chemistry are struggling with. One of the great strengths of the statistical approach applied here is its ability to differentiate structure from noise in a principled way. Therefore, assessments about the relative likelihood of a given transformation compared to a set of other transformations can be made while dealing with the noise in the data in a statistically rigorous manner. By eliminating as much peripheral data from the evaluation process as possible the application of SBMs to the network allows the elimination of as many error sources as possible. Though the results are influenced by the researcher's choices of transformations to be tested, the method is able to efficiently separate likely from unlikely transformations. This allows both a thorough scan of broad areas of chemical space to find a high-likelihood set for in-depth investigation, as well as, the evaluation of a more well-defined set to evaluate chemical intuition against statistical insights. This work presents the first application of these methods to reaction prediction and, we believe, makes an important contribution to the field of reaction prediction. We hope this will help to ground reaction prediction in a rigorous treatment of statistical data and help to address some of the problems that have plagued the field thus far.

Acknowledgements

The authors would like to thank Dr Tiago Peixoto for freely making his code available in graph-tool (*29*) which enabled this work. P.-M. Jacob would like to thank Peterhouse and the University of Cambridge for funding in the form of a PhD studentship. We gratefully acknowledge collaboration with RELX Intellectual Properties SA, Elsevier and their technical support, which enabled us to mine Reaxys. This work was funded, in part, by the EPSRC project **"**Terpene-based manufacturing for sustainable chemical feedstocks**"** EP/K014889. This project is funded in part by the National Research Foundation (NRF), Prime Minister's Office,

Singapore, under its Campus for Research Excellence and Technological Enterprise (CREATE) program as a part of the Cambridge Centre for Advanced Research and Education in Singapore Ltd (CARES).

Supplementary Materials:

Materials and Methods

Supplementary Text

Figures S1-S42

Tables S1-S4

References (31-48)

References and Notes

1. Copyright © 2018 Elsevier Life Sciences IP Limited except certain content provided by third parties. Reaxys is a trademark of Elsevier Life Sciences IP Limited. For further information about Reaxys see https://www.elsevier.com/solutions/reaxys.
2. Elsevier R&D Solutions, Reaxys Fact Sheet (2016) available at https://www.elsevier.com/__data/assets/pdf_file/0005/91616/RDS_FactSheet_Reaxys_Oct_2016-WEB.pdf.
3. J. L. Reymond, L. Ruddigkeit, L. Blum, R. van Deursen, The enumeration of chemical space. *Wiley Interdiscip. Rev. Comput. Mol. Sci.* **2**, 717–733 (2012).
4. P. Kirkpatrick, C. Ellis, Chemical space. *Nature*. **432**, 823–823 (2004).
5. M. H. S. Segler, T. Kogej, C. Tyrchan, M. P. Waller, Generating Focussed Molecule Libraries for Drug Discovery with Recurrent Neural Networks (2017) available at http://arxiv.org/abs/1701.01329.
6. D. G. Brown, J. Boström, Analysis of Past and Present Synthetic Methodologies on Medicinal Chemistry: Where Have All the New Reactions Gone? *J. Med. Chem.* **59**, 4443–4458 (2016).

7. A. Cook *et al.*, Computer-aided synthesis design: 40 years on. *Wiley Interdiscip. Rev. Comput. Mol. Sci.* **2**, 79–107 (2012).

8. P.-M. Jacob, P. Yamin, C. Perez-Storey, M. Hopgood, A. A. Lapkin, Towards automation of chemical process route selection based on data mining. *Green Chem.* **19**, 140–152 (2017).

9. P.-M. Jacob, T. Lan, J. M. Goodman, A. A. Lapkin, A possible extension to the RInChI as a means of providing machine readable process data. *J. Cheminform.* **9**, 23 (2017).

10. A. J. Williams, S. Ekins, V. Tkachenko, Towards a gold standard: regarding quality in public domain chemistry databases and approaches to improving the situation. *Drug Discov. Today*. **17**, 685–701 (2012).

11. G. Skoraczyński *et al.*, Predicting the outcomes of organic reactions via machine learning: are current descriptors sufficient? *Sci. Rep.* **7**, 3582 (2017).

12. S. Szymkuc *et al.*, Computer-Assisted Synthetic Planning: The End of the Beginning. *Angew. Chemie Int. Ed.* **55**, 5904–5937 (2016).

13. M. Kowalik *et al.*, Parallel Optimization of Synthetic Pathways within the Network of Organic Chemistry. *Angew. Chemie Int. Ed.* **51**, 7928–7932 (2012).

14. C. M. Gothard *et al.*, Rewiring Chemistry: Algorithmic Discovery and Experimental Validation of One-Pot Reactions in the Network of Organic Chemistry. *Angew. Chemie*. **124**, 8046–8051 (2012).

15. A. A. Lapkin *et al.*, Automation of route identification and optimisation based on data-mining and chemical intuition. *Faraday Discuss.* **202**, 483–496 (2017).

16. M. Fialkowski, K. J. M. Bishop, V. A. Chubukov, C. J. Campbell, B. A. Grzybowski, Architecture and Evolution of Organic Chemistry. *Angew. Chemie Int. Ed.* **44**, 7263–7269 (2005).

17. K. J. M. Bishop, R. Klajn, B. A. Grzybowski, The Core and Most Useful Molecules in Organic Chemistry. *Angew. Chemie Int. Ed.* **45**, 5348–5354 (2006).

18. B. A. Grzybowski, K. J. M. Bishop, B. Kowalczyk, C. E. Wilmer, The "wired" universe of organic chemistry. *Nat. Chem.* **1**, 31–36 (2009).

19. P.-M. Jacob, A. Lapkin, Statistics of the network of organic chemistry. *React. Chem. Eng.* **3**, 102–118 (2018).

20. R. Guimerà, M. Sales-Pardo, A Network Inference Method for Large-Scale Unsupervised Identification of Novel Drug-Drug Interactions. *PLoS Comput. Biol.* **9**, e1003374 (2013).

21. R. Guimera, M. Sales-Pardo, Missing and spurious interactions and the reconstruction of complex networks. *Proc. Natl. Acad. Sci.* **106**, 22073–22078 (2009).

22. A. Clauset, C. Moore, M. E. J. Newman, Hierarchical structure and the prediction of missing links in networks. *Nature*. **453**, 98–101 (2008).

23. R. Guimerà, M. Sales-Pardo, Justice Blocks and Predictability of U.S. Supreme Court Votes. *PLoS One*. **6**, e27188 (2011).

24. M. H. S. Segler, M. P. Waller, Modelling Chemical Reasoning to Predict and Invent Reactions. *Chem. - A Eur. J.* **23**, 6118–6128 (2017).

25. P. V. Coveney, E. R. Dougherty, R. R. Highfield, Big data need big theory too. *Philos. Trans. R. Soc. A Math. Phys. Eng. Sci.* **374**, 20160153 (2016).

26. M. N. Schmidt, M. Morup, Nonparametric Bayesian modeling of complex networks: an introduction. *IEEE Signal Process. Mag.* **30**, 110–128 (2013).

27. T. P. Peixoto, Nonparametric Bayesian inference of the microcanonical stochastic block model. *Phys. Rev. E*. **95**, 12317 (2017).

28. T. P. Peixoto, Inferring the mesoscale structure of layered, edge-valued, and time-varying networks. *Phys. Rev. E*. **92**, 42807 (2015).

29. T. P. Peixoto, The graph-tool python library. *figshare* (2014), doi:10.6084/m9.figshare.1164194.

30. Materials and methods are available as supplementary materials at the Science website.

31. T. P. Peixoto, Parsimonious Module Inference in Large Networks. Phys. Rev. Lett. 110, 148701 (2013).

32. T. P. Peixoto, Hierarchical Block Structures and High-Resolution Model Selection in Large Networks. Phys. Rev. X. 4, 11047 (2014).

33. T. P. Peixoto, Efficient Monte Carlo and greedy heuristic for the inference of stochastic block models. Phys. Rev. E. 89, 12804 (2014).

34. T. P. Peixoto, Model Selection and Hypothesis Testing for Large-Scale Network Models with Overlapping Groups. Phys. Rev. X. 5, 11033 (2015).

35. M. Rosvall, C. T. Bergstrom, An information-theoretic framework for resolving community structure in complex networks. Proc. Natl. Acad. Sci. 104, 7327–7331 (2007).

36. X. Yan et al., Model selection for degree-corrected block models. J. Stat. Mech. Theory Exp. 2014, P05007 (2014).

37. T. P. Peixoto, in Advances in Network Clustering and Blockmodeling, P. Doreian, V. Batagelj, A. Ferligoj, Eds. (Wiley-VCH Verlag GmbH & Co. KGaA, New York, 2019; http://arxiv.org/abs/1705.10225).

38. T. Vallès-Català, T. P. Peixoto, R. Guimerà, M. Sales-Pardo, On the consistency between model selection and link prediction in networks (2017) available at http://arxiv.org/abs/1705.07967.

39. T. Fawcett, An introduction to ROC analysis. Pattern Recognit. Lett. 27, 861–874 (2006).

40. S. Agarwal, D. Dugar, S. Sengupta, Ranking Chemical Structures for Drug Discovery: A New Machine Learning Approach. J. Chem. Inf. Model. 50, 716–731 (2010).

41. L. Lü, T. Zhou, Link prediction in complex networks: A survey. Phys. A Stat. Mech. its Appl. 390, 1150–1170 (2011).

42. L. Lü, L. Pan, T. Zhou, Y.-C. Zhang, H. E. Stanley, Toward link predictability of complex networks. Proc. Natl. Acad. Sci. 112, 2325–2330 (2015).

43. A. Clauset, C. Moore, M. E. J. Newman, Hierarchical structure and the prediction of missing links in networks - Supplementary information. Nature. 453, 98–101 (2008).

44. A. Vázquez, Growing network with local rules: Preferential attachment, clustering hierarchy, and degree correlations. Phys. Rev. E. 67, 56104 (2003).

45. A.-L. Barabási, R. Albert, Emergence of Scaling in Random Networks. Science. 286, 509–512 (1999).

46. A. Mislove, H. S. Koppula, K. P. Gummadi, P. Druschel, B. Bhattacharjee, in Proceedings of the first workshop on Online social networks - WOSP ’08 (ACM Press, New York, New York, USA, 2008; http://portal.acm.org/citation.cfm?doid=1397735.1397742), p. 25.

47. A. Dengler, E. Fontain, M. Knauer, N. Stein, I. Ugi, Competing concepts in CAOS. Recl. des Trav. Chim. des Pays-Bas. 111, 262–269 (1992).

48. I. Ugi, A. Dengler, The algebraic and graph theoretical completion of truncated reaction equations. J. Math. Chem. 9, 1–10 (1992).

Supplementary Materials for

Prediction of chemical reactions using statistical models of chemical knowledge

Philipp-Maximilian Jacob, Alexei A. Lapkin

Correspondence to: aal35@cam.ac.uk

This PDF file includes:

Materials and Methods

Experimental Design

The sample size was determined by the area of chemical interest. A specific conversion (limonene to paracetamol) was to be investigated thus every molecule that could be reached within one reaction from any molecule on any path not longer than four steps connecting the two molecules was included to form the network. The basis here was the corpus of reactions obtained from Reaxys® under the R&D Collaboration Agreement with Elsevier. Reaxys aims to abstract the corpus of organic chemistry journals. It was decided to exclude any reaction which was incomplete, i.e. did not have any reactants or any products listed in Reaxys. Other than that no data was excluded. Heuristics were developed for which results to look at in detail, however, all results were used to calculate likelihood ratios. All statistical tests were completed a total of four times. The actual reaction prediction was not repeated. The tested hypothesis was that the SBM-based approach would be able to outperform other graph theoretical metrics and could be applied to reaction prediction. No prior assumption had been made about the most suitable SBM configuration. The SBMs were used as implemented in graph-tool.

Network Assembly

The study presented in (1) on the synthesis of limonene to paracetamol was used as a starting point to assemble a network for analysis. It had found 1068 five-step routes to connect limonene to paracetamol involving 47 unique chemical species. These 47 species were used as query species as they concisely describe the chemical space of interest. Using the Reaxys API all reactions using a given species as reactant were downloaded and saved. This was repeated for the remaining 46 species. Subsequently all reactions were downloaded that listed one of the 47 species as product. All duplicate and incomplete reactions, i.e. reactions that had either no products or no reactants declared in Reaxys, were then removed from the set using a script.

The obtained sanitized data was used to assemble a network using graph-tool (2) in Python2.7, excluding all multi-step reactions. The network was assembled in working through the raw data set and adding each species which is not contained in the network yet as a node and labelling it with its Reaxys ID. Then an edge is added from each reactant in that reaction to each product in that reaction. Each edge is labelled with its publication year to screen the graph. This allowed to decrease the network's size by stepping back in time. This resulted in a network of 178,122 nodes and 401,258 edges prior to removal of parallel edges. This was used as base for the analysis of alternative stochastic block models. For the link prediction study itself it was decided to reduce the size of the network further for performance reasons. To achieve this a wiring scheme was adopted registering only the heaviest reactant and the heaviest product. Thus for each reaction only the heaviest product and heaviest reactant were added to the network (each again labelled with its Reaxys ID) and then an arrow was drawn connecting that reactant to that product which was again labelled with its publication year to allow screening. This yielded a much more condensed representation of chemistry now numbering 161,760 chemical species and 132,539 reaction edges.

Analysis of Alternative Stochastic Block Models

Graph-tool (*2*) implements a non-parametric, microcanonical version of stochastic block models described and derived across a series of papers (*3–8*). These models are fit to the data using Bayesian inference. The details of the implementation will not be reproduced here as they can be

found in detail in the cited papers. In the papers Peixoto derives SBM configurations capturing nested blockmodels, degree-correction and overlapping blocks thus in a first step we carried out model comparison to find the most suitable configuration for this problem. In order to test the different configurations of the SBM derived by Peixoto across (*3–8*) and implemented in graph-tool it was necessary to reduce the total network size to limit the otherwise excessive runtime of some configurations. Thus, the network was screened to represent the state of publications in 1930. The obtained network contained 9,452 nodes and 19,448 edges upon removal of all parallel edges.

Two ways of comparing different SBM parametrisations in a statistically meaningful way are used: maximising the posterior likelihood and sampling across the posterior likelihood distribution. Both allow to establish the significance of the results by calculating the likelihood ratio for two candidate models. The first maximises the posterior likelihood of a given model and its parameters by applying the information theoretic Minimum Description Length (MDL) criterion (*9*). In minimising the description length, Σ, the posterior likelihood is maximised and, in effect, Occam's razor is implemented: The simplest model out of all models of equivalent explanatory power is preferred as it chooses the model that most compresses the data thus preventing overfitting (*3*, *4*, *8*). Using this approach the most probable partition could be found for each version of the stochastic block model and the obtained minimum description lengths (MDLs) could be compared. Since the model with the minimum description length is the most efficient representation of the data Σ allows for a principled way to carry out model selection and to discriminate between two alternative models.

To determine which of two candidate models (model 1 and model 2, say), describes the data better and to evaluate the degree of confidence in the preference, the joint posterior probability $P(\{\boldsymbol{b}_l\}, \mathcal{H}|\boldsymbol{A})$ (where $\mathcal{H}$ is the model class being used, $\boldsymbol{A}$ denotes the adjacency matrix, and $\boldsymbol{b}_l$ is the partition of blocks in level l) of each model can be compared via their ratio Λ_1 (*8*):

$$\Lambda_1 = \frac{P(\{\boldsymbol{b}_l\}, \mathcal{H}_1|\boldsymbol{A})}{P(\{\boldsymbol{b}_l\}', \mathcal{H}_2|\boldsymbol{A})}$$

$$\Lambda_1 = \frac{P(\boldsymbol{A}, \{\boldsymbol{b}_l\}|\mathcal{H}_1)}{P(\boldsymbol{A}, \{\boldsymbol{b}_l\}'|\mathcal{H}_2)} \times \frac{P(\mathcal{H}_1)}{P(\mathcal{H}_2)}$$

$$\Lambda_1 = \exp(-\Delta\Sigma)$$

where $\Delta\Sigma = \Sigma_1 - \Sigma_2$ is the difference in description length and it is assumed that both model classes are equally likely *a priori*, meaning that $P(\mathcal{H}_1) = P(\mathcal{H}_2)$. Thus, using Λ_1 is identical to using the MDL criterion, but allows for quantification of the degree of confidence with $\Lambda_1 < 1$, indicating a preference for $\mathcal{H}_2$ and partition $\{\boldsymbol{b}_l\}'$.

In real world networks it is often the case that many fits of an SBM have a roughly equal posterior likelihood (*8*, *10*). Thus these models are equally valid and should also be taken into account. Therefore the second approach samples across the entire posterior likelihood distribution instead. Employing a Bayesian framework this can be achieved by simply taking all model fits and weighting them according to their posterior probability. The entire model classes should be compared by evaluating the so called model evidence by summing over all hierarchical partitions (*8*):

$$P(\boldsymbol{A}|\mathcal{H}) = \sum_{\{\boldsymbol{b}_l\}} P(\boldsymbol{A}, \{\boldsymbol{b}_l\})$$

Using this the posterior odds ratio can be computed (*8*):

$$\Lambda_2 = \frac{P(\mathcal{H}_1|\boldsymbol{A})}{P(\mathcal{H}_2|\boldsymbol{A})}$$

$$\Lambda_2 = \frac{P(\boldsymbol{A}|\mathcal{H}_1)}{P(\boldsymbol{A}|\mathcal{H}_2)} \times \frac{P(\mathcal{H}_1)}{P(\mathcal{H}_2)}$$

Which, again assuming no *a priori* preference on the models, reduces to the Bayes factor and establishes a degree of confidence in the outcome. If a given threshold is exceeded it is possible to reject one model in favour of the other (*8*, *11*).

To find the minimum description length the function `minimize_nested_blockmodel_dl` (or `minimize_blockmodel_dl` in the case of the non-nested model) was used in graph-tool.

The MDL was found for each of the configurations given in Table S1. This was repeated three times. The mean of these values was then found. Their standard deviation was used as an estimate for the error in the results.

Using these results the configuration with the smallest MDL, which corresponds to the preferred configuration, was determined. In order to calculate the statistical significance of this preference Λ_1 was calculated for all other possible configurations compared to the preferred configuration.

Seeing as an exact evaluation of the model evidence is not tractable approximations have to be used (*8*) which are implemented in graph-tool. Thus, during a first step the description length of the SBM is minimised to initialise the Markov chain. Subsequently 200,000 sweeps were carried out using graph-tool's MCMC algorithm to sample the posterior density, recording the description length at each step. At the end by subtracting the average description length from the entropy (calculated using the Bethe approximation) the model evidence can be evaluated (*8*). This was carried out for the configurations of the SBM recorded in Table S1.

This algorithm was run three times for each configuration to find a mean and standard deviation. Seeing as the above method returns the log-likelihood, Λ_2 could be calculated by subtracting the two values for the two hypotheses being compared and raising the answer to the base of e.

Link Prediction

Fitting a generative model to data makes a statement about the mechanisms that generated a network. Thus, in so far as the model is a good description of the data, it is possible to make generalisations and predictions about what has not been observed. This means that it is possible to use SBMs to determine the probability of a non-observed edge to be missing from the network (*12*). This can be particularly useful if a given network observation is incomplete, or contains errors or noise.

We are interested in network G, denoted by adjacency matrix $\boldsymbol{A}$ and a set of edges δG, denoted by $\delta \boldsymbol{A}$. It is assumed that the edges denoted by $\delta \boldsymbol{A}$ are missing, or spurious for that matter. When

trying to establish the probability of one of these edges to be missing the quantity of interest is the posterior of $\delta \boldsymbol{A}$, $P(\delta \boldsymbol{A}|\boldsymbol{A})$. This posterior can be written as follows (*12*, *13*):

$$P(\delta \boldsymbol{A}|\boldsymbol{A}) \propto P(\delta \boldsymbol{A}|\boldsymbol{A}+\delta \boldsymbol{A}) \sum_{\{\boldsymbol{b}_l\}} \frac{P(\boldsymbol{A}+\delta \boldsymbol{A}|\{\boldsymbol{b}_l\})}{P(\boldsymbol{A}|\{\boldsymbol{b}_l\})} P(\{\boldsymbol{b}_l\}|\boldsymbol{A})$$

In above equation $P(\boldsymbol{A}+\delta \boldsymbol{A}|\{\boldsymbol{b}_l\})$ is the marginal likelihood of the network with the potentially missing edges added and $P(\boldsymbol{A}|\{\boldsymbol{b}_l\})$ that of the observed network.

Considering alternative choices of missing edges, $\{\delta \boldsymbol{A}_i\}$, as equally likely *a priori* allows to replace $P(\delta \boldsymbol{A}|\boldsymbol{A}+\delta \boldsymbol{A})$ as simply $\propto 1$ in above equation (*12*). Thus it is possible to compute the posterior up to a normalisation constant. If, however, comparing the relative probability between a set of missing edges, $\{\delta \boldsymbol{A}_i\}$, via their likelihood ratio this constant is irrelevant (*12*):

$$\Lambda_i = \frac{P(\delta \boldsymbol{A}_i|\boldsymbol{A})}{\sum_j P(\delta \boldsymbol{A}_j|\boldsymbol{A})}$$

To do this in practice a set of edges of interest, not currently contained in the network, was defined. For these the likelihood was to be calculated. This set was denoted as "missing edges".

Having chosen an SBM configuration as outlined in the previous section, the description length of the SBM was minimised to fit the SBM to the network as a starting point. Next the MCMC algorithm implemented in graph-tool was run for 10,000 sweeps, collecting the log-likelihood of $\frac{P(\boldsymbol{A}+\delta \boldsymbol{A}|\{\boldsymbol{b}_l\})}{P(\boldsymbol{A}|\{\boldsymbol{b}_l\})}$ for each edge of interest after each iteration. The result was appended to a vector containing all previous log-likelihoods for that edge collected so far.

The gathered data was post-processed to calculate the average likelihood, in terms of log-likelihoods, across the sweeps for each edge in the samples. In order to find the denominator to calculate the likelihood ratio the log-likelihoods of all edges were then summed. Subtracting the summed log-likelihoods from the log-likelihood of the edge of interest yielded the log-likelihood ratio. Taking e to the power of the answer of those calculations finally yielded Λ_i for edge i.

Comparing the resulting likelihood ratios for each edge allows to determine how much more likely a given edge is to exist than another edge. The results of these calculations, however, are not absolute judgements on the probability of existence of a given edge but a relative one, comparing its likelihood to that of another edge, a fact that also applies to the methodologies of (*10*) and (*14*).

Testing Accuracy of Link Prediction Algorithm

In a first step the accuracy of the link prediction algorithm for the network in question was tested by screening the network introduced previously and testing whether the algorithm would be able to correctly distinguish some of the reactions discovered during a future period from random edges if presented only with past data. This is of course not a fully random test where random edges have been deleted but one that resembles the actual challenge more closely (*15*).

A list of years was taken and the network screened to the state in each of these years. First all nodes (i.e. species) that were not in the network at that point in the past were filtered out, also removing all edges attached to these nodes. All edges that were to be discovered only later were

deleted and written to a separate list. Then a random selection of these edges was chosen as a test set (taken as true positives if showing up in the result set), called "chosen true edges" for the purposes of this chapter. Then ten times as many random edges which, to date, have not been discovered (taken as false positives if showing up in the result set) were added, called "chosen false edges". The likelihood ratios of all edges in that list were found.

To do so the network without the test set was fitted to the degree-corrected, non-overlapping, nested blockmodel and the description length minimised using the function implemented in graph-tool. Then 10,000 sweeps were carried out using the MCMC algorithm implemented in graph-tool, collecting the log-likelihood of $\frac{P(\boldsymbol{A}+\delta\boldsymbol{A}|\{\boldsymbol{b}_l\})}{P(\boldsymbol{A}|\{\boldsymbol{b}_l\})}$ for each edge in the test set as outlined under "Link Prediction". The log-likelihoods were then averaged and the likelihood ratios for the different edges found.

Upon ranking the edges in order of decreasing likelihood ratio magnitudes it was possible to evaluate the performance of the algorithm using, for example, Receiver Operating Characteristic (ROC) curves. An ROC plot plots the true positive rate against the false positive rate (i.e. benefits vs. costs) (*16*). This is done by ranking the edges in order of decreasing likelihood ratios. Working downwards through the list in the direction of decreasing likelihood ratio, all edges of greater likelihood ratio than the current cut-off are taken to be the result set. At each point the true positive rate is found by dividing the number of edges from "chosen true edges" that are found in the result set by the total number of "chosen true edges". Similarly, the false positive rate is calculated by dividing the number of edges in "chosen false edges" discovered in the result set by the total number of edges in "chosen false edges". Thus, a set of coordinates, (fp_rate, tp_rate), was obtained and plotted on the axes.

This procedure illustrates what share of incorrect predictions has to be accepted in order to discover a given share of true positives contained in the set of tested edges. If edges are picked at random from the entire test set this results in an ROC curve following the $x = y$ line, thus giving a convenient graphical comparison.

The area under the curve (AUC) combines the results of this procedure into a single figure along which to compare performance. This, conveniently, also equates to the probability that the methodology ranks a randomly chosen positive instance higher than a randomly chosen negative instance (*16*, *17*).

The assumption that all "random" edges in the top scoring section of the result set are truly false positives is not technically correct. It would be possible that the reason they are scoring highly is because they are indeed missing edges. As a consequence, the improvement over chance determined by the algorithm is thus, technically, a lower boundary on the performance of the algorithm.

