Angewandte Chemie
Eine Zeitschrift der Gesellschaft Deutscher Chemiker

Supporting Information

Co-crystal Prediction by Artificial Neural Networks**

*Jan-Joris Devogelaer, Hugo Meekes, Paul Tinnemans, Elias Vlieg, and René de Gelder**

anie_202009467_sm_miscellaneous_information.pdf

This page is intentionally left blank to respect a correct page numbering after the automatic addition of title and author information.

Contents

S1 Data extraction from the CSD

The procedures to collect cocrystal data from the CSD are the same as those presented in Refs. 1 and 2. Entries containing two distinct chemical residues that were organic, not ionic or polymeric, error-free and had their three-dimensional coordinates determined (including disorder) were searched in the CSD (v5.40). The individual constituents of the entries were then found by converting their structure data files (SD) to canonical SMILES strings with OpenBabel [3]. These molecular representations are unique, take aromaticity and chirality into account, and are easily converted into circular fingerprint vectors and molecular graphs. Cocrystals were filtered by comparing each constituent to a predefined list of common solvents and gasses (lists are made available in Ref. 2). In the case of racemic cocrystals (*i.e.* a cocrystal containing an achiral coformer and a pair of enantiomers of a chiral coformer), a single enantiomer is retained to prevent potential overfitting and unfair reprediction in later stages of model developement. The data set was further restricted to cocrystals with correctly determined explicit valencies of both coformers, as this was a necessary requirement for further processing to fingerprints and molecular graphs. Furthermore, only coformers with up to 60 heavy atoms (approx. 75% of coformers) were included, easing the conversion to the proposed data formats and focusing the set on cocrystals with directed interactions between relatively small molecules. These procedures resulted in a dataset of 8050 cocrystals, formed by 5334 unique coformers.

Prior to selecting the coformers on the basis of their explicit valencies and size, the cocrystals were converted to a physical network and stored in an adjacency matrix A, for which the row and column indices correspond to the coformers. Combinations of coformers for which a cocrystal is known in the CSD are labeled in A as 1, and 0 if undetermined. Such a matrix may be used to predict missing cocrystals with link-prediction algorithms [2], assigning large score values to combinations likely to interact based on network principles. Conversely, combinations that are highly unlikely are given smaller score values, which is exploited here to generate an invalid cocrystal set.

For coformers in the valid cocrystal set (5334 coformers), having more than five determined cocrystals in the CSD, all possible combinations were evaluated with the bipartite resource allocation index (detailed explanation in Ref. 2), storing those for which the score $\equiv 0$, corresponding to invalid coformer combinations. 8050 random samples were taken from this set, forming the invalid cocrystal set.

S2 Coformer featurization

When applying deep learning to molecular data, the input (in this case a pair of coformers) should be provided in the form of a chemical representation. A wide variety of such representations exist [4], including SMILES strings [5], molecular descriptor vectors [6], key-based [7] and circular [8] (*i.e.* extended-connectivity) fingerprint vectors, and molecular graphs [9]. Extended-connectivity (or radial; Morgan-type) fingerprints were generated from canonical SMILES strings of the coformers using a DeepChem [10] wrapper-function for RDKit [11] (DeepChem v2.3.0 with GPU-enabled support, installed for Python v3.5.6). The radius and length of the fingerprints were not *a priori* set to fixed values but assumed to be configurational parameters of the model, meaning that the featurization to fingerprints is different per FP-model.

Similarly, the canonical SMILES of the coformers were transformed to molecular graphs with DeepChem. Molecules are characterized by an $N_{atoms} \times N_{atoms}$ adjacency matrix, containing the connectivities between the atoms, and an $N_{atoms} \times N_{features}$ feature matrix (Figure 2) describing the features of each atom. As features are mostly categorical of nature (Table S1), the feature vector is one-hot encoded to allow for their further processing with machine learning techniques. Therefore, each feature is transformed into an array with a length equal to its number of choices. Each option is given its own bit in the array, which is set to 1 when present. For example, if one desires to encode the atom type and has three options (*e.g.* carbon, oxygen or nitrogen), then carbon corresponds to $[1, 0, 0]$, oxygen to $[0, 1, 0]$ and nitrogen to $[0, 0, 1]$. Finally, the one-hot encoded bit vectors of all properties are joined together, forming the atomic feature vector (of length 78).

Although these two representations are technically two-dimensional in the sense that no atomic coordinates are taken into account, the inclusion of for example the hybridization state of the atoms and optionally their chirality results in the subtle presence of three-dimensional information. Therefore, these molecular representations seem to be very suitable for the purpose of cocrystal prediction.