Though calculating and plotting an ROC curve reveals information about how the algorithm performs for different thresholds it does not necessarily answer the question of where the best threshold lies. One way of finding this point is by calculating the accuracy, which is defined as follows (*16*):

$$accuracy = \frac{number\ of\ true\ positives + number\ of\ true\ negatives}{number\ of\ positives + number\ of\ negative}$$

This is the number of correct classifications divided by the total number of classifications. As already outlined for calculating the ROC it is straightforward to calculate the number of true positives. The number of true negatives was taken to be the number of edges from the list "chosen false edges" that were found in the set of edges with a likelihood ratio lower than the chosen threshold. It is possible to calculate the accuracy, for a range of thresholds, and compare which threshold returns the highest accuracy, assuming that everything above the threshold is classified as a positive and everything beneath it as a negative. In this work the impact on the accuracy of taking progressively larger parts of the results (working in decreasing order of likelihood ratio) as positives was tested.

In order to further evaluate the performance of the algorithm it was compared against a number of other, more straightforward, metrics. One way of approaching link prediction is to determine the similarity between two nodes. If they are sufficiently similar they are deemed to be likely to have a link between them. What will be used here are local structural similarity indices.

The first metric used was the Jaccard index, which is the number of common neighbours shared between two nodes divided by the size of the set of all neighbours of the two nodes (*18*):

$$s_{xy}^{Jaccard} = \frac{|\Gamma(x) \cap \Gamma(y)|}{|\Gamma(x) \cup \Gamma(y)|}$$

where s_{xy} is the similarity score between vertex x and y and $\Gamma(x)$ is the set of neighbours of vertex x and $\Gamma(y)$ that of vertex y, though the implementation used by graph-tool only considers the out-neighbours.

Second, the Adamic-Adar Index was calculated, which gives the sum of weights of common neighbours of two vertices calculated as follows (*18*, *19*):

$$s_{xy}^{AA} = \sum_{z \in \Gamma(x) \cap \Gamma(y)} \frac{1}{\log k_z}$$

where the implementation used by graph-tool only considers the out-neighbours and thus uses the in-degrees.

Following the example of (*10*) the shortest path similarity was also calculated, which was simply taken to be one over the length of the shortest path between two vertices (and zero if no path exists) (*20*). Clauset *et al.* also compare their model to the number of common neighbours, very similar to the Jaccard index, defined as follows (*18–20*):

$$s_{xy}^{CN} = |\Gamma(x) \cap \Gamma(y)|$$

A final metric used was the degree product. This is based on the fact that scale-free networks are thought to grow via preferential attachment, where the probability of a vertex connecting to another given vertex is proportional to its degree (*21*, *22*). Unlike the models to which this theory was first applied the network here is directed. Hence the metric was slightly modified. As Mislove *et al.* point out, in a directed network preferential attachment subdivides into preferential creation (the probability to create a new link which is proportional to a node's out-

degree) and preferential inception (the probability to receive a new link which is proportional to the node's in-degree) (*23*). The form used in (*10*) was modified to yield:

$$s_{xy}^{PA} = k_{out_x} \times k_{in_y}$$

This measures the similarity if vertex x is the source of the edge and vertex y the target.

Each of these metrics was calculated for each proposed edge in the test set for each of the years used during the SBM testing. It was then assumed that the edges in the test set scoring highest on the relevant similarity metrics were most probable to represent missing edges. The procedure was repeated four times and the mean of the values was plotted with the standard deviation taken as estimate of the error.

Prediction of Links

For the link prediction an up-to-date network was used to minimise the rediscovery of already known reactions. This meant that the size of the overall network had to be reduced to be able to process it given the available computational resources. The same network as used during the testing of the accuracy of the link prediction algorithm was used, however, instead of adopting an "all-to-all" wiring scheme it was instead decided to adopt a scheme registering and connecting only the heaviest reactant to the heaviest product, as already discussed in (*24*) which resulted in a network of 161,760 chemical species and 132,539 edges, each representing a reaction record in Reaxys.

It was decided to investigate the routes from limonene to paracetamol more closely to see if any useful reactions could be predicted in this area of chemistry. For this section of the work usefulness was defined as a reaction which reduces the number of steps taken to carry out the conversion of limonene to paracetamol, i.e. one that provides a "short cut". Alternatively also accepted was a prediction that leads to an increase in the number of efficient paths available to carry out the synthesis, i.e. one that opens up chemical space, providing more options. Seeing as the previous chapter worked on the assumption that five steps were required to carry out the synthesis, a more efficient path was taken to be a path with a length of four or less steps.

To implement this, a path search was carried out in graph-tool, finding all paths of a length of less than or equal to four steps. Having obtained this list of paths the vertices, i.e. chemical species, lying on these paths were extracted.

Though it is possible that there exists a molecule which currently does not lie on any of the paths, and which would provide a more efficient route if suitable reactions to it were found, identifying such a molecule raises the complexity of the problem at hand significantly. Hence it was decided to limit the search to reactions between the molecules currently lying on the paths.

Generating the list of reactions which meet the requirements of usefulness defined above is straightforward: all that is required is to generate the set of all possible pair-wise combinations of the molecules on the existing four step paths connecting limonene to paracetamol (considering the pair (a,b) to be distinct from (b,a)) taking the first node in each pair as source and the second as target of an edge.

This was carried out for the set of vertices at hand. Self-loops were removed from the list (i.e. an edge where a given vertex is both source and target) and any edges already existing in the network were also removed from the set of edges of interest.

The generated list of edges of interest was then fed to the link prediction algorithm parametrised as described in the previous section on testing the accuracy of the link prediction algorithm and the same procedure was followed to obtain the likelihood ratios before ranking the edges in order of decreasing magnitudes of the likelihood ratios.

The paracetamol case study presented in the previous chapter applied the rule that the intermediate being converted into a new intermediate had to contain precisely one ring structure before and after the reaction. Thus, for reasons of consistency, this was applied to the result set of the link prediction too and any reactions where this was not the case were screened out.

The reduced set of predictions was analysed to determine what improvement in the number of four step routes each reaction would result in. Each edge was added to the network, the change in the number of four step paths leading from limonene to paracetamol was recorded, and the edge was deleted from the network again before the next edge was added. Any edge which created at least one additional four step path was retained in the result set and the remainder was screened out.

Finally, the remaining result set was analysed to remove any reactions contained in Reaxys but absent from this subnetwork by comparing it to the much larger network used by us in (*25*) so that the set only contained novel transformations.

Next, the exact procedure was repeated but the search generating the test set was trying to find all five-step paths (though still analysing how many new four-step paths would be added by the prediction). This was done in order to get a larger sample set and a greater chance of it containing true positives.

In addition, a possible change in Gibbs free energy due to the transformation occurring was calculated for the three highest-ranking transformations. The reaction equations were balanced as outlined in (*26*) by using small molecules that are common byproducts of syntheses such as water, carbon dioxide, etc as also outlined in (*27*, *28*). The Gibbs energy of reaction was calculated by Mr Yehia Amar (group member) in Gaussian 09 with geometry minimisation using ωB97XD level of theory and a cc-PVDZ basis set followed by frequency calculation with the same level of theory to confirm that the found minimum is the true minimum. Single point energy calculation was then carried out using the same level of theory and a basis set of cc-PVTZ assuming no solvation. Clearly the Gibbs free energy of reaction assuming the molecules are in a vacuum will not yield the same answer as when calculating it in a solvent. Due to the very limited amount of data available on these reaction suggestions however there is no solvent information available. As the calculation is only a rough estimate and liable to change under the ultimate conditions in the laboratory the added uncertainty due to the calculation being carried out without solvent is taken to be acceptable.

Supplementary Text

Analysis of Alternative Stochastic Block Models

Finding the MDL for the various SBM configurations yields the results shown in Table S2.

To visualise the results Λ_1 was plotted for the different comparisons in Figure S1. Λ_1 was calculated such that the preferred option has a value of 1.0.

Based on the MDL data, the degree corrected, non-overlapping, nested SBM provides the best possible fit. Looking at the values of Λ_1 in Figure S1 there is a clear, statistically significant preference towards this configuration. Though one should be wary of applying a hard and fast confidence threshold, a line at $\Lambda_1 = 0.01$ has been drawn onto the plot as this is a rule-of-thumb often used to determine whether a preference is statistically significant or not (*7*). From this it is apparent that the preference over the non-nested, degree-corrected, non-overlapping SBM might not be statistically significant.

Having minimised the description length, the posterior distribution was subsequently sampled for each of the possible configurations across 200,000 iterations. The results of this are shown in Table S3.

The results indicate that the non-nested, degree corrected, overlapping SBM provides the best fit to the data, followed by the nested version of the same. Based on the data in Table S3 the logarithm of the posterior odds ratio was calculated and plotted in Figure S2 to evaluate the significance and strength of the preference. Given that it provided the partition with the best fit to the data the degree of preference towards the nested version was tested here.

As can be seen in Figure S2 the nested version of the non-overlapping, degree corrected SBM has an almost as strong, statistically significant preference over all other models as the non-nested version of the same (though of course the non-nested version's statistical preference over the nested version remains).

For the purposes of this study it was decided to use the nested version of the non-overlapping, degree corrected SBM. Scope however clearly exists to explore how the non-nested, non-overlapping, degree corrected SBM would perform in link prediction for this data set.

Supplementary References

1. A. A. Lapkin *et al.*, Automation of route identification and optimisation based on data-mining and chemical intuition. *Faraday Discuss.* **202**, 483–496 (2017).
2. T. P. Peixoto, The graph-tool python library. *figshare* (2014), doi:10.6084/m9.figshare.1164194.
3. T. P. Peixoto, Parsimonious Module Inference in Large Networks. *Phys. Rev. Lett.* **110**, 148701 (2013).
4. T. P. Peixoto, Hierarchical Block Structures and High-Resolution Model Selection in Large Networks. *Phys. Rev. X*. **4**, 11047 (2014).
5. T. P. Peixoto, Efficient Monte Carlo and greedy heuristic for the inference of stochastic block models. *Phys. Rev. E*. **89**, 12804 (2014).
6. T. P. Peixoto, Model Selection and Hypothesis Testing for Large-Scale Network Models

with Overlapping Groups. *Phys. Rev. X*. **5**, 11033 (2015).
7. T. P. Peixoto, Inferring the mesoscale structure of layered, edge-valued, and time-varying networks. *Phys. Rev. E*. **92**, 42807 (2015).
8. T. P. Peixoto, Nonparametric Bayesian inference of the microcanonical stochastic block model. *Phys. Rev. E*. **95**, 12317 (2017).
9. M. Rosvall, C. T. Bergstrom, An information-theoretic framework for resolving community structure in complex networks. *Proc. Natl. Acad. Sci.* **104**, 7327–7331 (2007).
10. A. Clauset, C. Moore, M. E. J. Newman, Hierarchical structure and the prediction of missing links in networks. *Nature*. **453**, 98–101 (2008).
11. X. Yan *et al.*, Model selection for degree-corrected block models. *J. Stat. Mech. Theory Exp.* **2014**, P05007 (2014).
12. T. P. Peixoto, in *Advances in Network Clustering and Blockmodeling*, P. Doreian, V. Batagelj, A. Ferligoj, Eds. (Wiley-VCH Verlag GmbH & Co. KGaA, New York, 2019; http://arxiv.org/abs/1705.10225).
13. T. Vallès-Català, T. P. Peixoto, R. Guimerà, M. Sales-Pardo, On the consistency between model selection and link prediction in networks (2017) (available at http://arxiv.org/abs/1705.07967).
14. R. Guimera, M. Sales-Pardo, Missing and spurious interactions and the reconstruction of complex networks. *Proc. Natl. Acad. Sci.* **106**, 22073–22078 (2009).
15. M. H. S. Segler, M. P. Waller, Modelling Chemical Reasoning to Predict and Invent Reactions. *Chem. - A Eur. J.* **23**, 6118–6128 (2017).
16. T. Fawcett, An introduction to ROC analysis. *Pattern Recognit. Lett.* **27**, 861–874 (2006).
17. S. Agarwal, D. Dugar, S. Sengupta, Ranking Chemical Structures for Drug Discovery: A New Machine Learning Approach. *J. Chem. Inf. Model.* **50**, 716–731 (2010).
18. L. Lü, T. Zhou, Link prediction in complex networks: A survey. *Phys. A Stat. Mech. its Appl.* **390**, 1150–1170 (2011).
19. L. Lü, L. Pan, T. Zhou, Y.-C. Zhang, H. E. Stanley, Toward link predictability of complex networks. *Proc. Natl. Acad. Sci.* **112**, 2325–2330 (2015).
20. A. Clauset, C. Moore, M. E. J. Newman, Hierarchical structure and the prediction of missing links in networks - Supplementary information. *Nature*. **453**, 98–101 (2008).
21. A. Vázquez, Growing network with local rules: Preferential attachment, clustering hierarchy, and degree correlations. *Phys. Rev. E*. **67**, 56104 (2003).
22. A.-L. Barabási, R. Albert, Emergence of Scaling in Random Networks. *Science*. **286**, 509–512 (1999).
23. A. Mislove, H. S. Koppula, K. P. Gummadi, P. Druschel, B. Bhattacharjee, in *Proceedings of the first workshop on Online social networks - WOSP '08* (ACM Press, New York, New York, USA, 2008; http://portal.acm.org/citation.cfm?doid=1397735.1397742), p. 25.
24. M. Fialkowski, K. J. M. Bishop, V. A. Chubukov, C. J. Campbell, B. A. Grzybowski, Architecture and Evolution of Organic Chemistry. *Angew. Chemie Int. Ed.* **44**, 7263–7269 (2005).
25. P.-M. Jacob, A. Lapkin, Statistics of the network of organic chemistry. *React. Chem. Eng.* **3**, 102–118 (2018).
26. P.-M. Jacob, P. Yamin, C. Perez-Storey, M. Hopgood, A. A. Lapkin, Towards automation of chemical process route selection based on data mining. *Green Chem.* **19**, 140–152 (2017).

27. A. Dengler, E. Fontain, M. Knauer, N. Stein, I. Ugi, Competing concepts in CAOS. *Recl. des Trav. Chim. des Pays-Bas*. **111**, 262–269 (1992).
28. I. Ugi, A. Dengler, The algebraic and graph theoretical completion of truncated reaction equations. *J. Math. Chem.* **9**, 1–10 (1992).

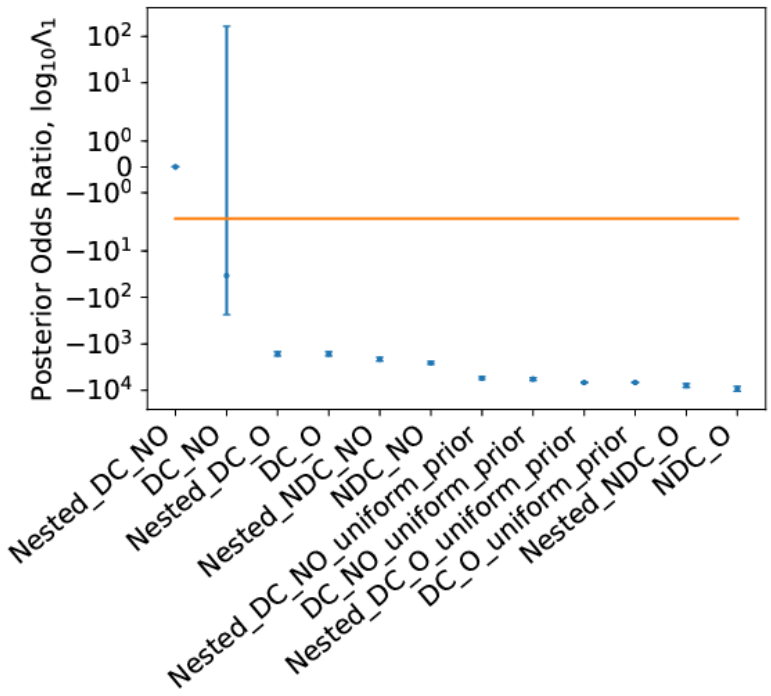

Fig. S1. Posterior odds ratio relative to the best model according the MDL criterion. The solid line represents $\Lambda_1 = 10^{-2}$.

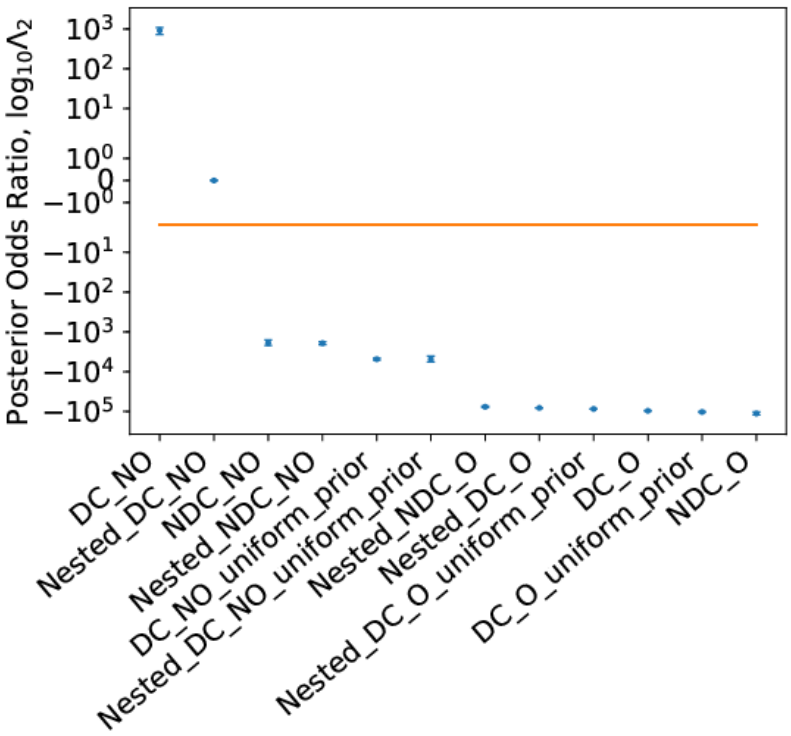

Fig. S2. Posterior odds ratio upon sampling the posterior distribution between the given configuration and (the nested, non-overlapping, degree corrected version of the SBM. The solid line represents $\Lambda_2 = 10^{-2}$.

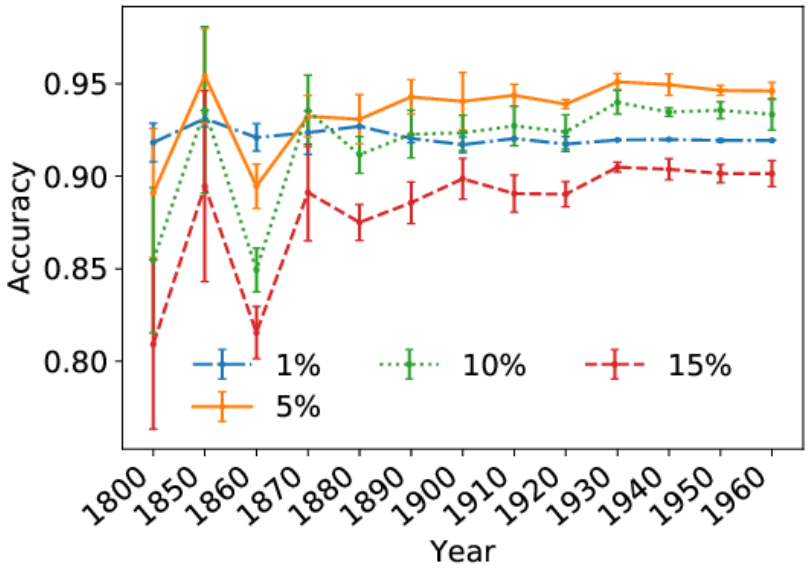

Fig. S3. The accuracy of the algorithm calculated at a number of points in time. Each line shows the accuracy when taking the corresponding top-x% of results, when ranked according to decreasing likelihood ratio, as positives. The error bars were obtained by taking the standard deviation across four repeats of the measurements.

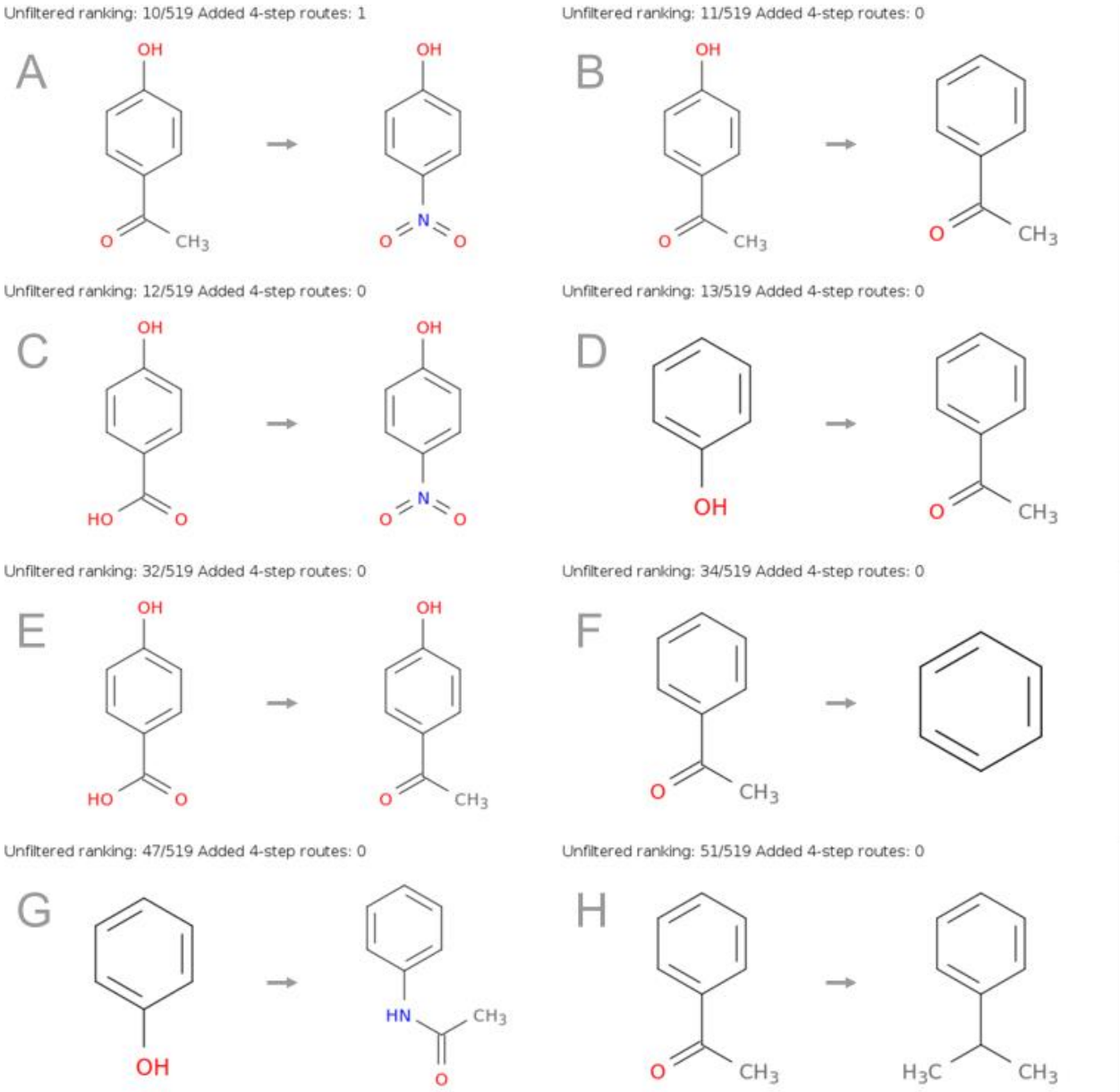

Fig. S4. A selection of reaction predictions considered most likely by the algorithm that turned out to already be known when comparing it to a larger section of Reaxys. Due to the heaviest-to-heaviest wiring scheme implemented only the main reactant and product are shown.

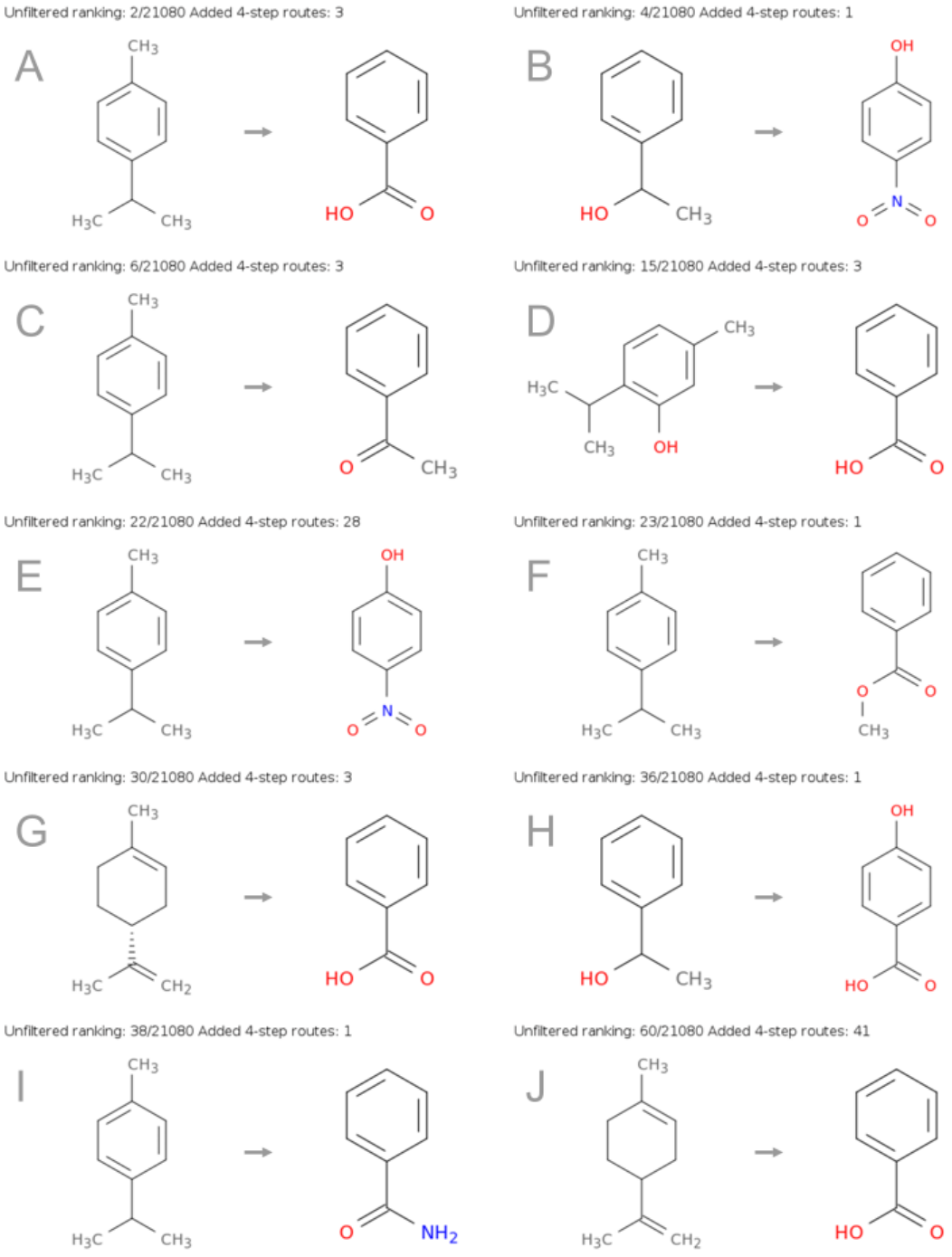

Fig. S5. A selection of transformations, ranked by the algorithm, that were identified as novel and increasing the number of four-step-paths showing transformations 1-10 in decreasing order of likelihood ratio magnitude.

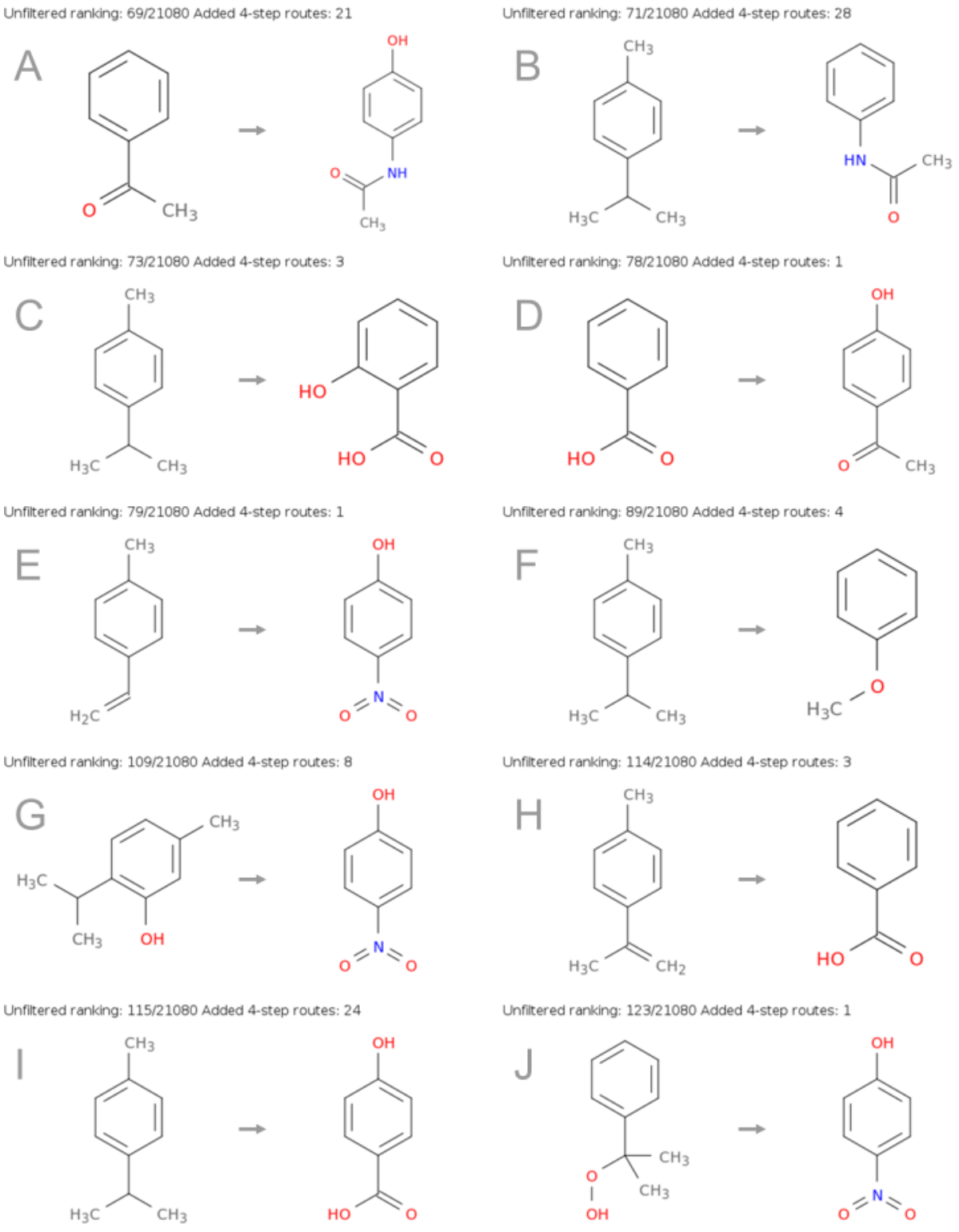

Fig. S6. A selection of transformations, ranked by the algorithm, that were identified as novel and increasing the number of four-step-paths showing transformations 11-20 in decreasing order of likelihood ratio magnitude.

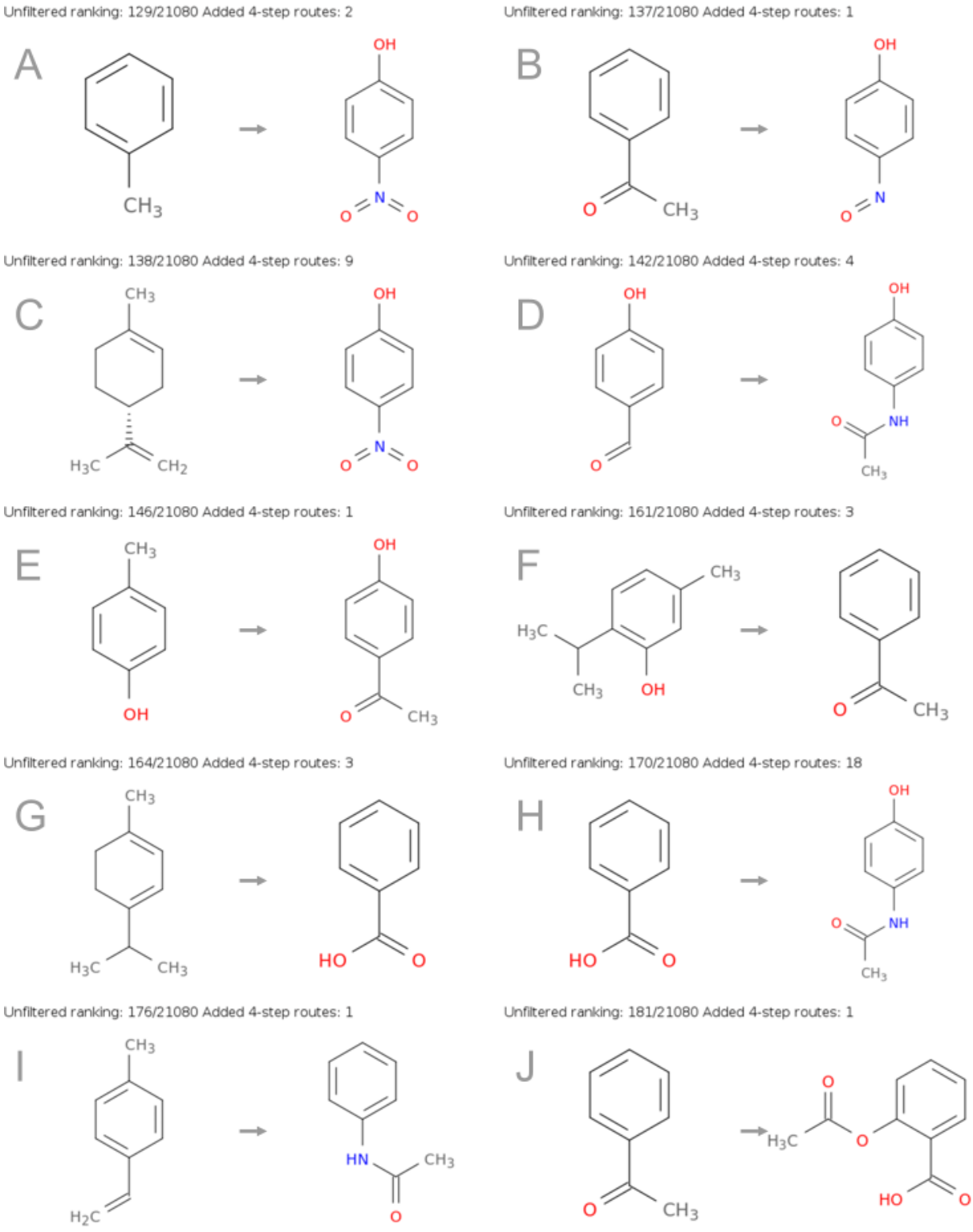

Fig. S7. A selection of transformations, ranked by the algorithm, that were identified as novel and increasing the number of four-step-paths showing transformations 21-30 in decreasing order of likelihood ratio magnitude.

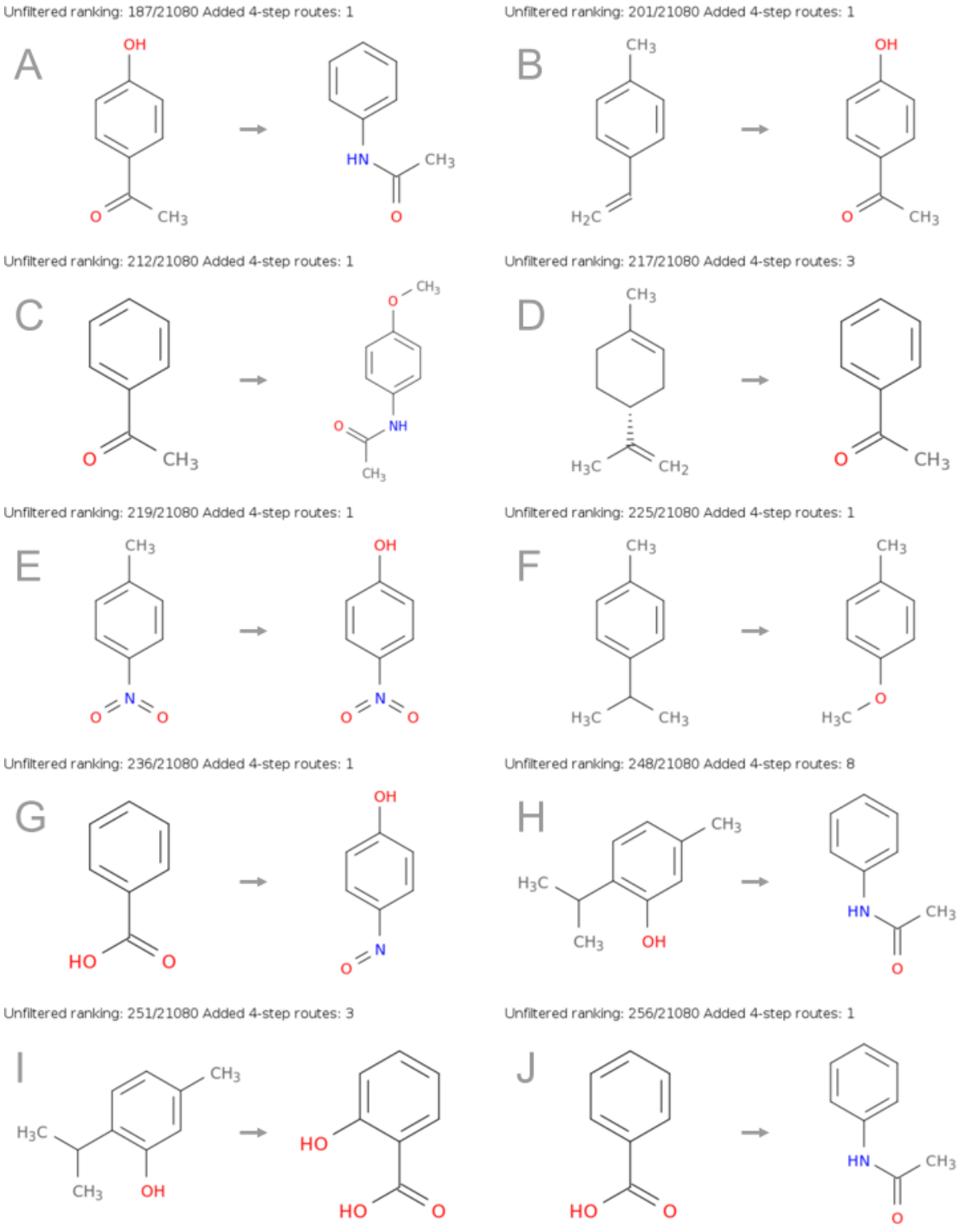

Fig. S8. A selection of transformations, ranked by the algorithm, that were identified as novel and increasing the number of four-step-paths showing transformations 31-40 in decreasing order of likelihood ratio magnitude.

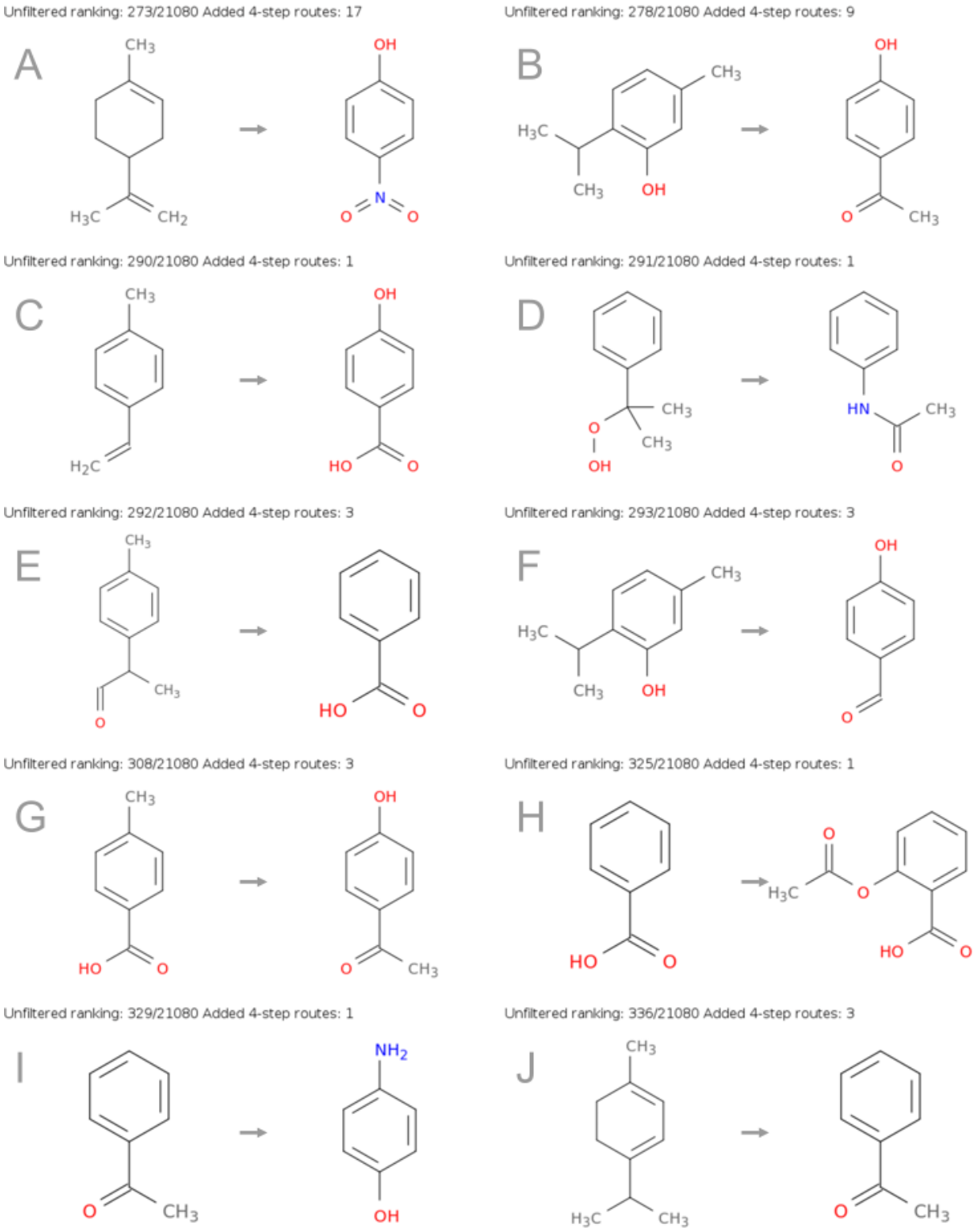

Fig. S9. A selection of transformations, ranked by the algorithm, that were identified as novel and increasing the number of four-step-paths showing transformations 41-50 in decreasing order of likelihood ratio magnitude.

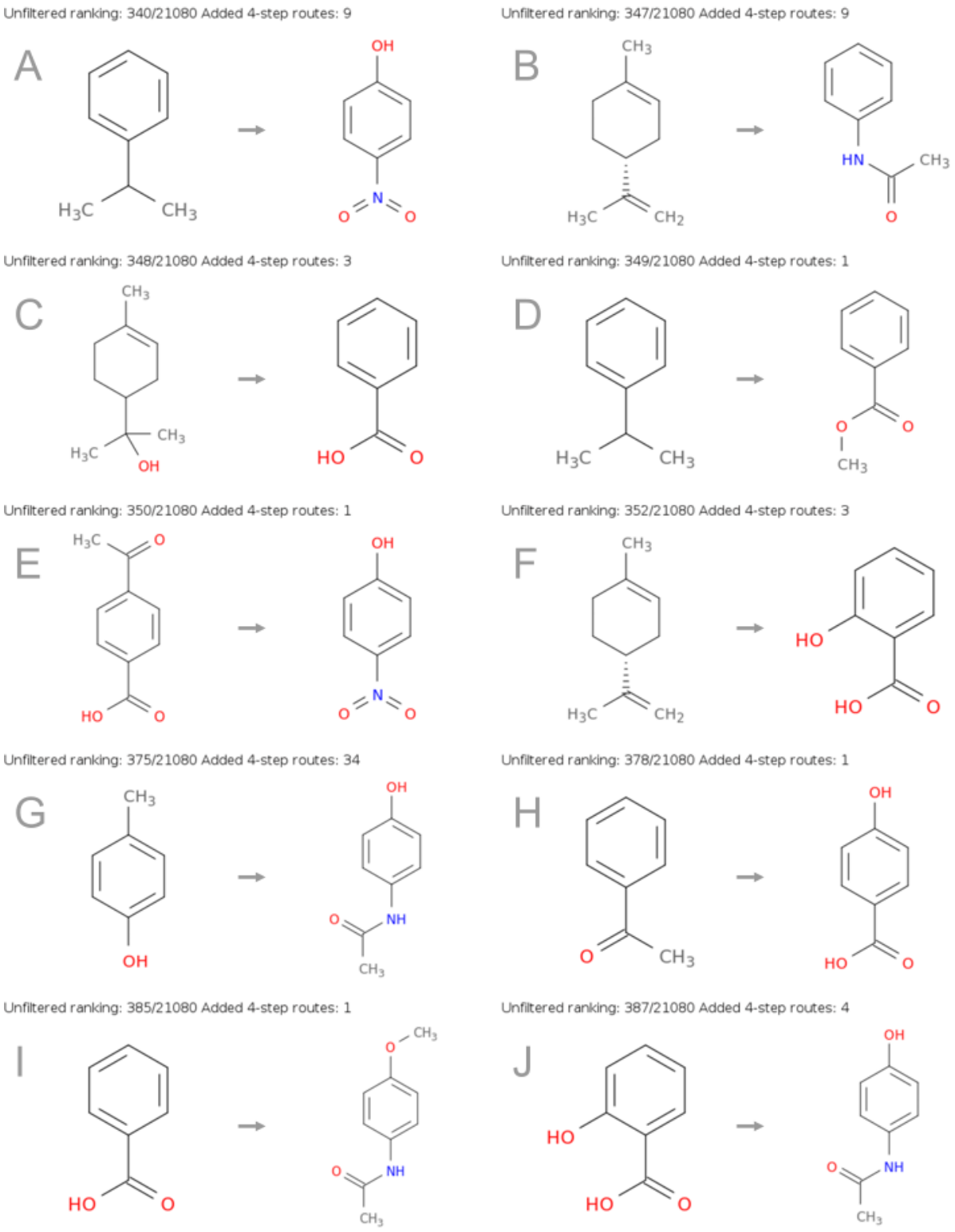

Fig. S10. A selection of transformations, ranked by the algorithm, that were identified as novel and increasing the number of four-step-paths showing transformations 51-60 in decreasing order of likelihood ratio magnitude.

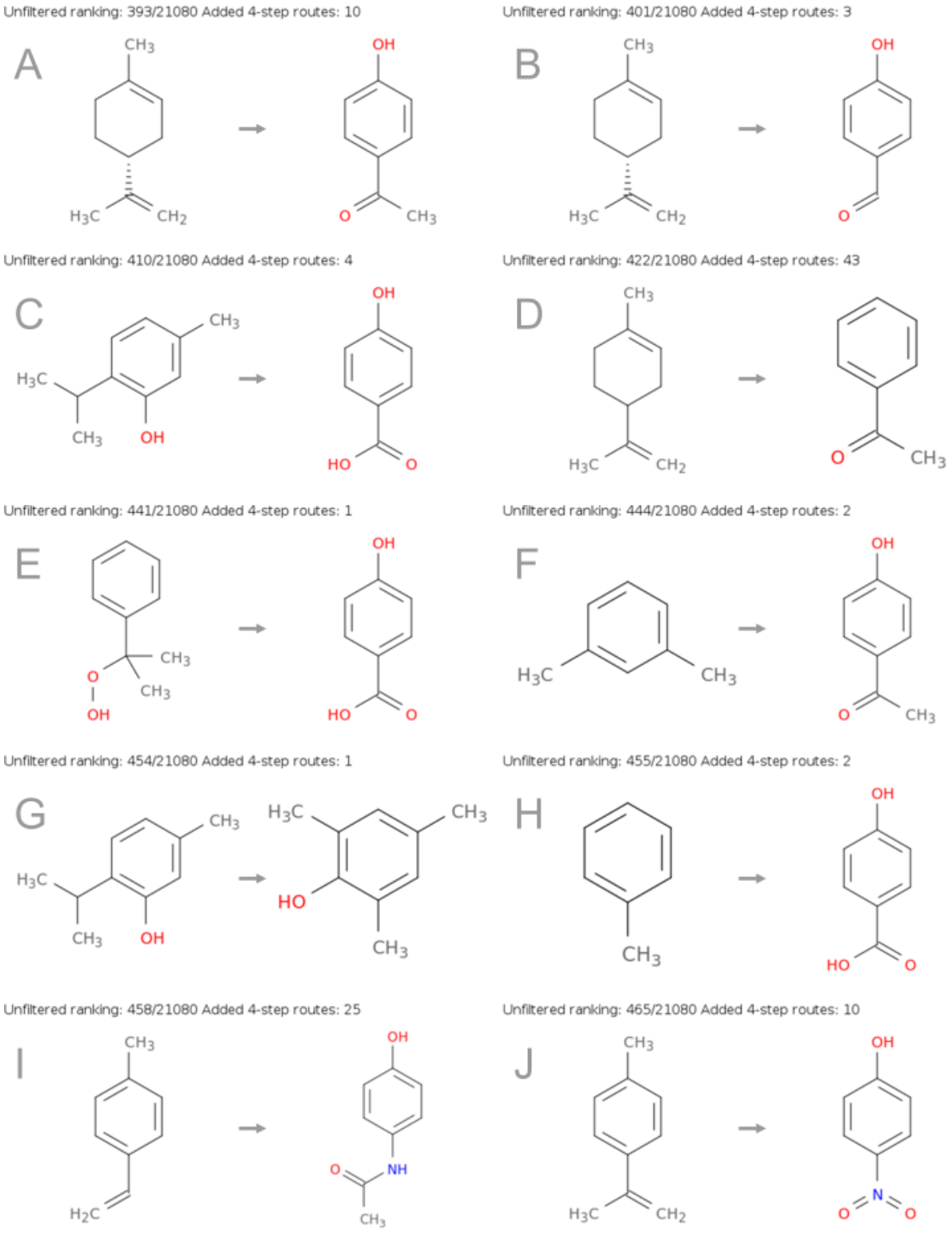

Fig. S11. A selection of transformations, ranked by the algorithm, that were identified as novel and increasing the number of four-step-paths showing transformations 61-70 in decreasing order of likelihood ratio magnitude.