Table S1: Atom features that are encoded in the feature matrix. The length corresponds to the number of bits occupied by the specific one-hot encoded feature in the final feature vector (of length 78). Only a single bit is required for boolean properties (*e.g.* aromatic).

Feature	Options	Length
Element type	C, N, O, S, F, Si, P, Cl, Br, Mg, Na, Ca, Fe, As, Al, I, B, V, K, Tl, Yb, Sb, Sn, Ag, Pd, Co, Se, Ti, Zn, H, Li, Ge, Cu, Au, Ni, Cd, In, Mn, Zr, Cr, Pt, Hg, Pb, Unknown	44
Degree	0, 1, 2, ..., 9, 10	11
Implicit valence	0, 1, 2, ..., 5, 6	7
Formal charge	0 or 1	1
Number of radical electrons	0 or 1	1
Hybridization	sp, sp^2, sp^3, sp^{3}d or sp^{3}d^2	5
Aromatic	True or False	1
Number of hydrogens	0, 1, 2, 3, 4	5
Chirality	R, S	2
Chirality possible	True or False	1
		$\Sigma = 78$

S3 Model implementation and selection

S3.1 Implementation details

Both model types (Figure 1a and 1b) were implemented in Python (v3.5.6) with Keras [12] and DeepChem [10], and are divided into 4 modules (Table S2). Each model first preprocesses the two coformers in a shared manner (modifying each coformer in the same way) and afterwards merges them into a learnable, united cocrystal vector. The latter is then further processed through a sequence of hidden layers and used for the final prediction. In fact, each hidden layer consists of a sequence of a fully-connected (or dense) layer, a batch normalization layer [13] and a dropout layer [14]. Several of such layer sequences may be present within the module.

Fully-connected (or *dense*) layers receive as input a weighted linear combination of the output from all nodes in the previous layer. After subtraction of a bias term, the input is passed though a non-linear activation function and a single outcome is, after being processed through the batch normalization layer, transmitted to the next layer sequence, repeating the same computation. During the training phase, the model is initialized with random weights and bias terms, and is subsequently exposed to batches of labeled (or known) training data, for which predictions are generated. The error on these predictions contributes to a loss function, which is simultaneously minimized by adjusting (or *learning*) the model's weights while cycling over the available training data (supervised learning). The model weights are adjusted in the direction opposite to the loss gradients with the *backpropagation* algorithm [15].

For the preprocessing module of the molecular graph-based model type, the fully-connected layer is swapped for a graph convolutional layer (open-source implementations of Altae-Tran et al. [16]). The molecular graphs are passed through a series of learnable convolution layers, updating their node features with those of their local chemical environment with each convolutional pass. This creates both a tunable and hierarchical representation of the molecule [17], which, after transformation into a one-dimensional array, is combined into a cocrystal vector and used as input for the abovementioned neural networks containing only hidden layers. In unreported results, dropout for such layers did not appear to affect the training outcome and was therefore omitted. Also, after a batch normalization layer, the feature vector of each node of the graph is pooled, updating the features with the maximum activation across itself and its neighbors. At the end of the preprocessing, the molecular graph of each coformer is condensed into a one-dimensional array by passing it through a graph gathering layer, after which both coformers are merged.

Both model types process through batches of training data and are optimized with Adam optimizer [18] (learning rate=0.001, β_1=0.9, β_2=0.999). The model training was performed on an Intel© Core™ i9-7940X (CPU) and an NVIDIA GeForce© RTX 2080 SUPER™ (GPU). The training of one FP model took approximately one minute and that of a GCN model around ten minutes, resulting in a total training time of a little less than one hour for the ten-membered ensemble.

S3.2 Selection of the model configurations

The number of configurational parameters for both model types that are adjustable is quite large (Table S2). As each parameter is variable and modules can consist of multiple layers, each with independently defined sizes, the space of possible model configurations becomes too large to manually search for an optimum. Therefore, sequential model-based optimization techniques such as Bayesian optimization [19] provide a convenient tool to explore the large parameter space for the best model configurations. For this purpose, we used the Python package Hyperopt [20, 21].

The procedure for finding the optimal configurations is iterative and identical for both model types. At the start of each iteration, the available cocrystal data set is randomly split into a training (90%) and validation (10%) set. Next, a model (defined by the combination of a model type and set of configurational parameters) is initialized with a configuration chosen by the

Table S2: Configurational parameters and their possible values for both model types.