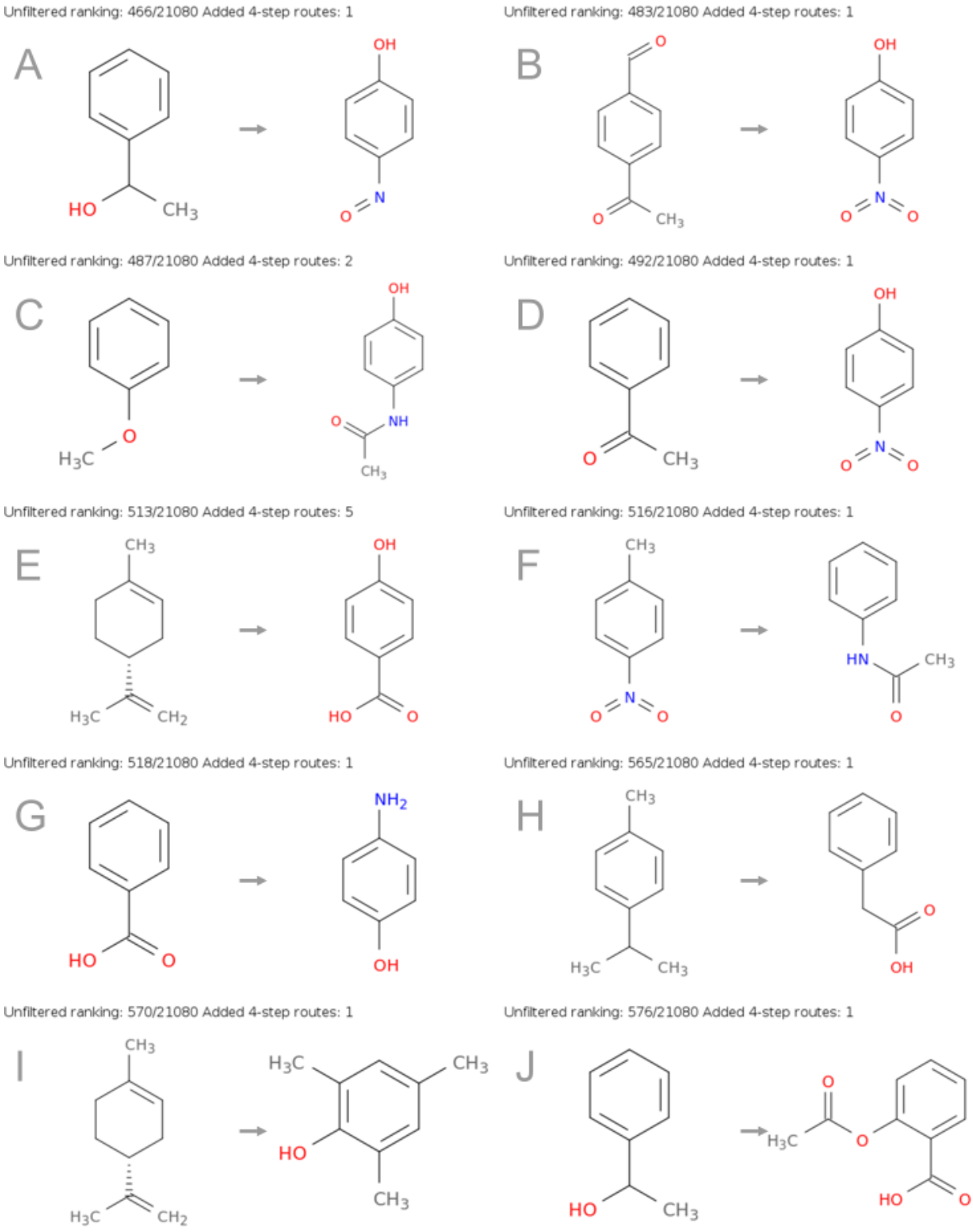

Fig. S12. A selection of transformations, ranked by the algorithm, that were identified as novel and increasing the number of four-step-paths showing transformations 71-80 in decreasing order of likelihood ratio magnitude.

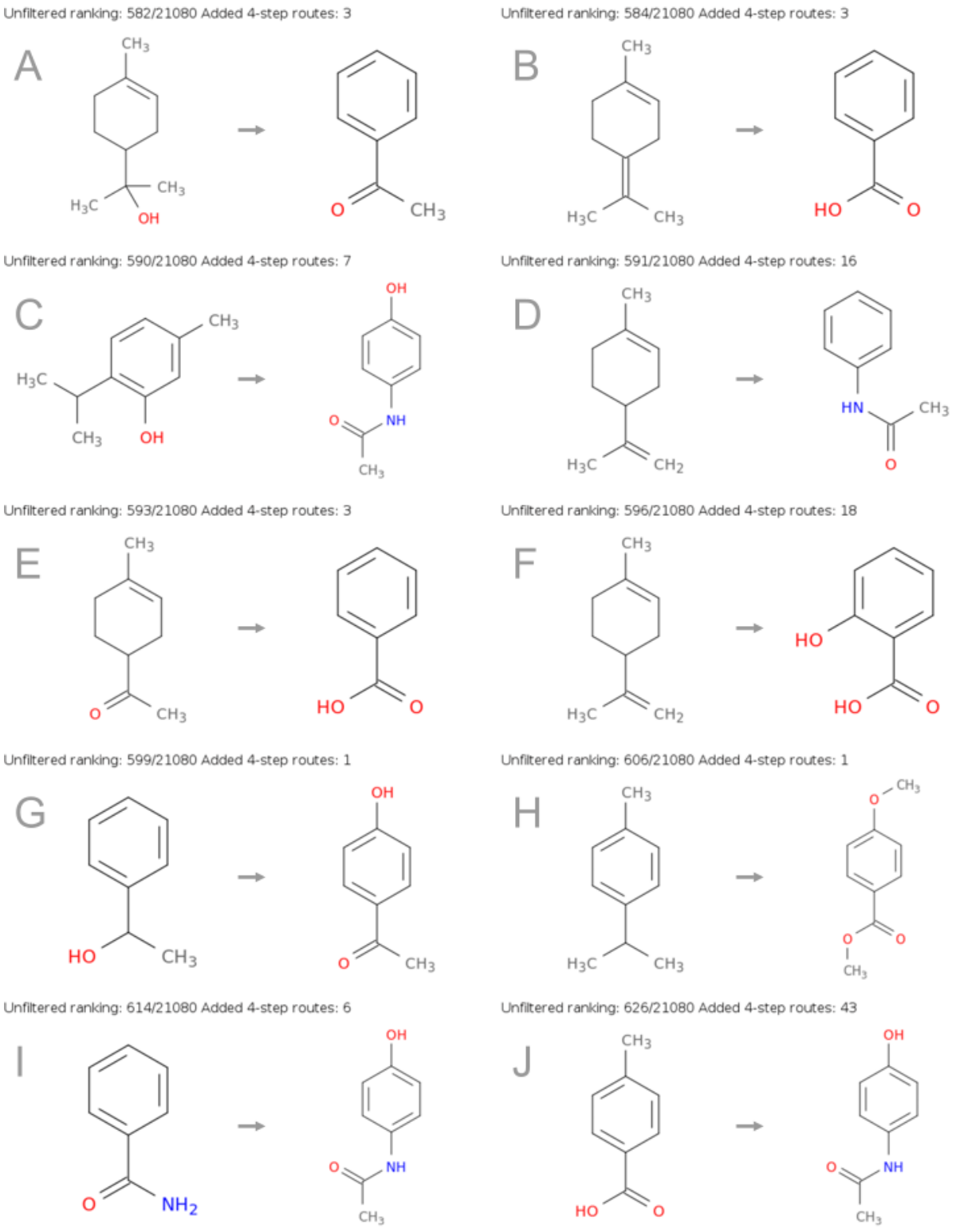

Fig. S13. A selection of transformations, ranked by the algorithm, that were identified as novel and increasing the number of four-step-paths showing transformations 81-90 in decreasing order of likelihood ratio magnitude.

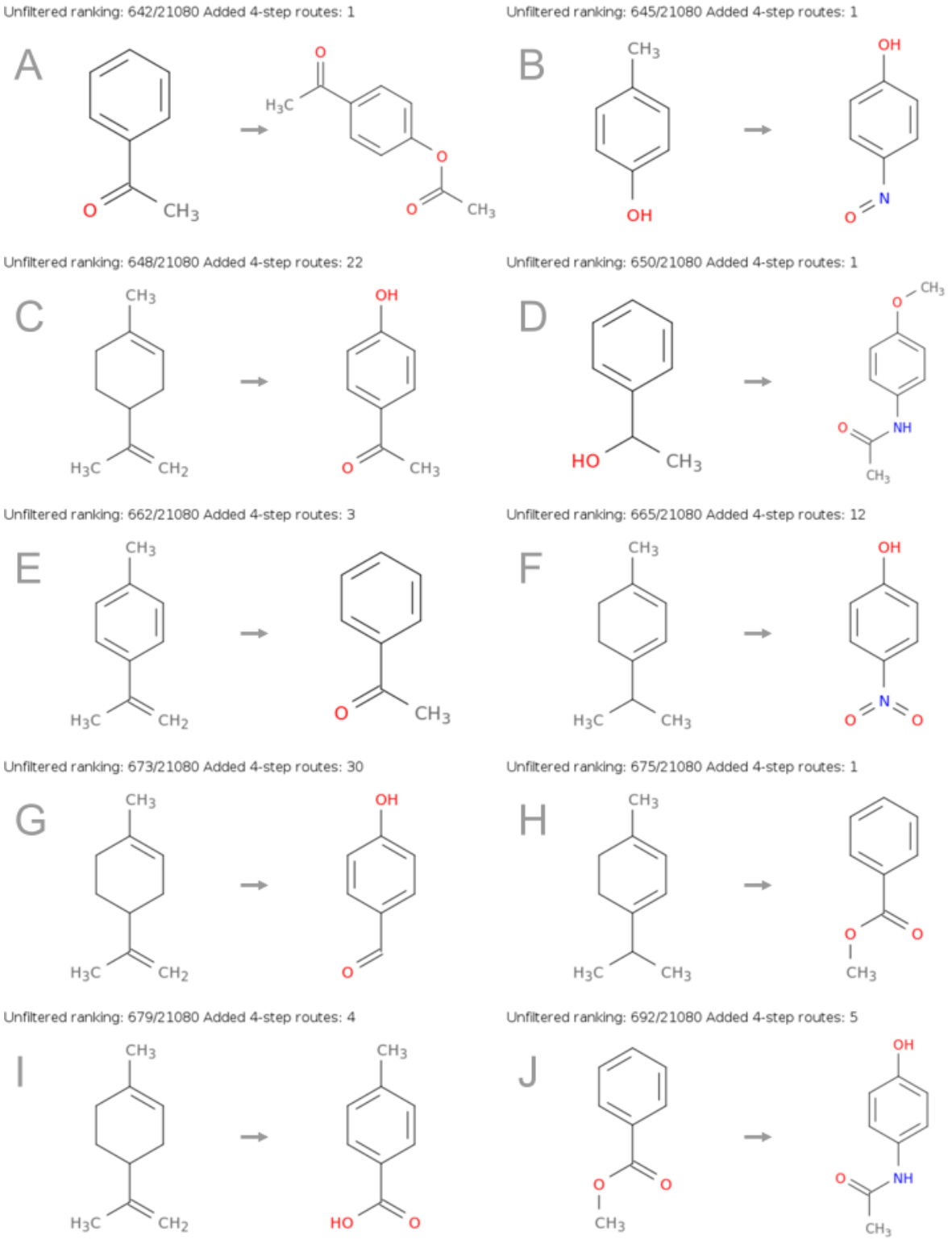

Fig. S14. A selection of transformations, ranked by the algorithm, that were identified as novel and increasing the number of four-step-paths showing transformations 91-100 in decreasing order of likelihood ratio magnitude.

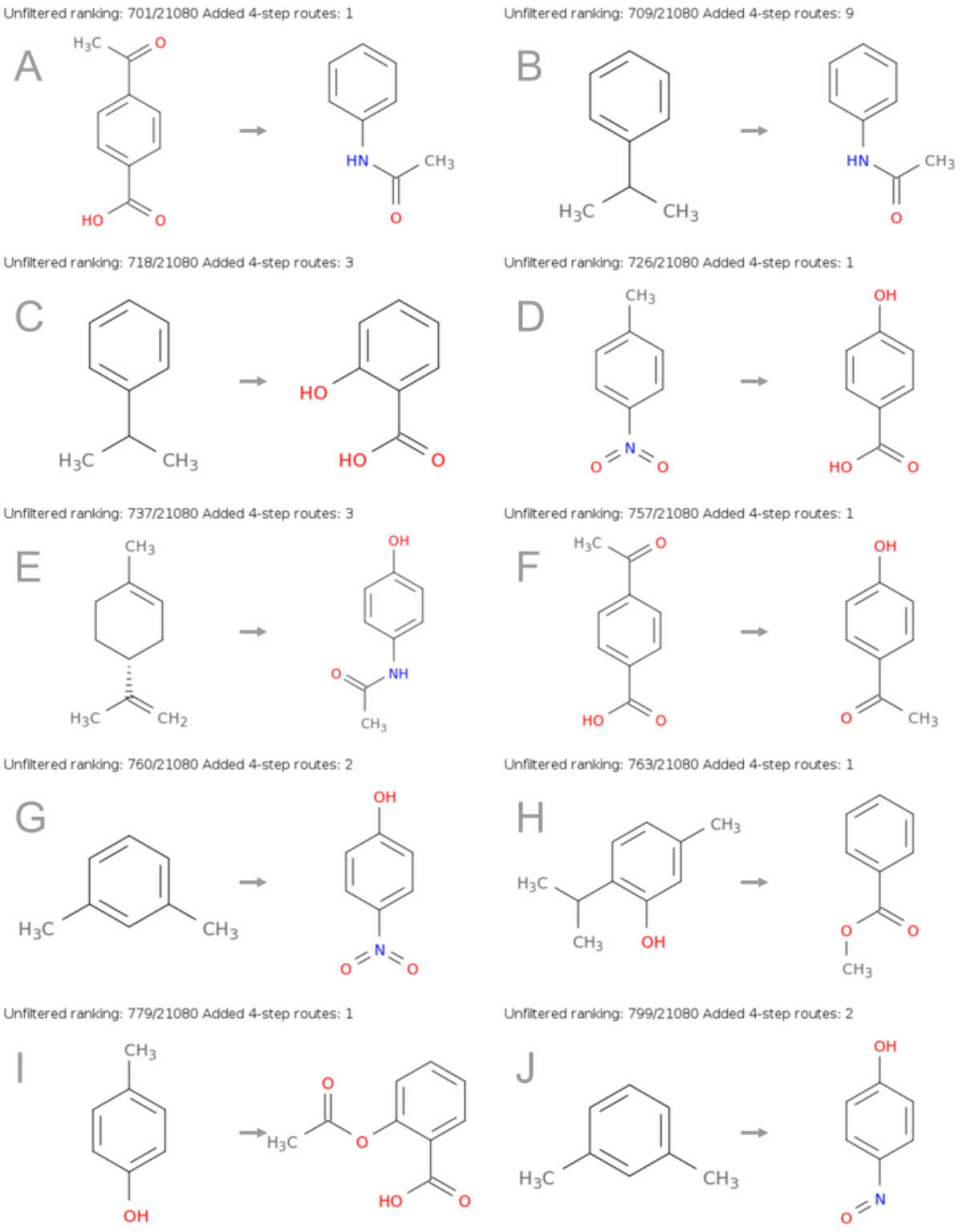

Fig. S15. A selection of transformations, ranked by the algorithm, that were identified as novel and increasing the number of four-step-paths showing transformations 101-110 in decreasing order of likelihood ratio magnitude.

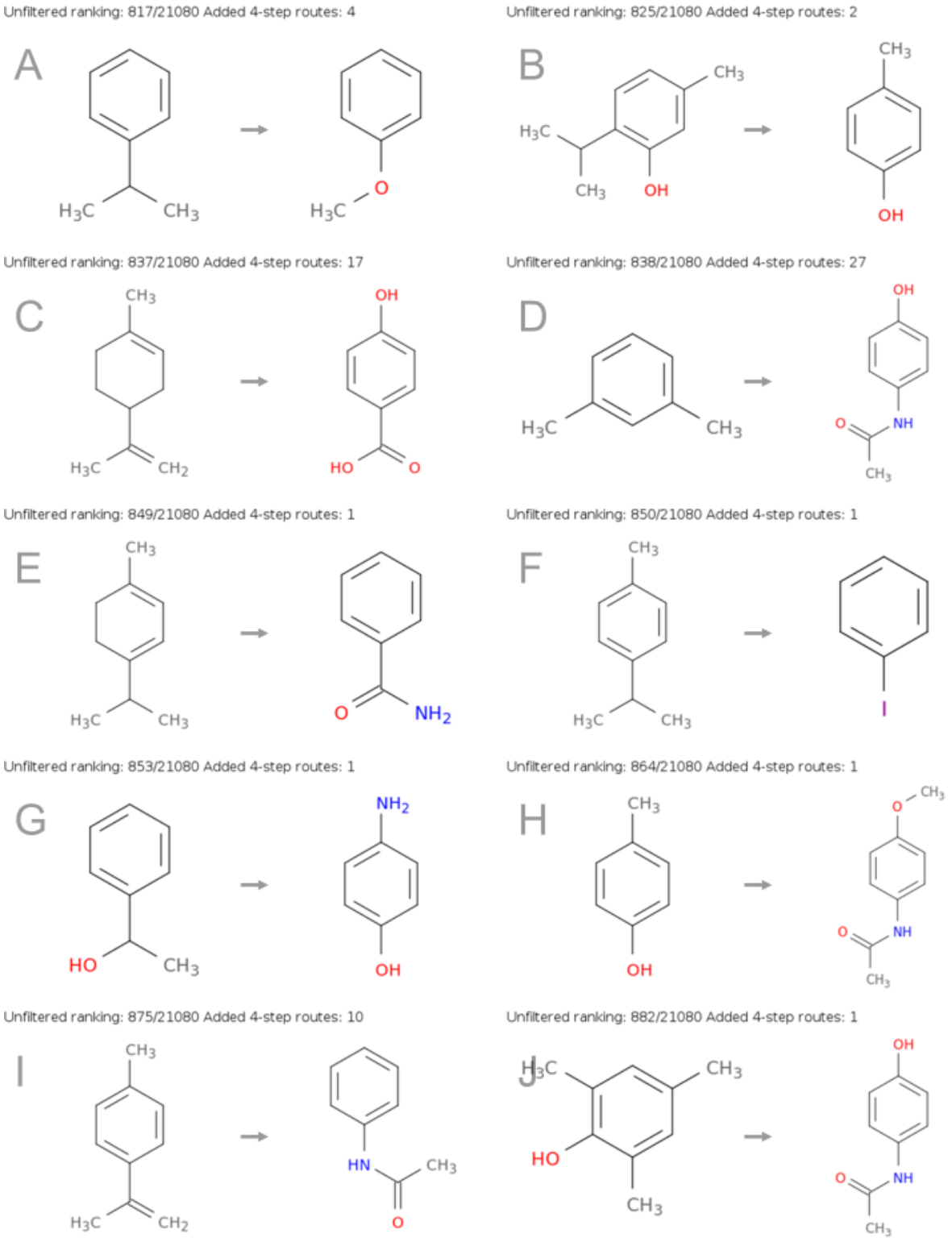

Fig. S16. A selection of transformations, ranked by the algorithm, that were identified as novel and increasing the number of four-step-paths showing transformations 111-120 in decreasing order of likelihood ratio magnitude.

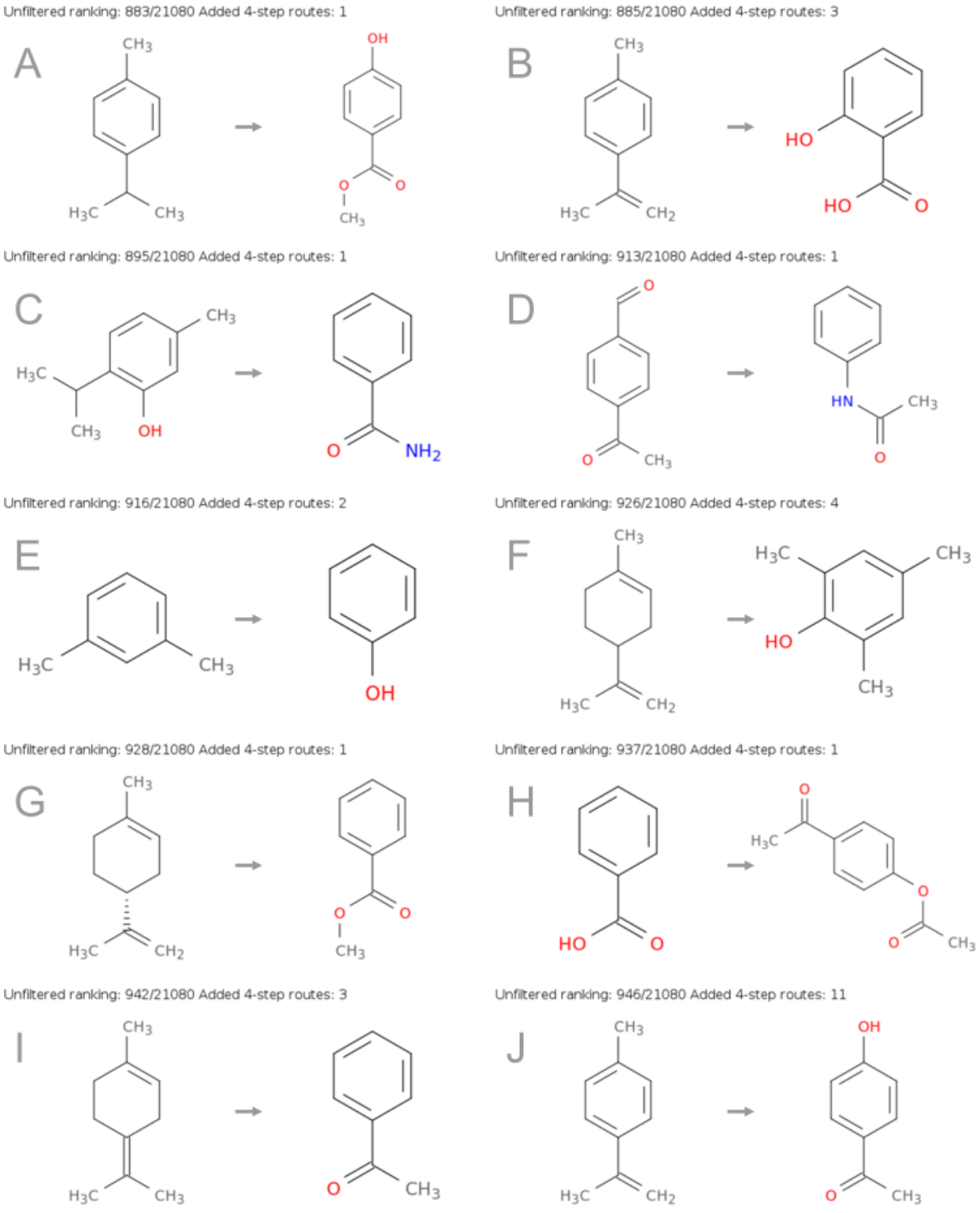

Fig. S17. A selection of transformations, ranked by the algorithm, that were identified as novel and increasing the number of four-step-paths showing transformations 121-130 in decreasing order of likelihood ratio magnitude.

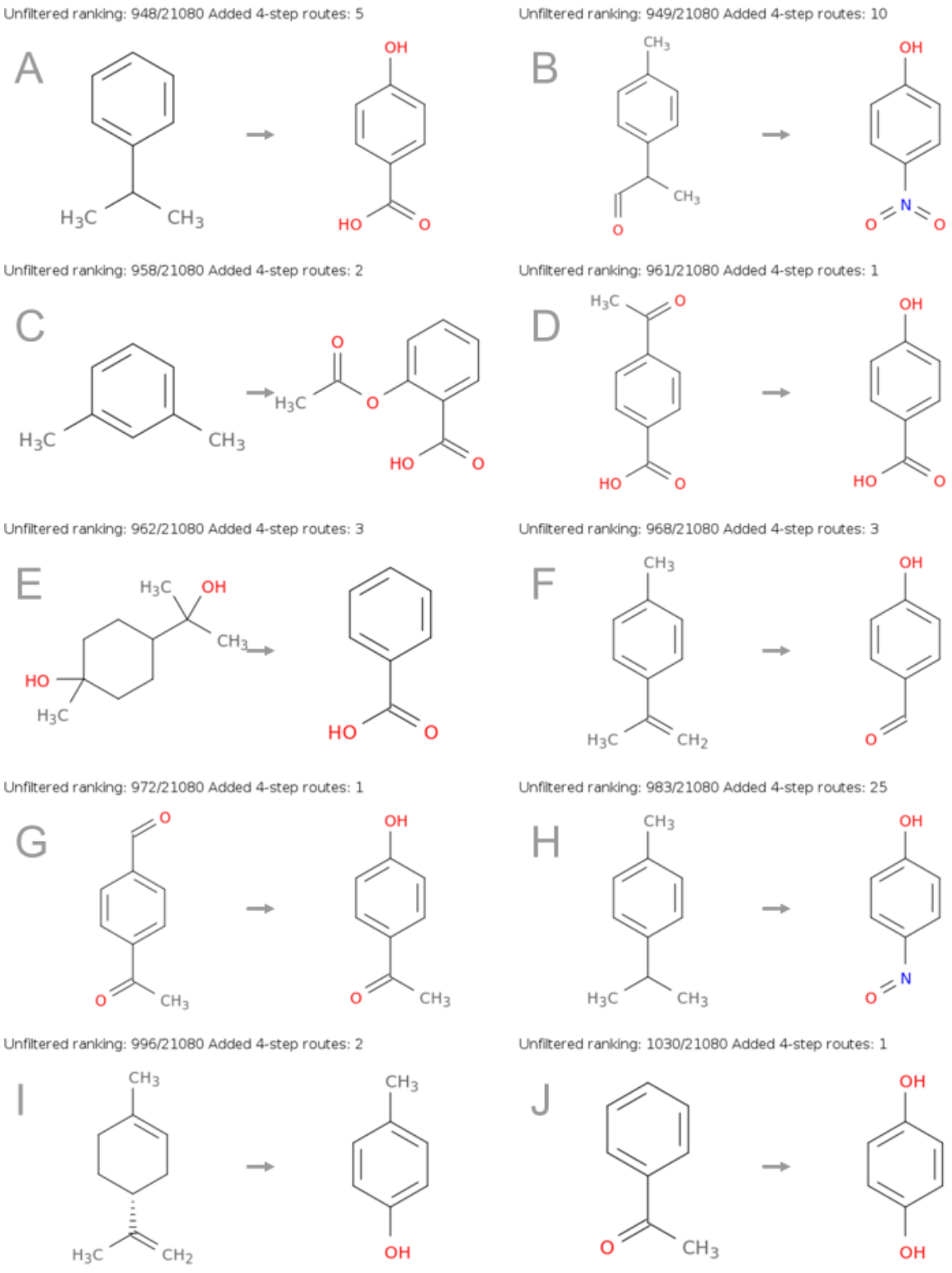

Fig. S18. A selection of transformations, ranked by the algorithm, that were identified as novel and increasing the number of four-step-paths showing transformations 131-140 in decreasing order of likelihood ratio magnitude.

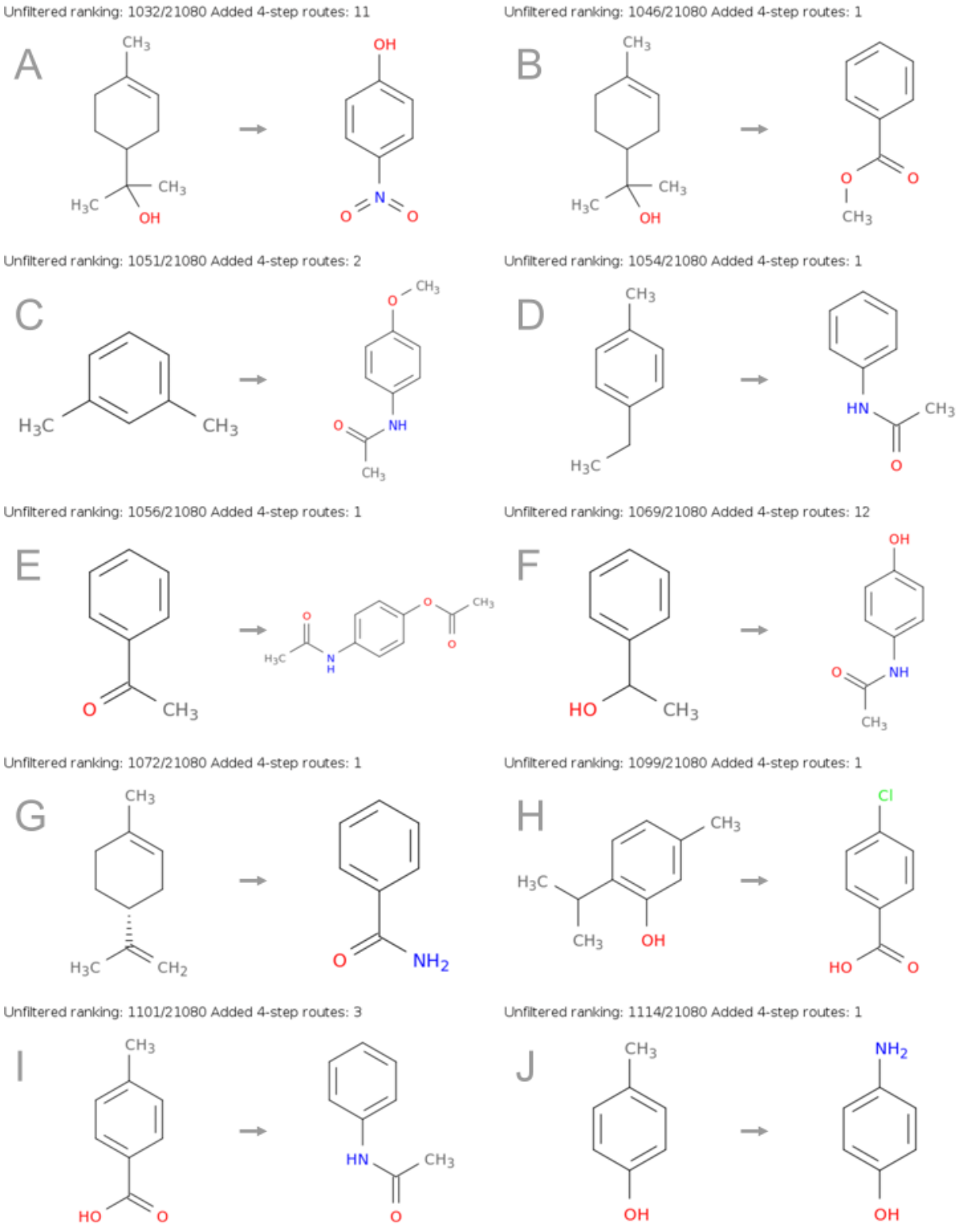

Fig. S19. A selection of transformations, ranked by the algorithm, that were identified as novel and increasing the number of four-step-paths showing transformations 141-150 in decreasing order of likelihood ratio magnitude.

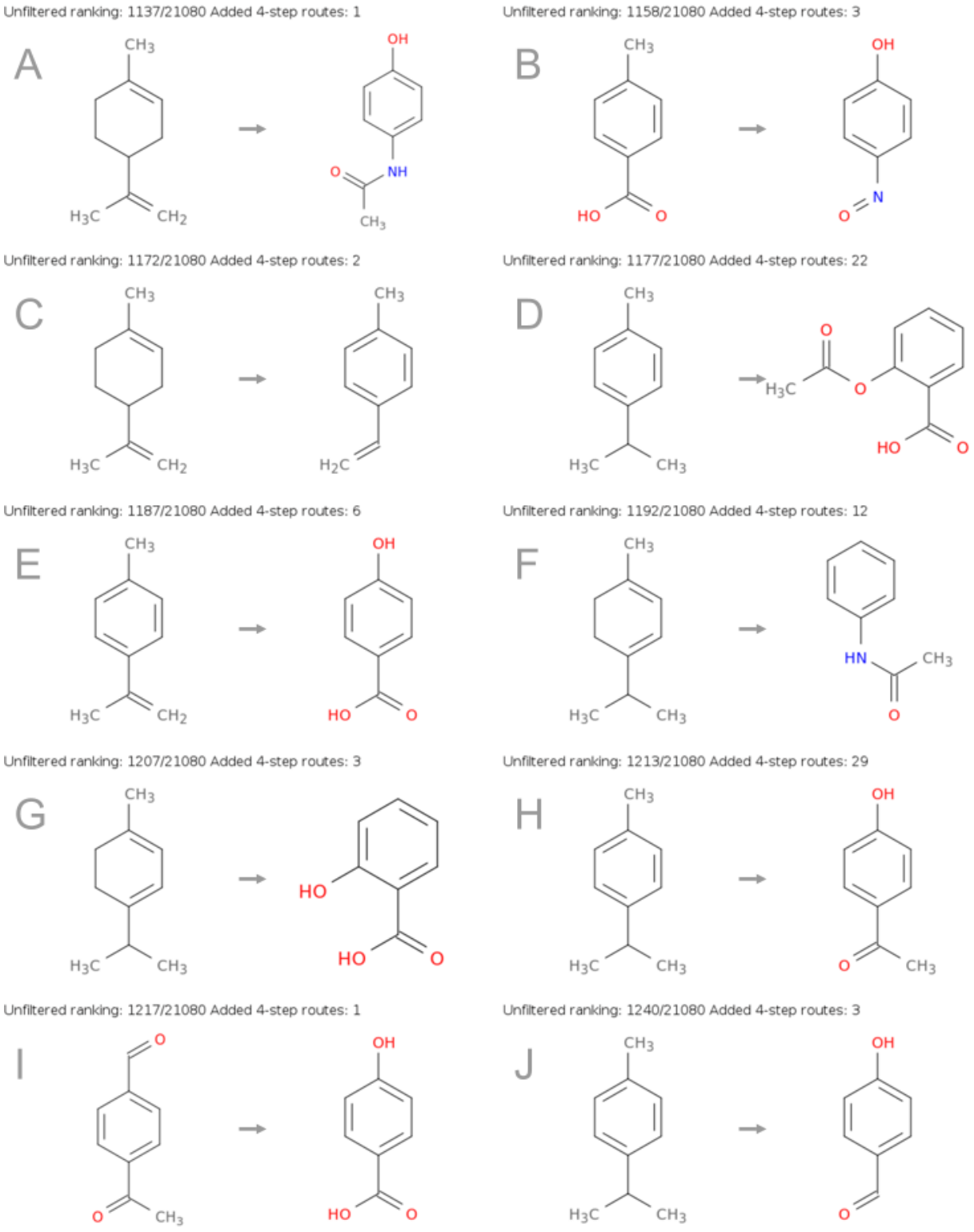

Fig. S20. A selection of transformations, ranked by the algorithm, that were identified as novel and increasing the number of four-step-paths showing transformations 151-160 in decreasing order of likelihood ratio magnitude.

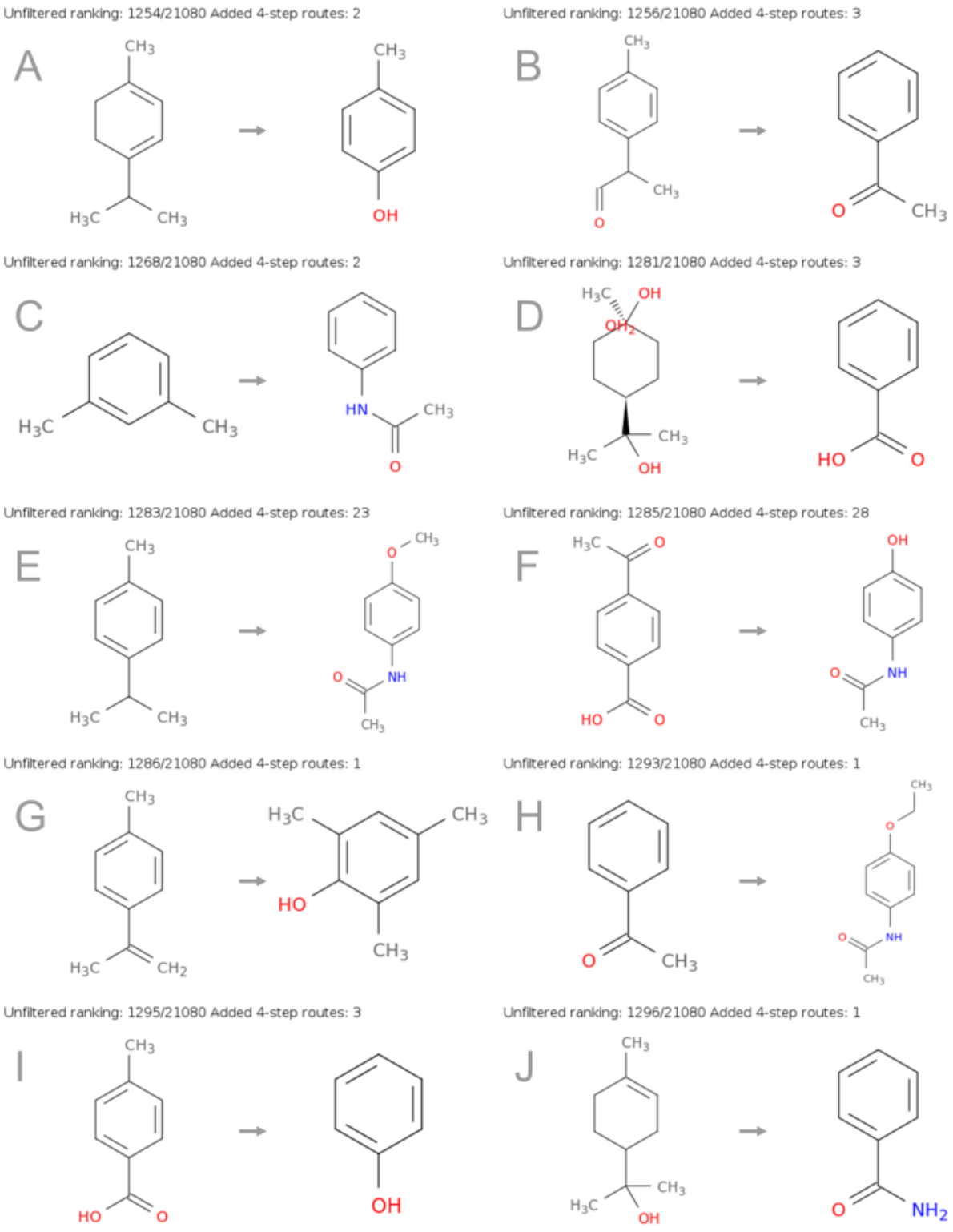

Fig. S21. A selection of transformations, ranked by the algorithm, that were identified as novel and increasing the number of four-step-paths showing transformations 161-170 in decreasing order of likelihood ratio magnitude.

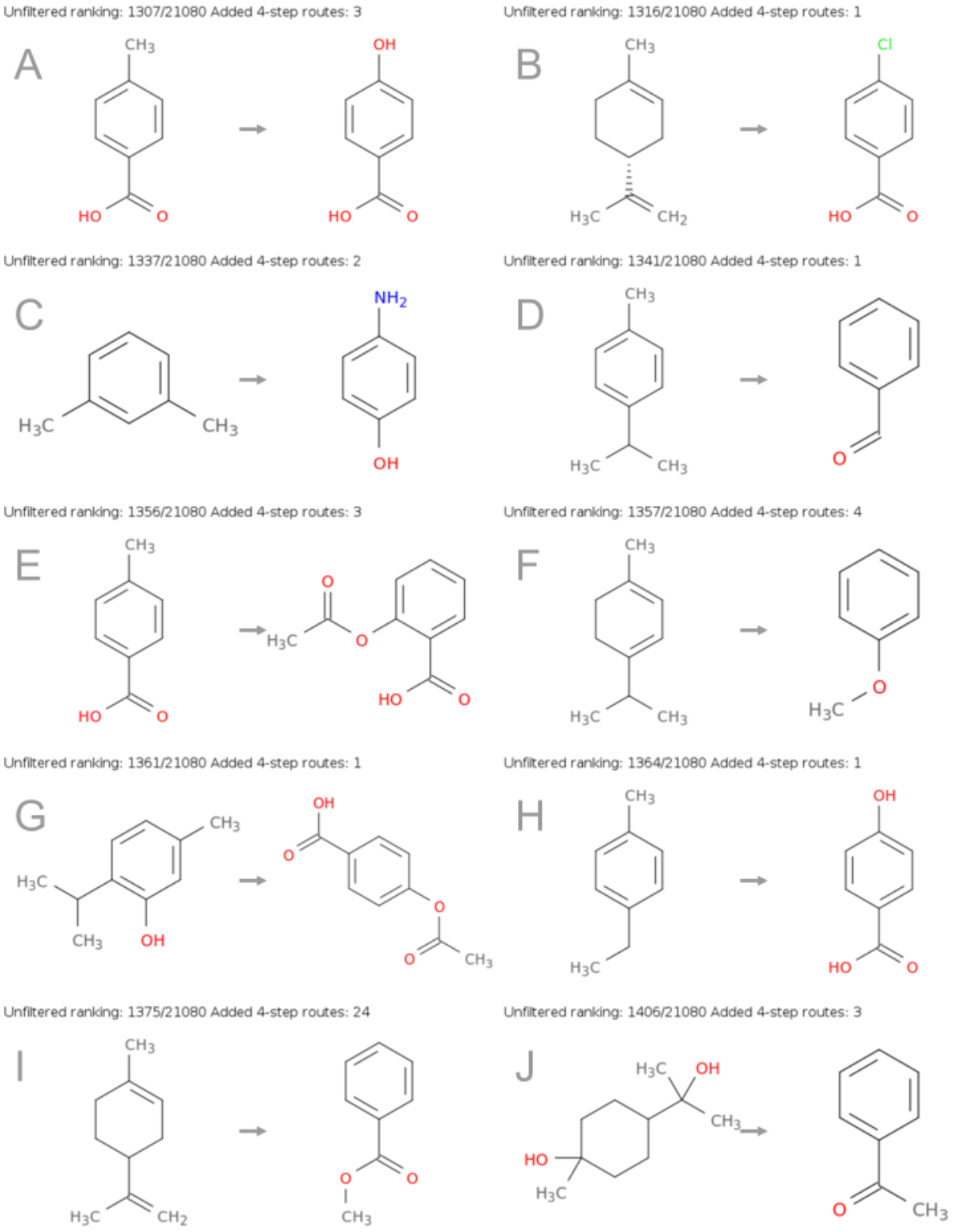

Fig. S22. A selection of transformations, ranked by the algorithm, that were identified as novel and increasing the number of four-step-paths showing transformations 171-180 in decreasing order of likelihood ratio magnitude.

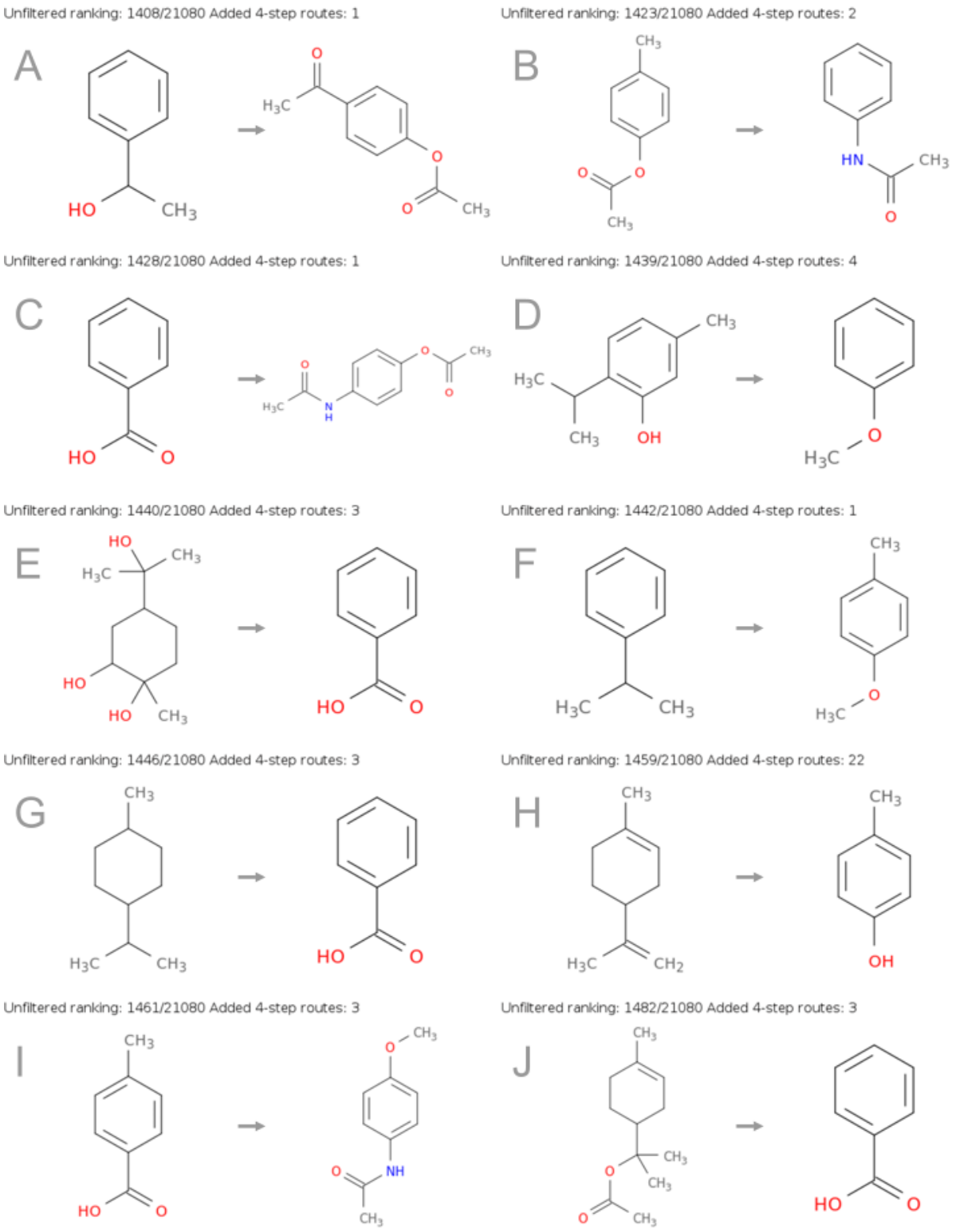

Fig. S23. A selection of transformations, ranked by the algorithm, that were identified as novel and increasing the number of four-step-paths showing transformations 181-190 in decreasing order of likelihood ratio magnitude.

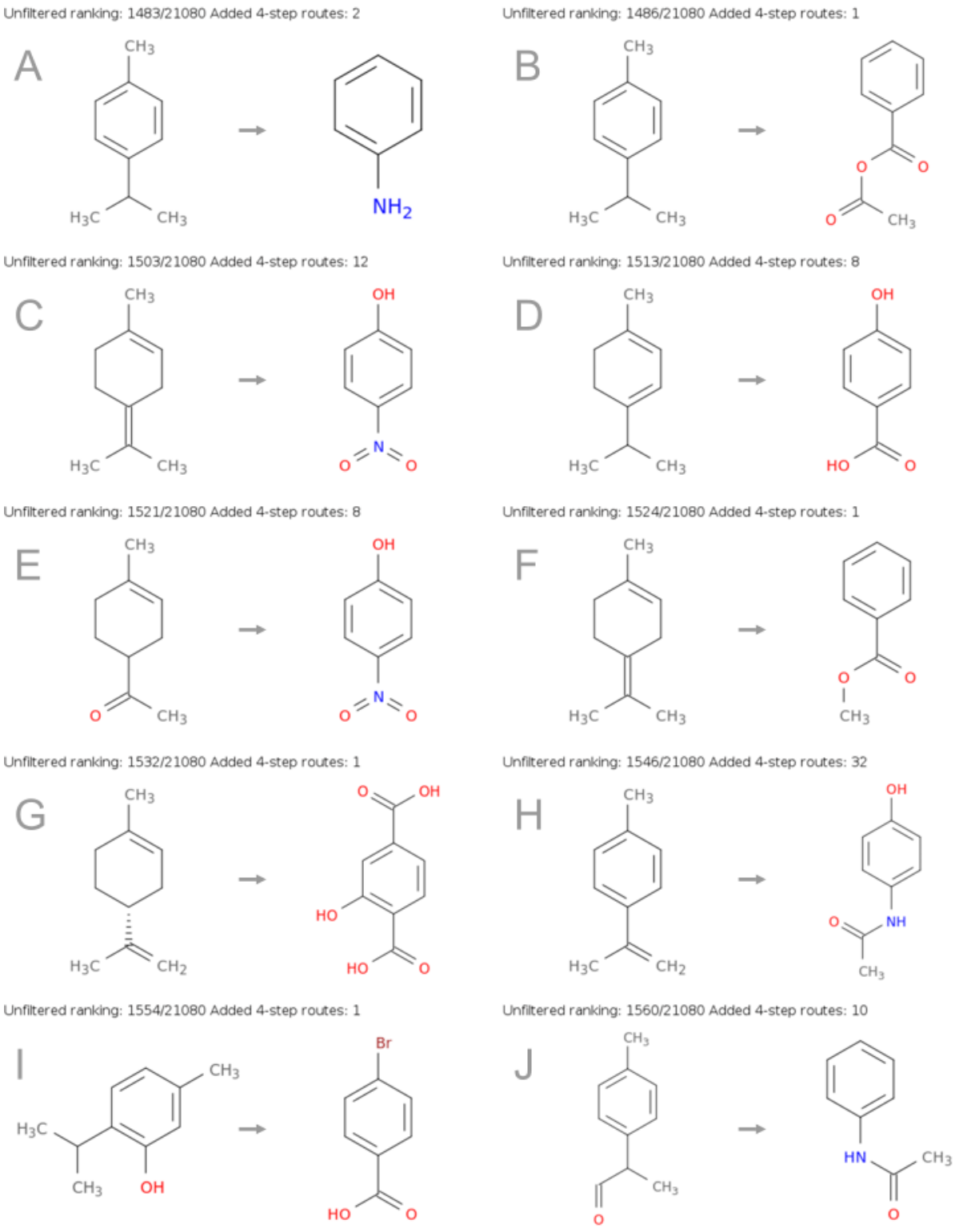

Fig. S24. A selection of transformations, ranked by the algorithm, that were identified as novel and increasing the number of four-step-paths showing transformations 191-200 in decreasing order of likelihood ratio magnitude.