Model	FP model	GCN model
Fingerprint size (2^x)	$x \in \{7, 8, ..., 11, 12\}$	-
Fingerprint radius (r)	$r \in \{1, 2, 3\}$	-
Batch size (2^b)	$b \in \{6, 7, 8\}$	7
	Preprocessing module	
Layer type	Dense	Graph Convolution
Layer size (2^x)	$x \in \{7, 8, 9, 10, 11\}$	$x \in \{6, 7, 8\}$
Layer activation function	ReLU, ELU or Tanh	ReLU, ELU or Tanh
Layer dropout (d)	$d \in [0, 0.75]$	0
Number of layers	1,2 or 3	1,2 or 3
Graph gathering activation function	-	ReLU, ELU or Tanh
	Merging module	
Vector operation	Add or Concatenate	Add or Concatenate
	Feedforward module	
Layer type	Dense	Dense
Layer size (2^x)	$x \in \{7, 8, 9, 10, 11\}$	$x \in \{6, 7, 8, 9, 10\}$
Layer activation function	ReLU, ELU or Tanh	ReLU, ELU or Tanh
Layer dropout (d)	$d \in [0, 0.75]$	$d \in [0.1, 0.6]$
Number of layers	1,2 or 3	1,2 or 3
	Predictive module	
Layer type	Dense	Dense
Layer size	1	2
Layer activation function	Sigmoid	Softmax
Loss function	Binary cross entropy	Softmax cross entropy

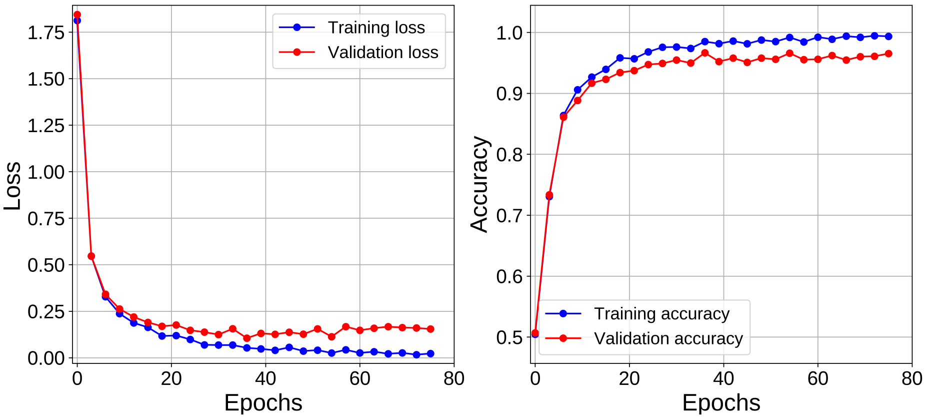

Figure S1: Example of training a GCN model. (left) The loss on both the training and validation set quickly decreases with an increasing number of epochs. (right) Evolution of the model's accuracy.

Bayesian optimizer, and is trained on the provided training set. The so-called *loss* metric of the model, related to how well training samples are being classified, is recorded during training for both the training and validation set, the latter containing data that remains unseen by the model during training. This loss metric (*i.e.* average log loss or cross-entropy loss) is defined as $L = -\frac{1}{N}\left[\sum_{i=1}^{N} t_i \log o_i + (1-t_i)\log(1-o_i)\right]$, where t_i is the truth or true label (*i.e.* 0 or 1) of sample i and o_i is the output of the classifier ($\in [0,1]$). FP models were allowed to train over 30 epochs (*i.e.* the number of times the entire training data set is worked through) and GCN models over 75.

An example of such a training cycle is shown in Figure S1, illustrating the evolution of a GCN model's loss to near zero for the training set (blue dots; perfect classification) as the number of epochs increases. Likewise, the model's training accuracy evolves to near 100%. The model thus consistently improves itself in discriminating true cocrystals from invalid ones in the training set. This behavior is also reflected in the performance metrics on the validation set, which the model is also able to classify well. Unlike the training set, the improvement stagnates around 40 epochs, indicating the onset of overfitting. The excellent classification performance on the external validation set is only possible if the training set covers characteristic patterns for cocrystallization in the validation set (and thus the entire data set) and if these patterns are effectively encoded in the neural network's internal parameters. Hence, it is confirmed that the proposed data set forms an adequate basis for cocrystal prediction.