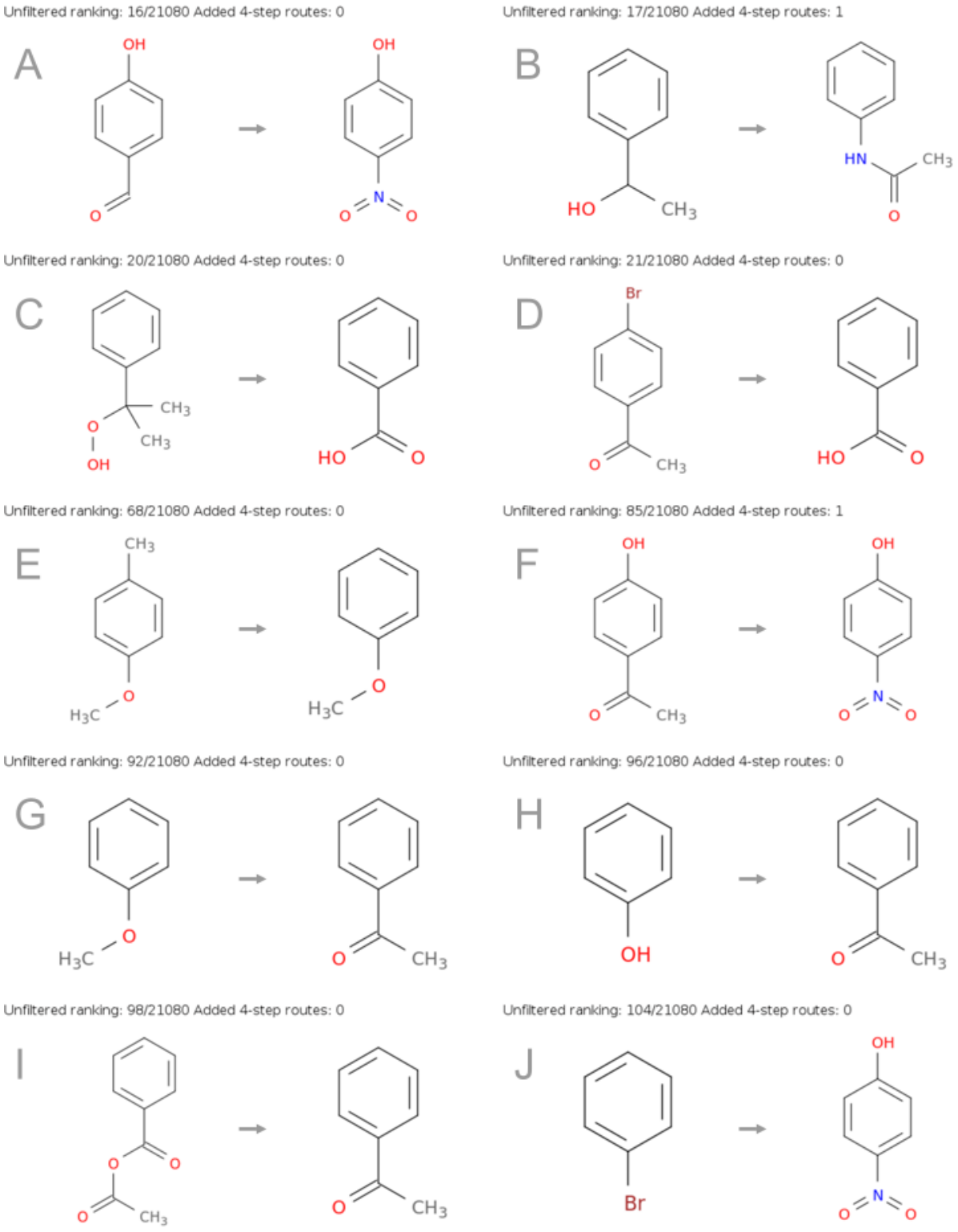

Fig. S25. A selection of transformations evaluated using the link prediction algorithm that turned out to be contained in Reaxys and their unfiltered rankings, showing transformations 1-10 in decreasing order of likelihood ratio magnitude.

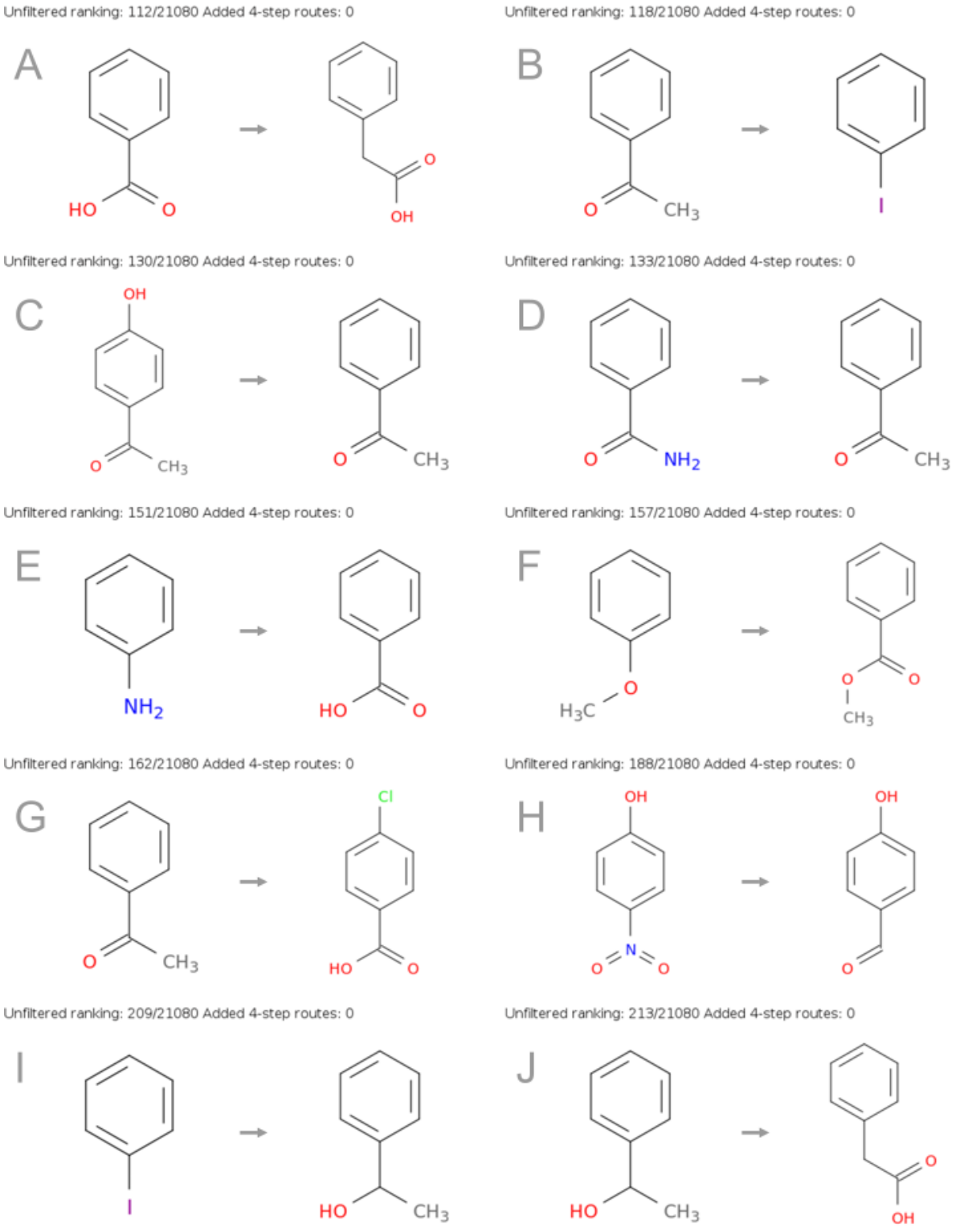

Fig. S26. A selection of transformations evaluated using the link prediction algorithm that turned out to be contained in Reaxys and their unfiltered rankings, showing transformations 11-20 in decreasing order of likelihood ratio magnitude.

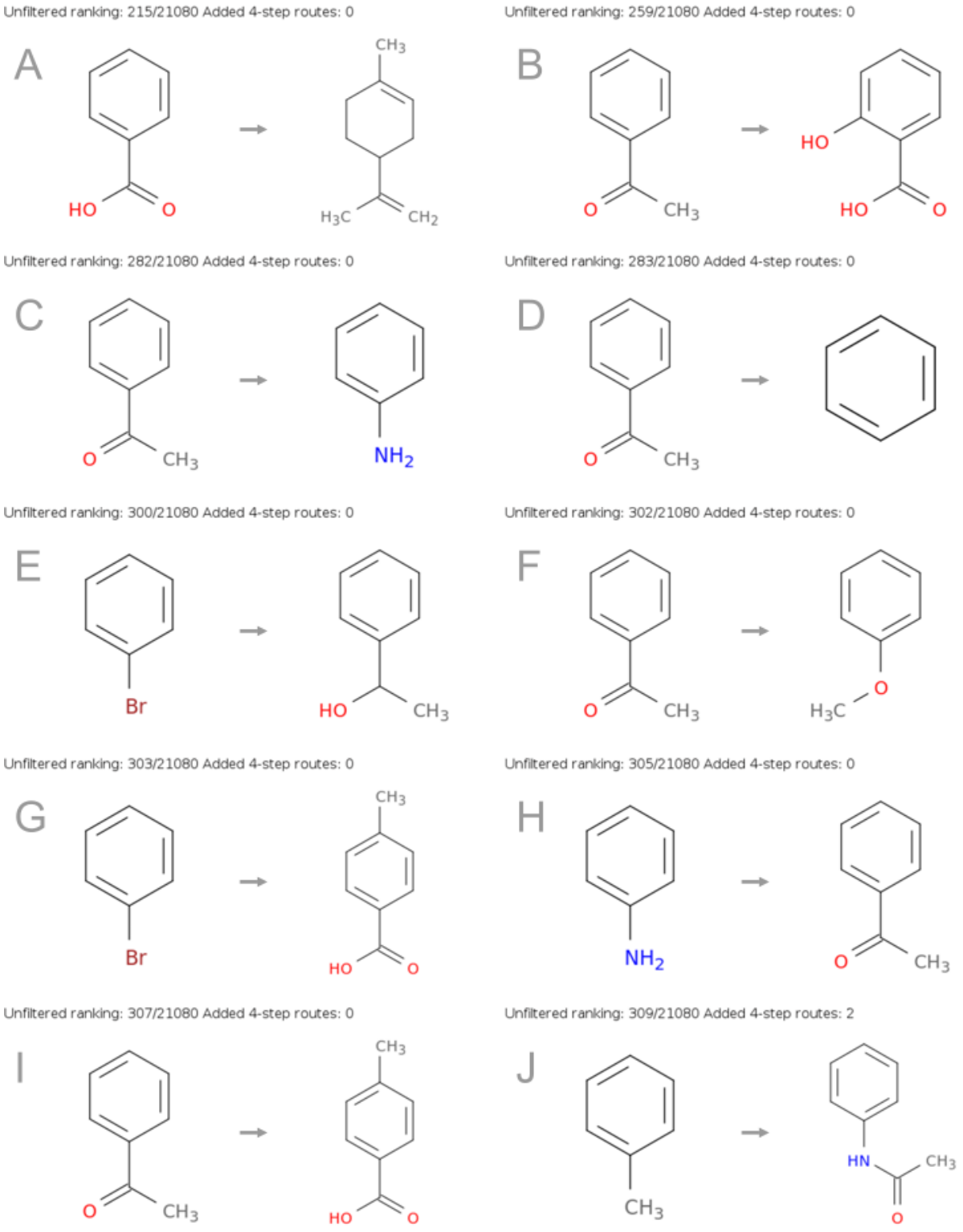

Fig. S27. A selection of transformations evaluated using the link prediction algorithm that turned out to be contained in Reaxys and their unfiltered rankings, showing transformations 21-30 in decreasing order of likelihood ratio magnitude.

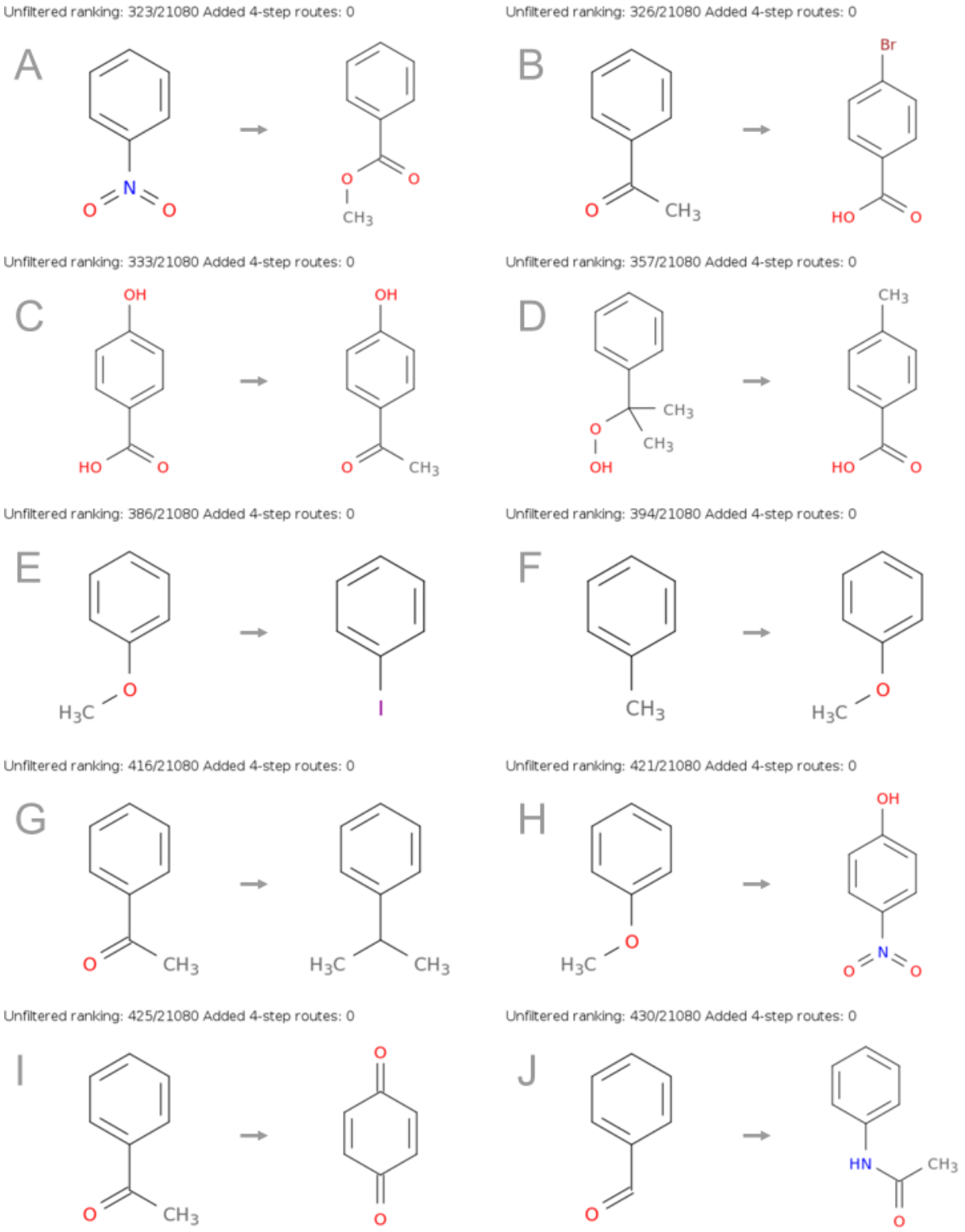

Fig. S28. A selection of transformations evaluated using the link prediction algorithm that turned out to be contained in Reaxys and their unfiltered rankings, showing transformations 31-40 in decreasing order of likelihood ratio magnitude.

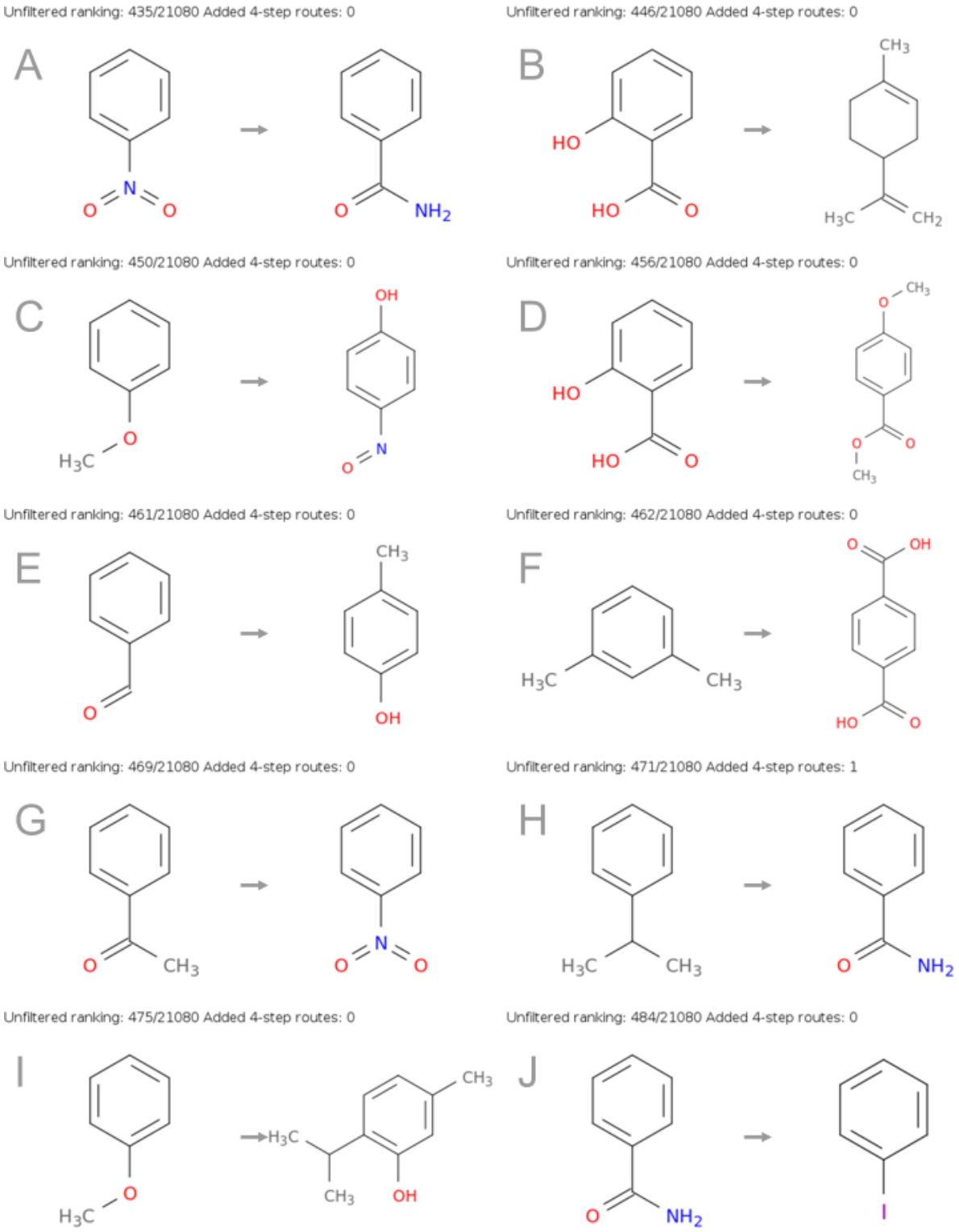

Fig. S29. A selection of transformations evaluated using the link prediction algorithm that turned out to be contained in Reaxys and their unfiltered rankings, showing transformations 41-50 in decreasing order of likelihood ratio magnitude.

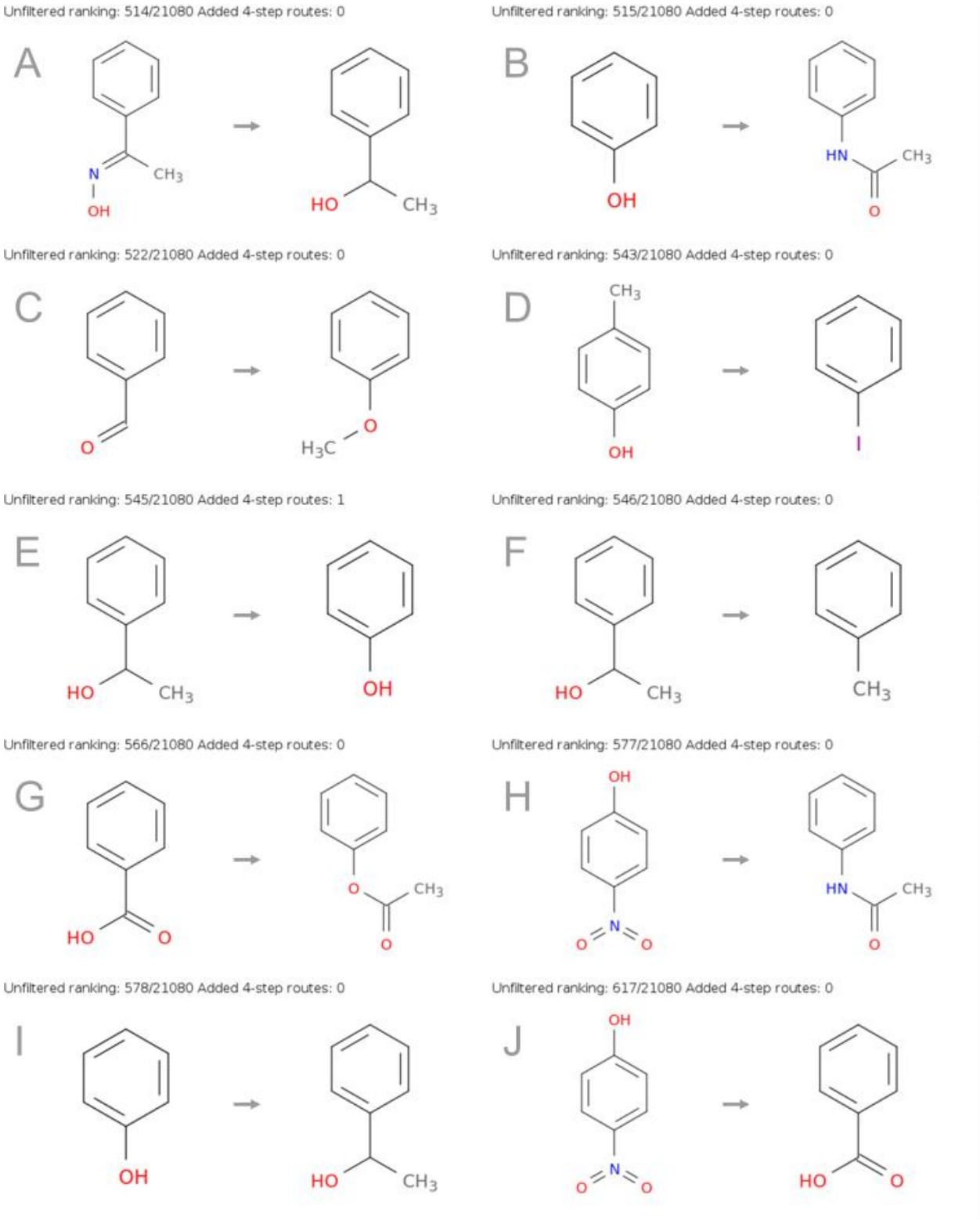

Fig. S30. A selection of transformations evaluated using the link prediction algorithm that turned out to be contained in Reaxys and their unfiltered rankings, showing transformations 51-60 in decreasing order of likelihood ratio magnitude.

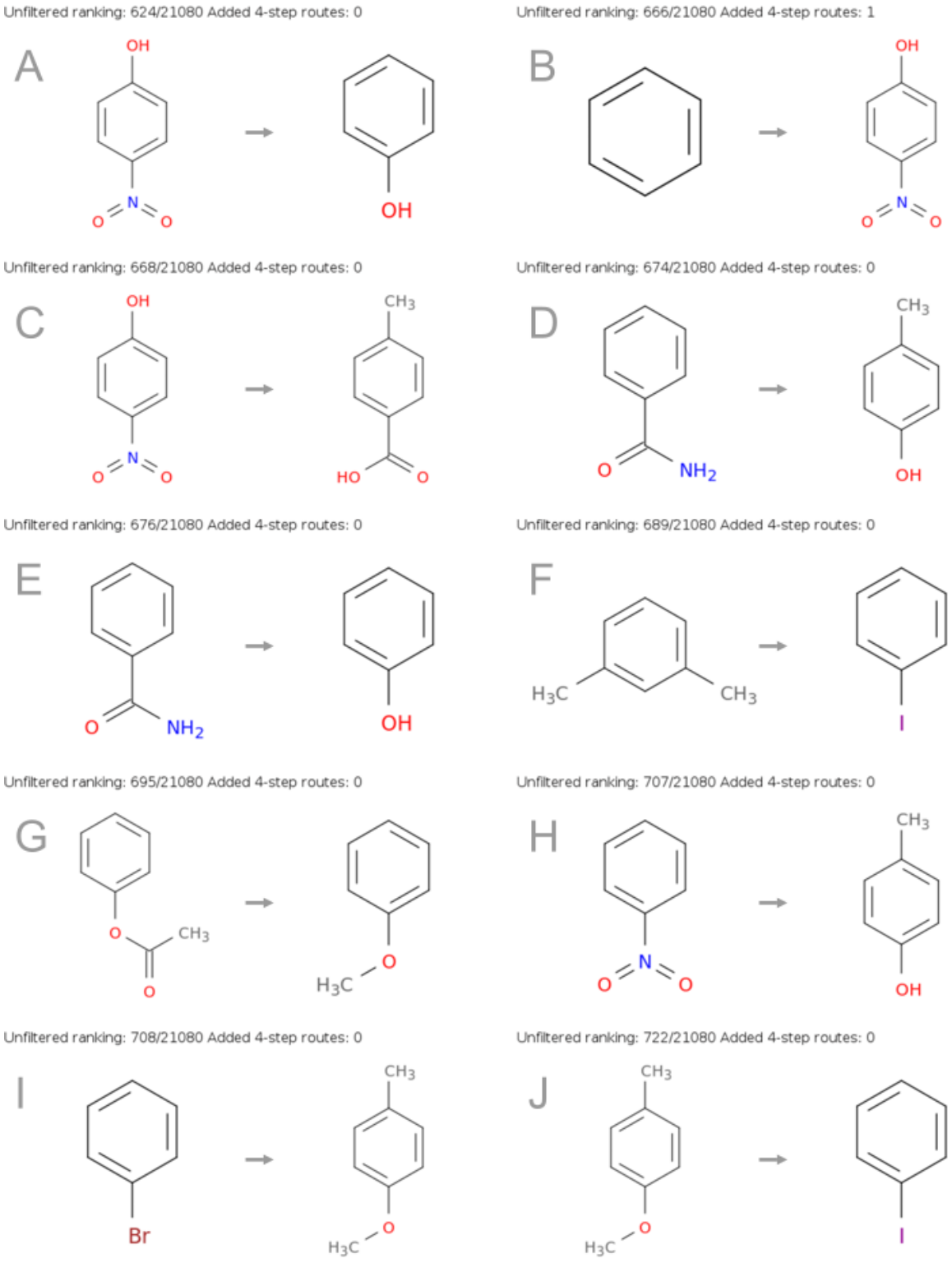

Fig. S31. A selection of transformations evaluated using the link prediction algorithm that turned out to be contained in Reaxys and their unfiltered rankings, showing transformations 61-70 in decreasing order of likelihood ratio magnitude.

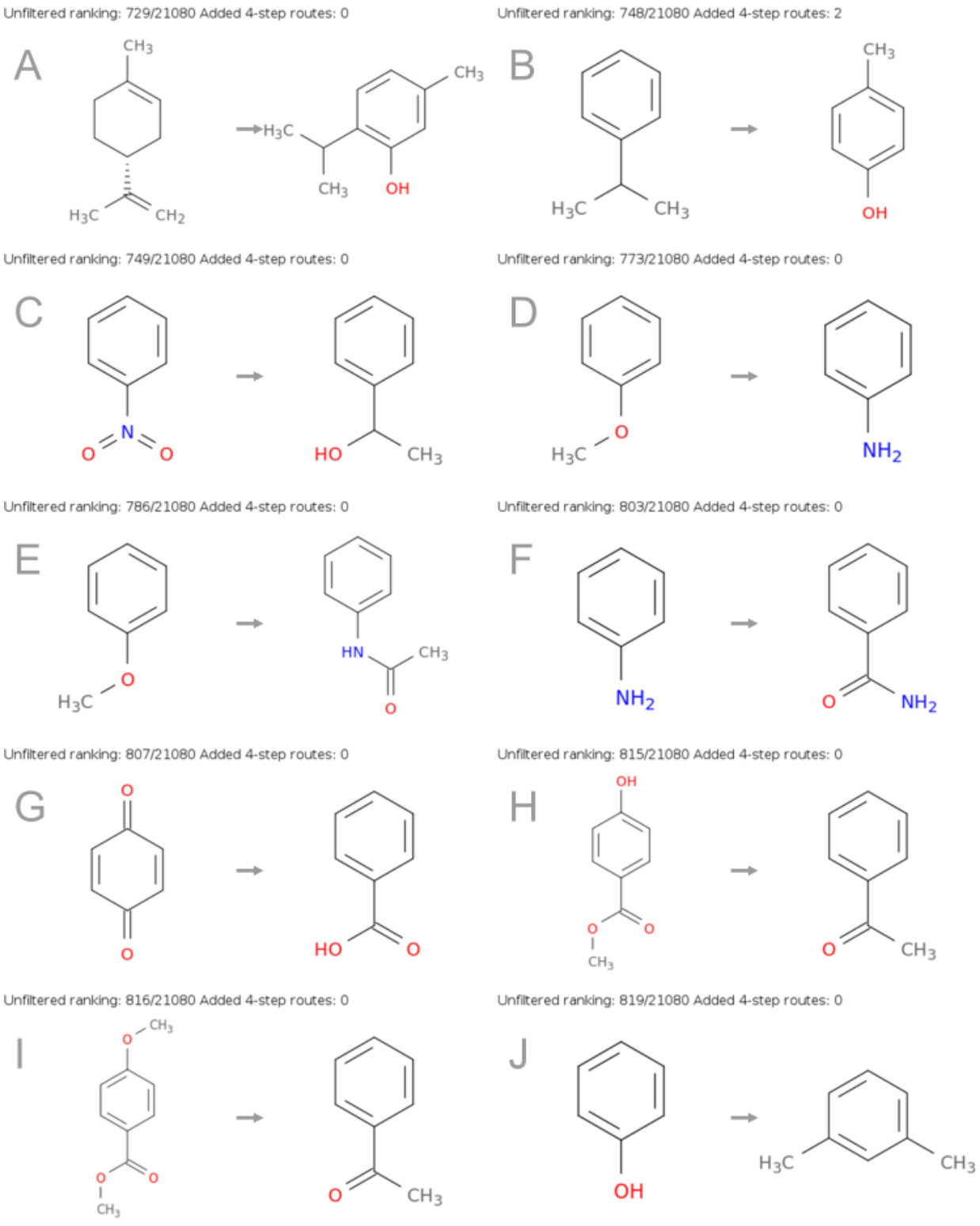

Fig. S32. A selection of transformations evaluated using the link prediction algorithm that turned out to be contained in Reaxys and their unfiltered rankings, showing transformations 71-80 in decreasing order of likelihood ratio magnitude.

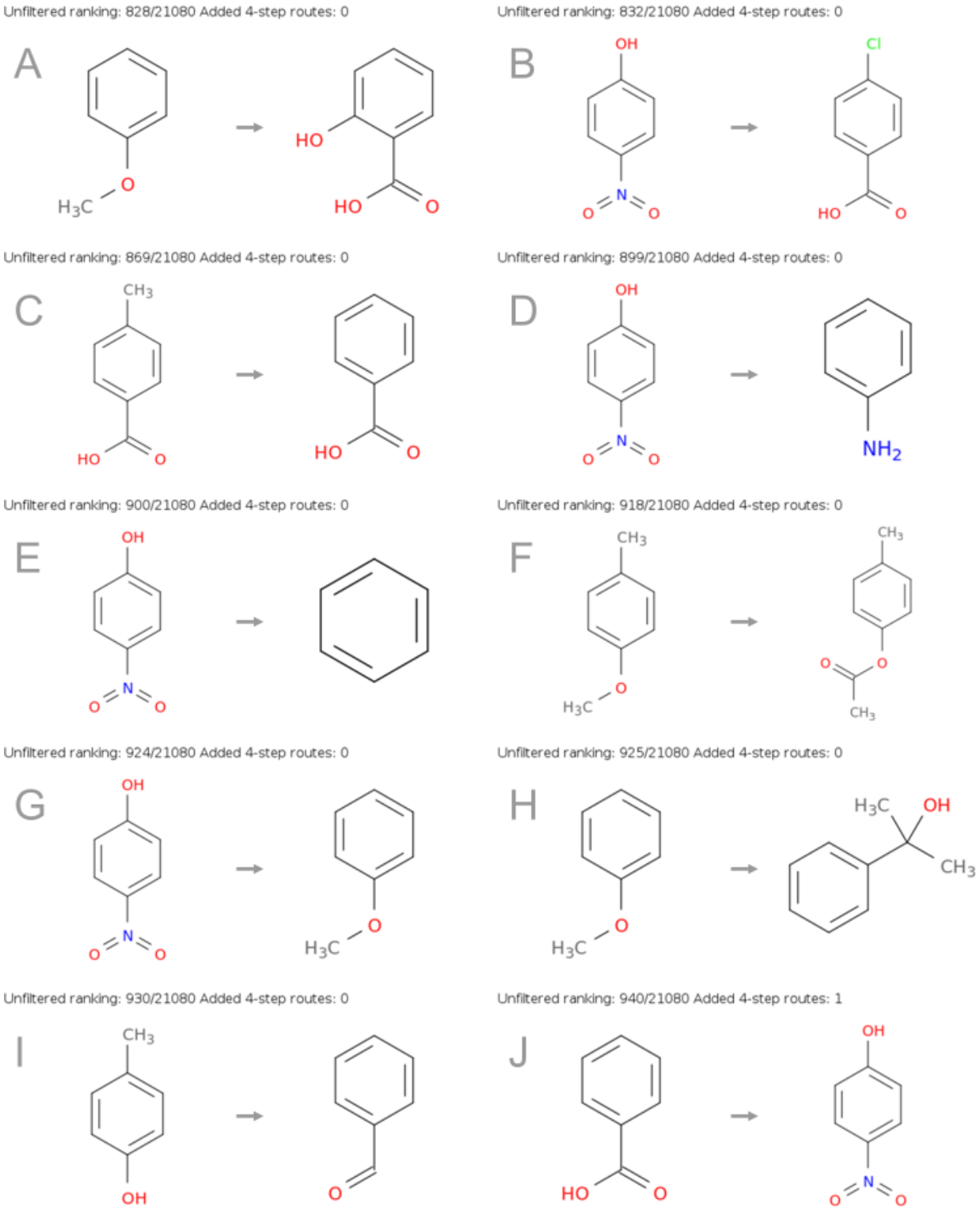

Fig. S33. A selection of transformations evaluated using the link prediction algorithm that turned out to be contained in Reaxys and their unfiltered rankings, showing transformations 81-90 in decreasing order of likelihood ratio magnitude.

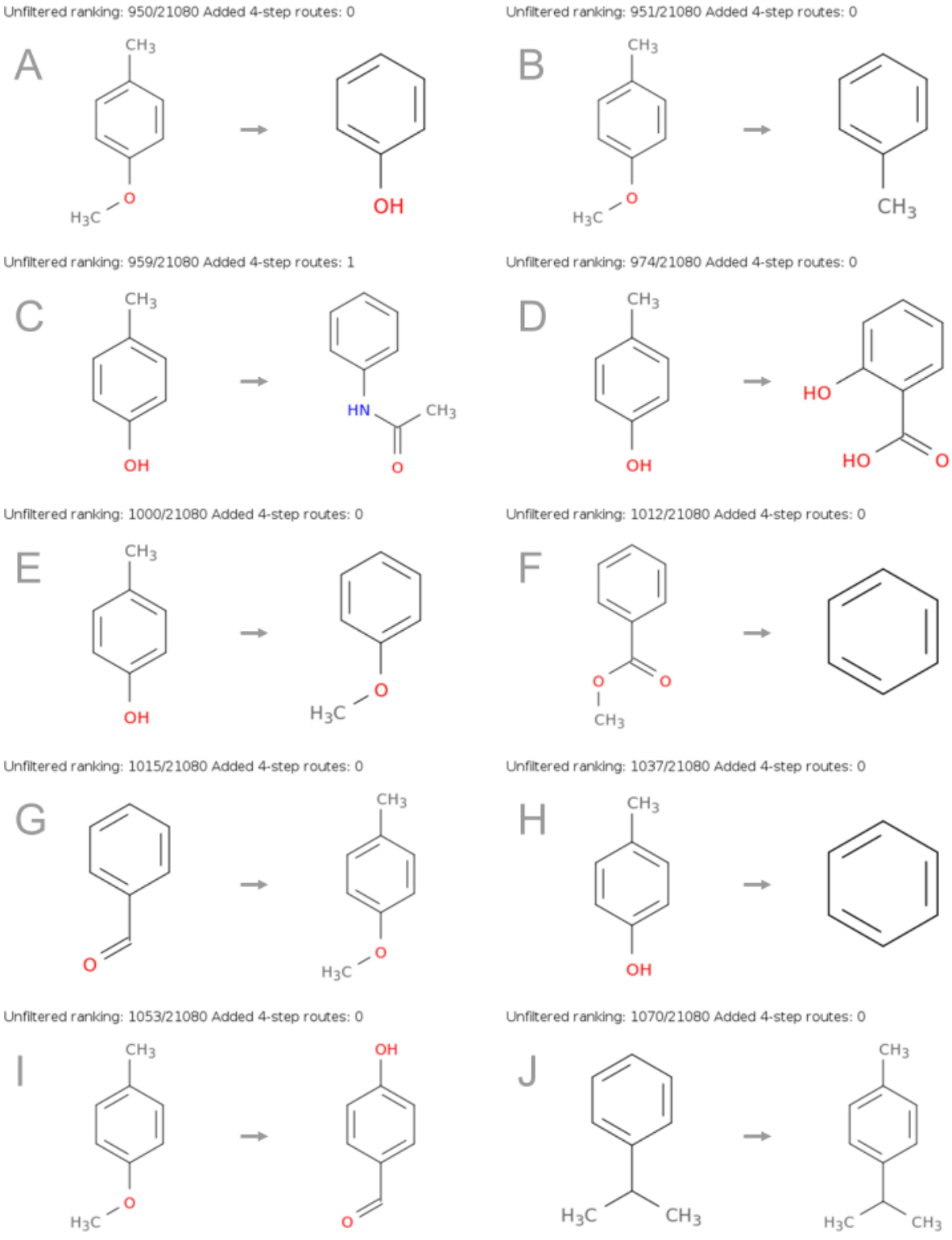

Fig. S34. A selection of transformations evaluated using the link prediction algorithm that turned out to be contained in Reaxys and their unfiltered rankings, showing transformations 91-100 in decreasing order of likelihood ratio magnitude.

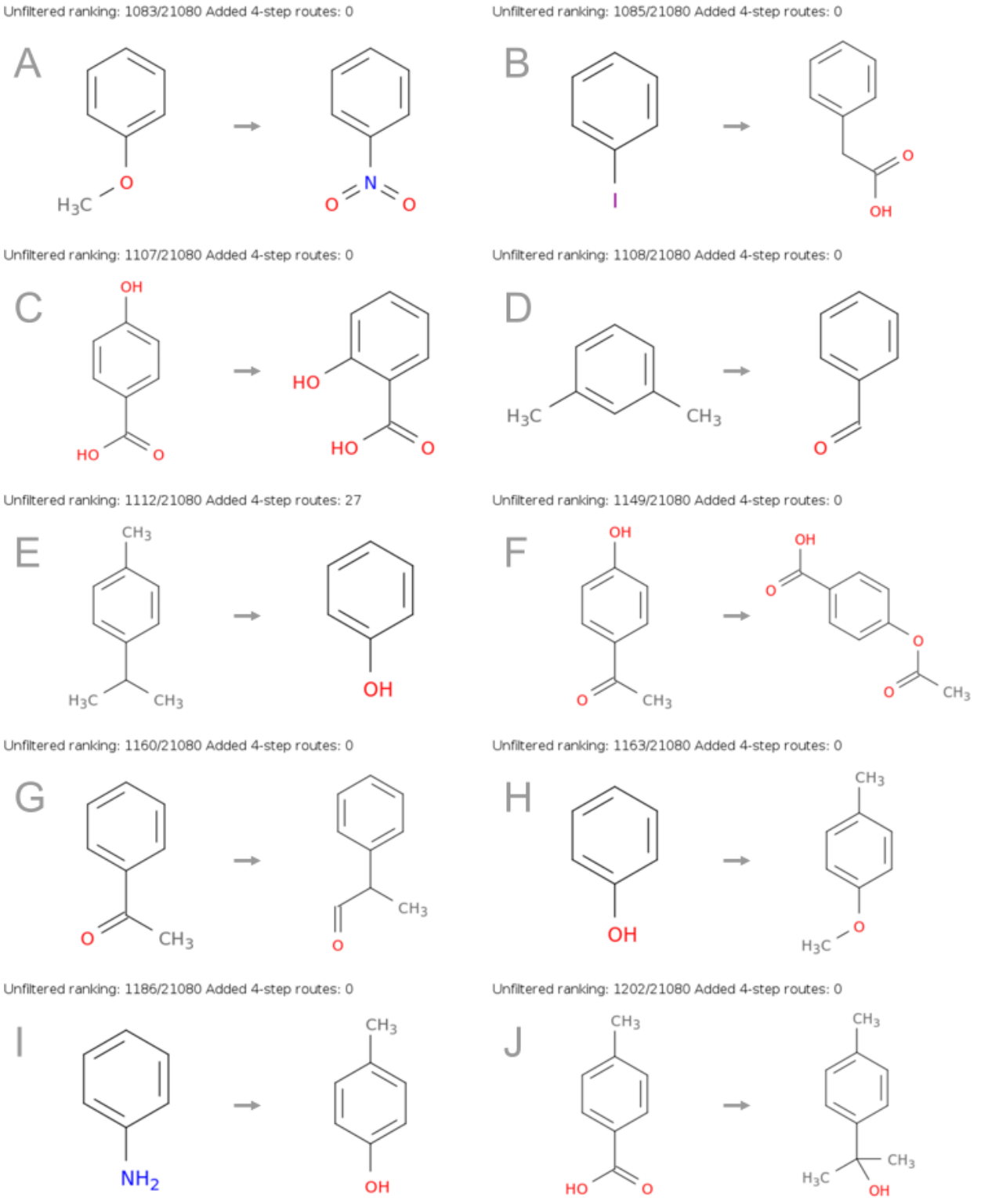

Fig. S35. A selection of transformations evaluated using the link prediction algorithm that turned out to be contained in Reaxys and their unfiltered rankings, showing transformations 101-110 in decreasing order of likelihood ratio magnitude.

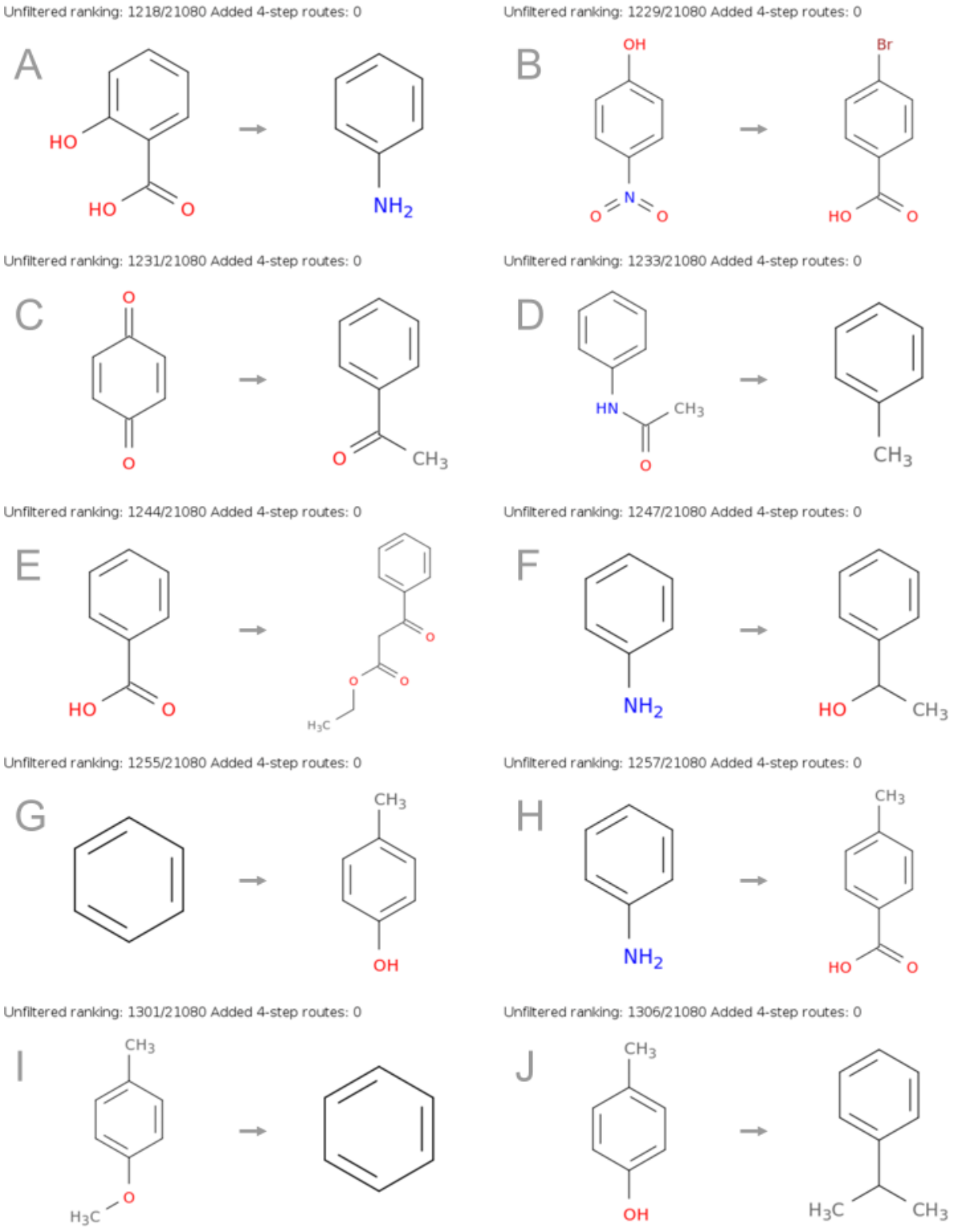

Fig. S36. A selection of transformations evaluated using the link prediction algorithm that turned out to be contained in Reaxys and their unfiltered rankings, showing transformations 111-120 in decreasing order of likelihood ratio magnitude.

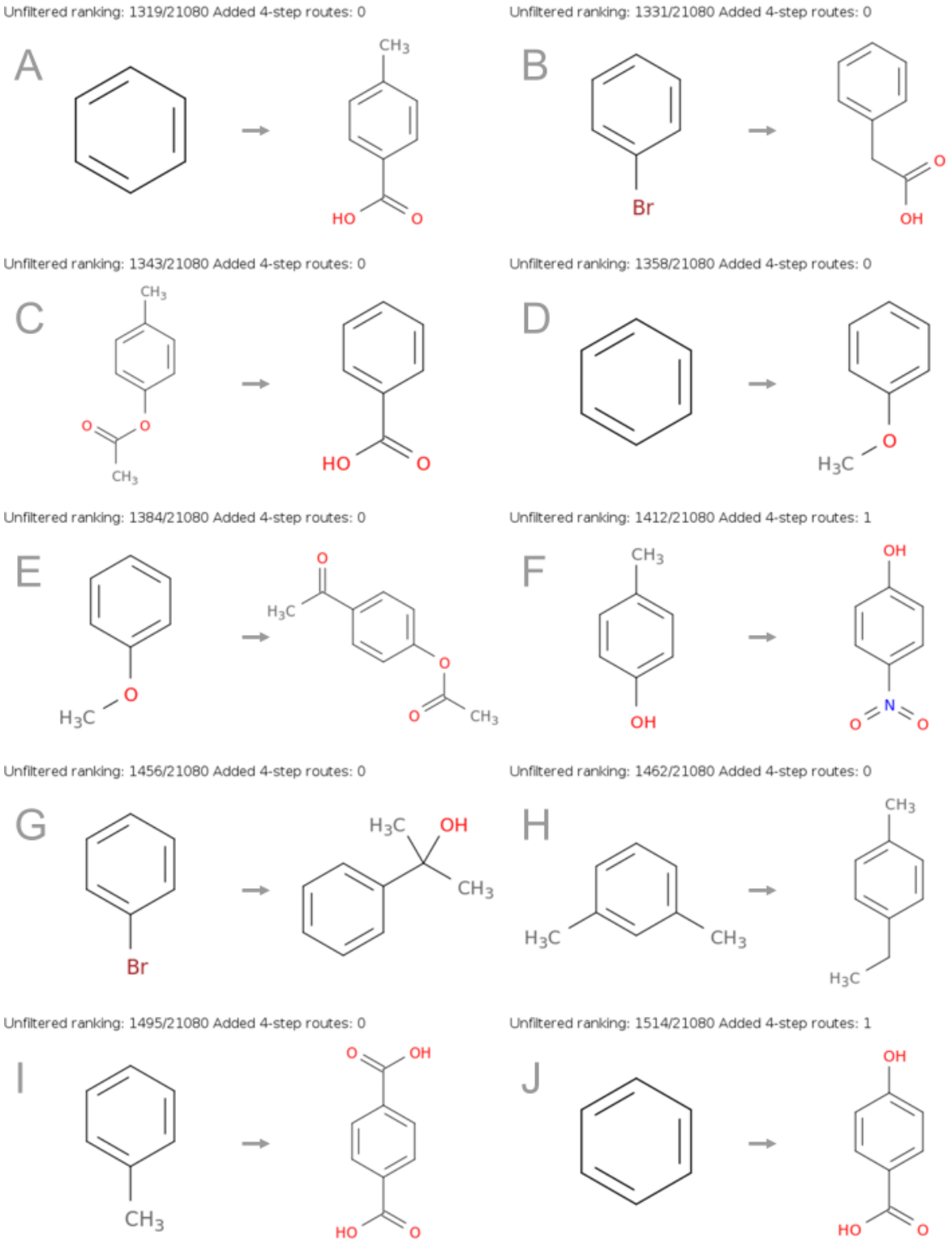

Fig. S37. A selection of transformations evaluated using the link prediction algorithm that turned out to be contained in Reaxys and their unfiltered rankings, showing transformations 121-130 in decreasing order of likelihood ratio magnitude.

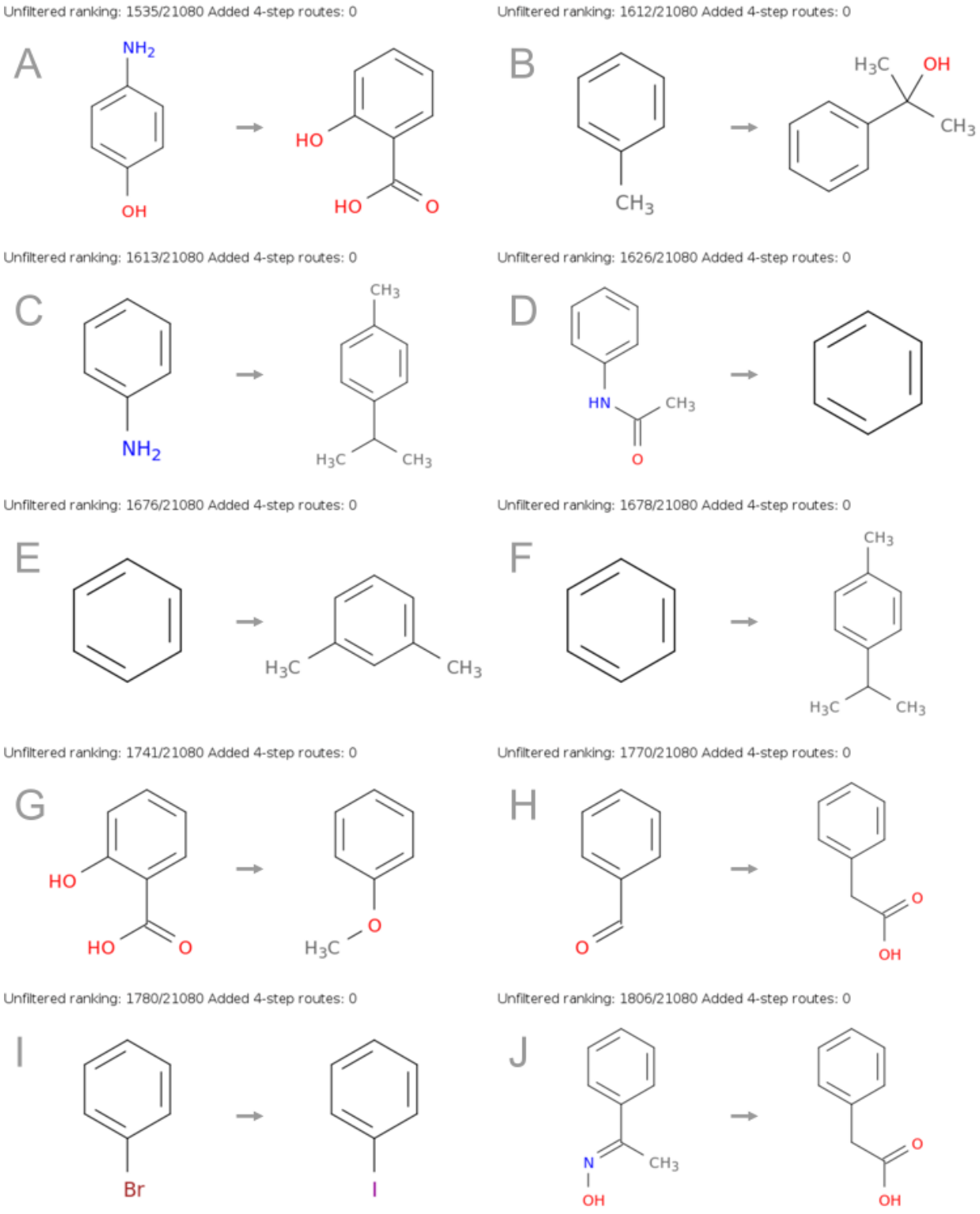

Fig. S38. A selection of transformations evaluated using the link prediction algorithm that turned out to be contained in Reaxys and their unfiltered rankings, showing transformations 131-140 in decreasing order of likelihood ratio magnitude.

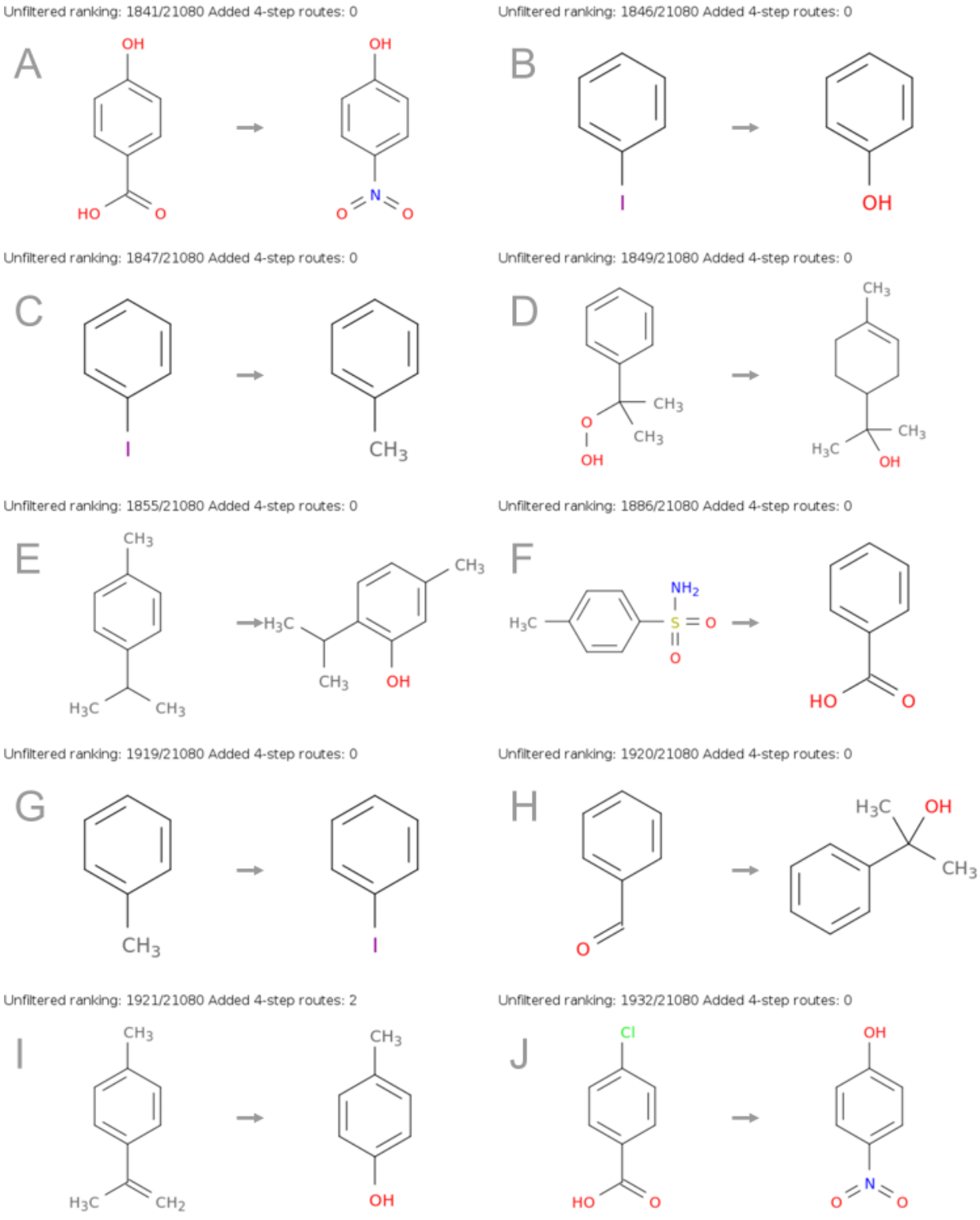

Fig. S39. A selection of transformations evaluated using the link prediction algorithm that turned out to be contained in Reaxys and their unfiltered rankings, showing transformations 141-150 in decreasing order of likelihood ratio magnitude.

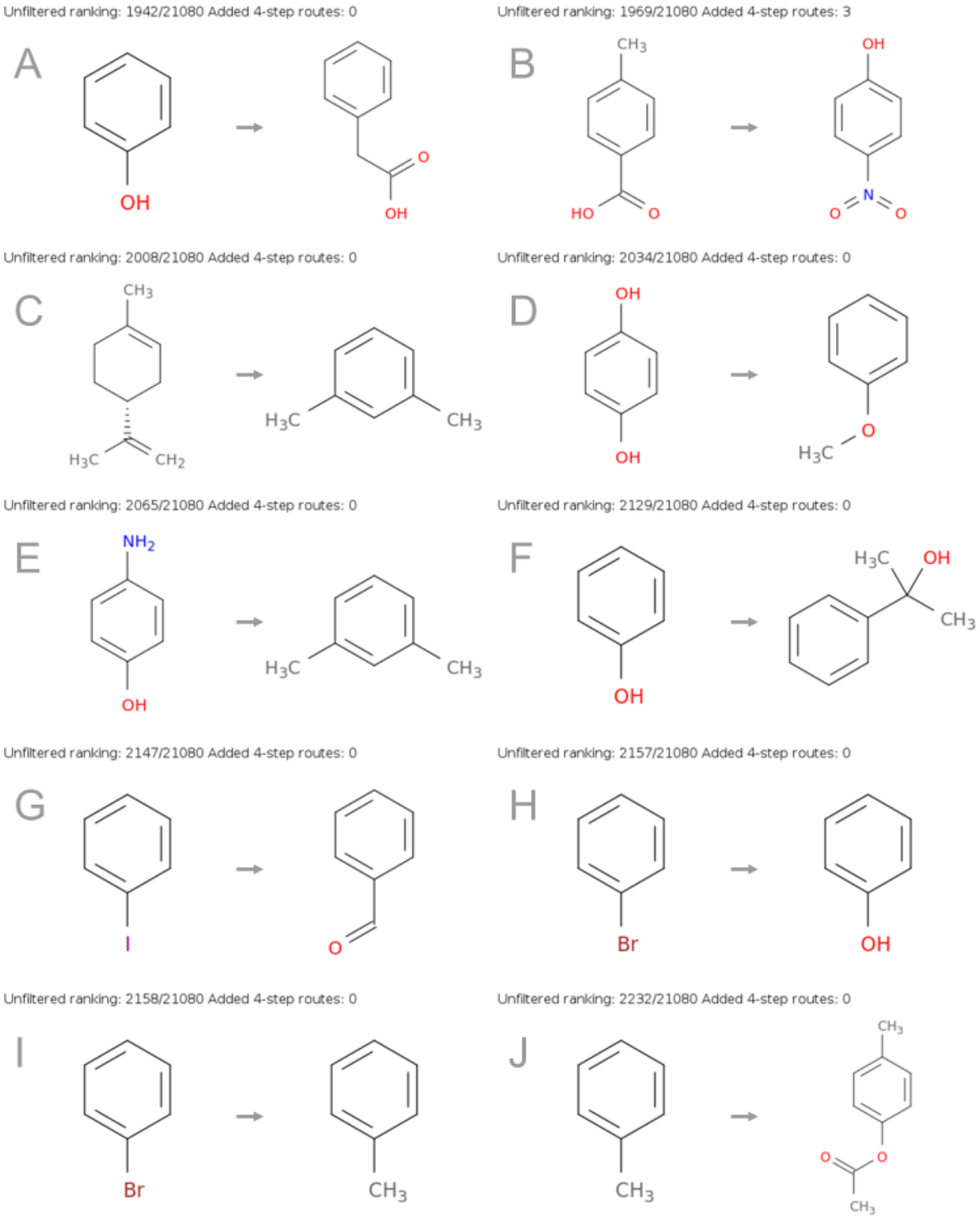

Fig. S40. A selection of transformations evaluated using the link prediction algorithm that turned out to be contained in Reaxys and their unfiltered rankings, showing transformations 151-160 in decreasing order of likelihood ratio magnitude.

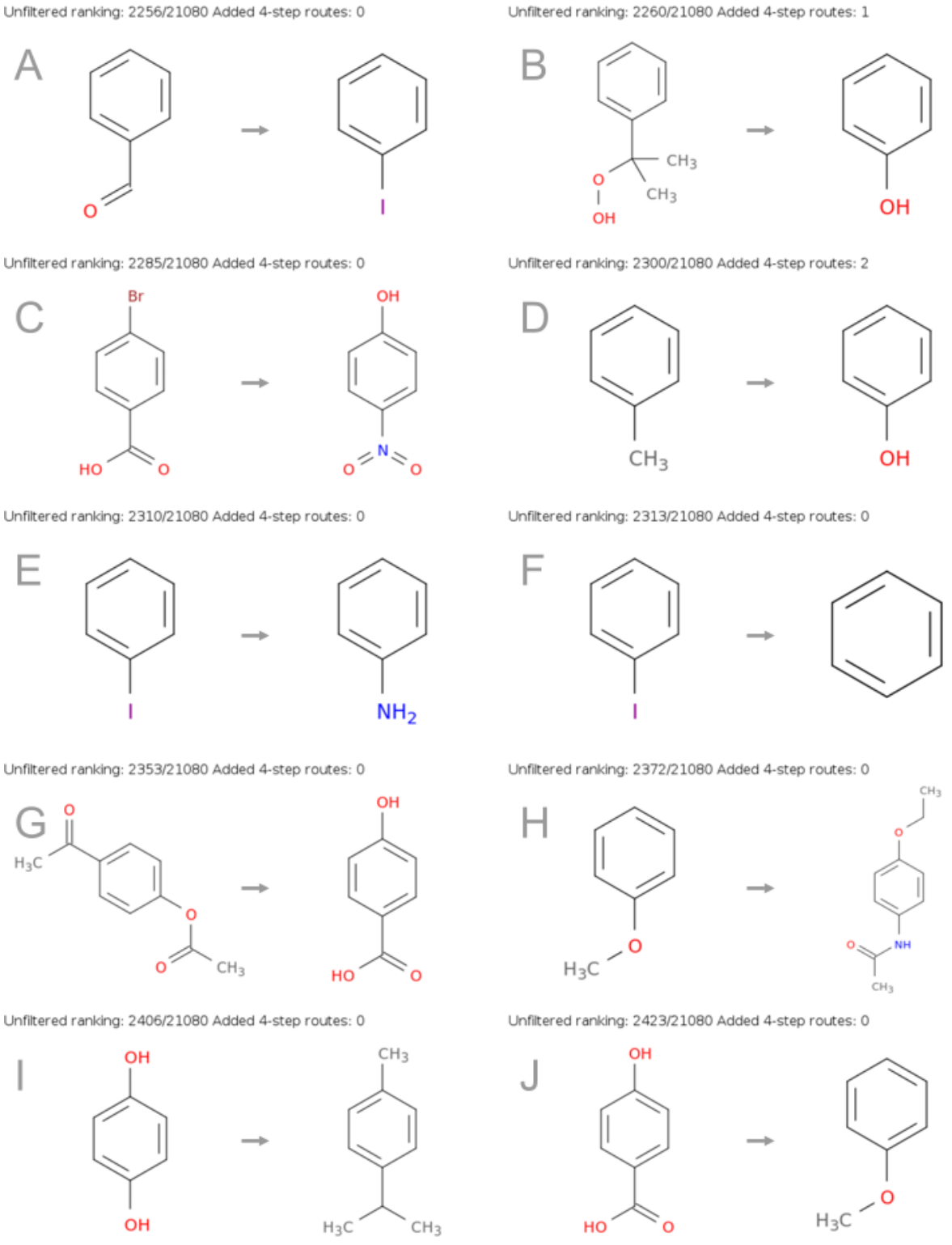

Fig. S41. A selection of transformations evaluated using the link prediction algorithm that turned out to be contained in Reaxys and their unfiltered rankings, showing transformations 161-170 in decreasing order of likelihood ratio magnitude.

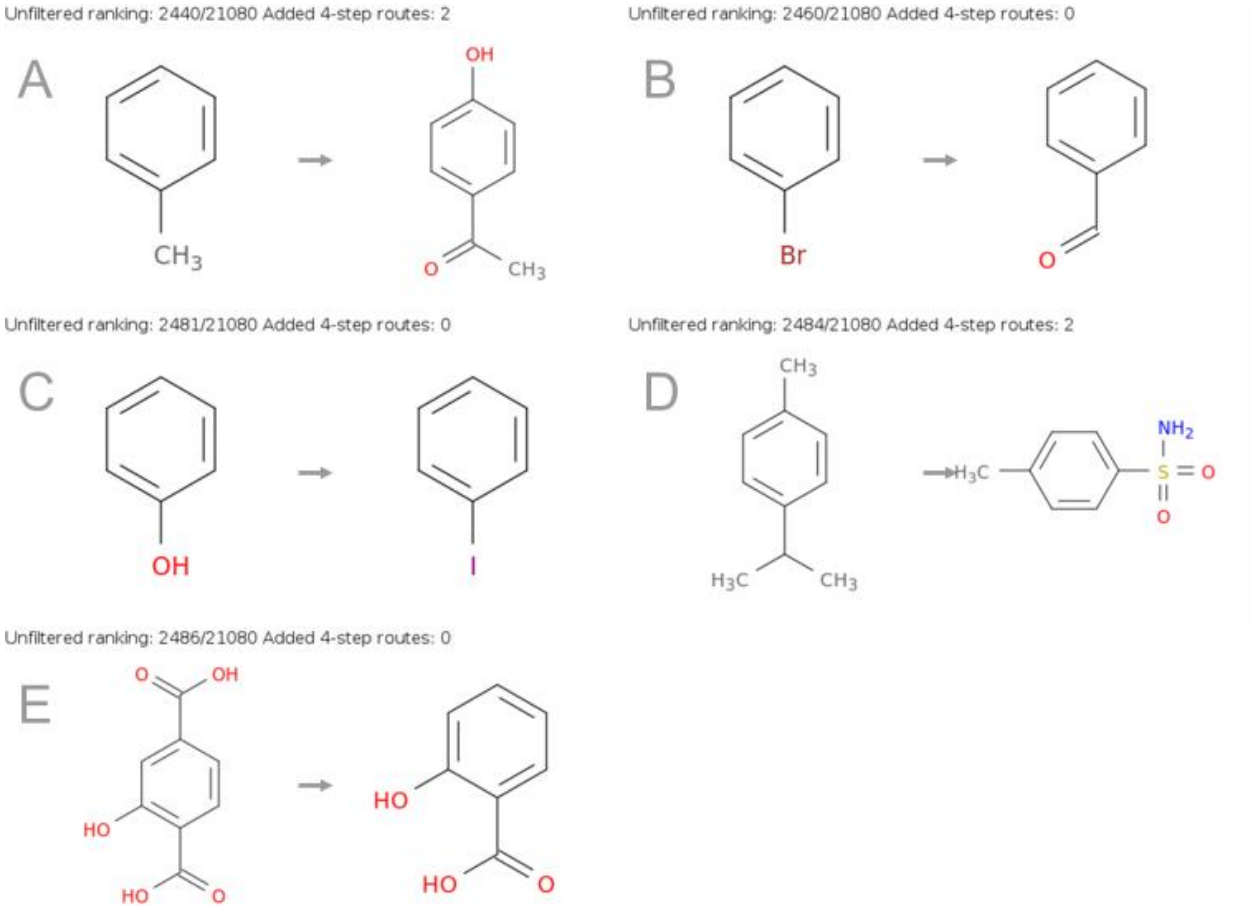

Fig. S42. A selection of transformations evaluated using the link prediction algorithm that turned out to be contained in Reaxys and their unfiltered rankings, showing transformations 171-175 in decreasing order of likelihood ratio magnitude.

Table S1. Configurations of SBMs for which the MDL was found and the posterior likelihood distribution sampled in order to calculate likelihood ratios.

Configuration #	Nested	Degree corrected	Overlapping	Non-informative prior
1	TRUE	TRUE	TRUE	no
2	TRUE	TRUE	FALSE	no
3	TRUE	FALSE	FALSE	n/a
4	TRUE	FALSE	TRUE	n/a
5	TRUE	TRUE	FALSE	yes
6	TRUE	TRUE	TRUE	yes
7	FALSE	TRUE	TRUE	no
8	FALSE	FALSE	TRUE	n/a
9	FALSE	FALSE	FALSE	n/a
10	FALSE	TRUE	FALSE	no
11	FALSE	TRUE	FALSE	yes
12	FALSE	TRUE	TRUE	yes

Table S2. MDL values for each run for the different SBM configurations along with the mean and standard deviation (St. Dev.) for each configuration. "N-I prior" stands for "non-informative prior".

Nested	Degree corrected	Overlapping	N-I prior	MDL_1	MDL_2	MDL_3	Mean	St. Dev.
TRUE	TRUE	TRUE	no	119186	119186	119186	119186	0
TRUE	TRUE	FALSE	no	115328	115778	114978	115361	401
TRUE	FALSE	FALSE	n/a	120365	120522	120015	120300	260
TRUE	FALSE	TRUE	n/a	131814	135094	133912	133607	1662
TRUE	TRUE	FALSE	yes	127801	128938	127264	128001	855
TRUE	TRUE	TRUE	yes	131036	131036	131036	131036	0
FALSE	TRUE	TRUE	no	119186	119186	119186	119186	0
FALSE	FALSE	TRUE	n/a	133952	138371	138371	136898	2551
FALSE	FALSE	FALSE	n/a	121360	121296	121225	121294	67
FALSE	TRUE	FALSE	no	115313	115303	115702	115440	228
FALSE	TRUE	FALSE	yes	128351	127990	129439	128593	754
FALSE	TRUE	TRUE	yes	131036	131036	131036	131036	0

Table S3. The natural log of the model evidence for each run for the different SBM configurations along with the mean and standard deviation (St. Dev.) for each configuration as well as the total run time for one set of calculations. "N-I prior" stands for "non-informative prior".

Nested	Degree corrected	Over-lapping	N-I prior	ME_1	ME_2	ME_3	Mean	St. Dev.	Run time [h]
TRUE	TRUE	TRUE	no	-302634	-302634	-302634	-302634	0	22
TRUE	TRUE	FALSE	no	-115089	-115041	-115454	-115195	226	2
TRUE	FALSE	FALSE	n/a	-119460	-119416	-120006	-119627	329	2
TRUE	FALSE	TRUE	n/a	-292054	-284535	-298193	-291594	6841	6
TRUE	TRUE	FALSE	yes	-126832	-128122	-124303	-126419	1943	2
TRUE	TRUE	TRUE	yes	-314484	-314484	-314484	-314484	0	27
FALSE	TRUE	TRUE	no	-336184	-341637	-334139	-337320	3876	148
FALSE	FALSE	TRUE	n/a	-398161	-368207	-358687	-375019	20600	68
FALSE	FALSE	FALSE	n/a	-118807	-119849	-119932	-119529	627	1
FALSE	TRUE	FALSE	no	-113491	-112863	-112900	-113085	352	1
FALSE	TRUE	FALSE	yes	-126899	-125939	-126094	-126311	515	1
FALSE	TRUE	TRUE	yes	-353976	-349243	-353976	-352398	2732	146

Table S4. Changes of rankings of reaction suggestions between the four-step and five-step link prediction case study showing the old and new filtered ranking and the corresponding unfiltered percentage ranking.

Transformation	Old ranking	New ranking	Old top percent	New top percent
Figure 2.A	1	8	1%	0.5%
Figure 2.B	2	4	1%	0.5%
Figure 2.C	3	11	2%	0.5%
Figure 2.D	4	14	2%	0.5%
Figure 2.E	5	17	3%	0.5%
Figure 2.F	6	26	4%	1%
Figure 2.G	7	63	4%	2%
Figure 2.H	8	25	5%	1%
Figure 3.A	9	28	5%	1%
Figure 3.B	10	42	8%	1.5%
Figure 3.C	11	51	8%	2%
Figure 3.D	12	49	8%	2%
Figure 3.E	13	31	9%	1%
Figure 3.F	14	58	10%	2%
Figure 3.G	15	19	11%	1%
Figure 3.H	16	38	11%	2%