When the training cycle is finished, the minimum loss on the validation set is returned to the Bayesian optimizer (epoch 36 in Figure S1), which repeatedly selects the next configuration

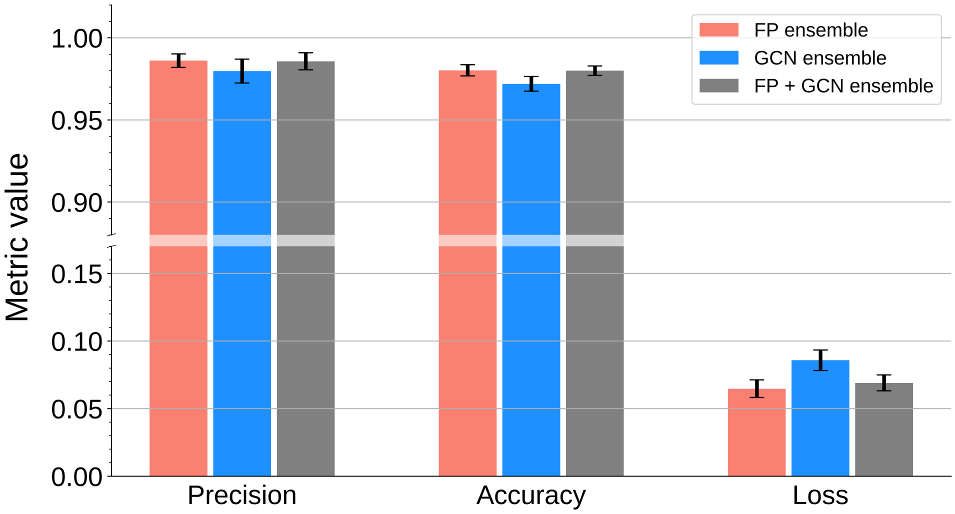

Figure S2: Performance metrics of the ensemble models over a ten-fold cross-validation experiment. The height of the bars shows the average value on the ten validation sets. The black error bars correspond to ± one unit of standard deviation.

to be tested based on previous evaluations by optimizing the Expected Improvement with the Tree-structured Parzen Estimator approach (TPE) [22]. Initially, this procedure is run for three random configurations from the space, which the optimizer uses as a starting point. After 50 iterations, a ranked list of possible model configurations and their associated performance on data set aside is produced for each model type, which was used to select the five best models for each type (Tables S3 and S4). These models were placed in their individual model ensembles and a combined model ensemble, containing all ten models.

S4 Model validation

S4.1 Cross-validation

The performance of the FP, GCN, and FP + GCN model ensembles was validated on the available cocrystal data by ten-fold cross-validation. The data set was first randomly divided in ten equal parts or folds. Each fold is used once for validation and nine times for training, and the performance on each of the validation sets is recorded while training on the data from residual nine sets. To prevent overfitting, the FP models and GCN models were trained for 30 and 50 epochs, respectively. The average precision, accuracy and loss values over ten folds were computed for the model ensembles and are shown in Figure S2.

Table S3: Configurational parameters (top) and performance metrics (bottom) of the five best **FP models**. The models are ordered by their increasing loss. The epoch indicates the point where the smallest loss was observed.

Model	Radius	Fingerprint size	pr_1	pr_2	pr_3	drop_{pr}	act_{pr}	merge	ff_1	ff_2	ff_3	drop_{ff}	act_{ff}	Batch size
1	1	1024	1024	512	128	0.373	ReLU	Concat.	512			0.358	ELU	128
2	2	2048	2048			0.409	ReLU	Add	128	256		0.230	ReLU	64
3	3	2048	1024	1024	2048	0.675	ReLU	Add	128			0.559	ELU	256
4	2	4096	2048			0.285	ReLU	Concat.	256	256		0.052	ReLU	64
5	2	2048	512	512	1024	0.392	ReLU	Add	2048			0.592	ELU	64

Model	Loss	Accuracy	Epoch
1	0.072	0.972	14
2	0.087	0.971	28
3	0.088	0.965	28
4	0.089	0.971	8
5	0.089	0.974	10

Table S4: Configurational parameters (top) and performance metrics (bottom) of the five best **GCN models**. The models are ordered by their increasing loss. The epoch indicates the point where the smallest loss was observed.

Model	pr_1	pr_2	pr_3	$\mathrm{drop}_{\mathrm{pr}}$	$\mathrm{act}_{\mathrm{pr}}$	Merge	ff_1	ff_2	ff_3	$\mathrm{drop}_{\mathrm{ff}}$	$\mathrm{act}_{\mathrm{ff}}$	$\mathrm{act}_{\mathrm{gather}}$	Batch size
1	64	256		0	Tanh	Add	512	512	512	0.410	ReLU	ReLU	128
2	64	256		0	ReLU	Add	512	512	512	0.409	ReLU	ReLU	128
3	128	256		0	ReLU	Add	1024	1024	1024	0.409	ReLU	Tanh	128
4	128	256		0	Tanh	Add	512	1024	512	0.424	ReLU	ReLU	128
5	256			0	ReLU	Concat.	512	1024		0.340	ReLU	Tanh	128

Model	Loss	Accuracy	Epoch
1	0.095	0.968	51
2	0.099	0.971	45
3	0.104	0.963	63
4	0.105	0.966	36
5	0.106	0.964	57

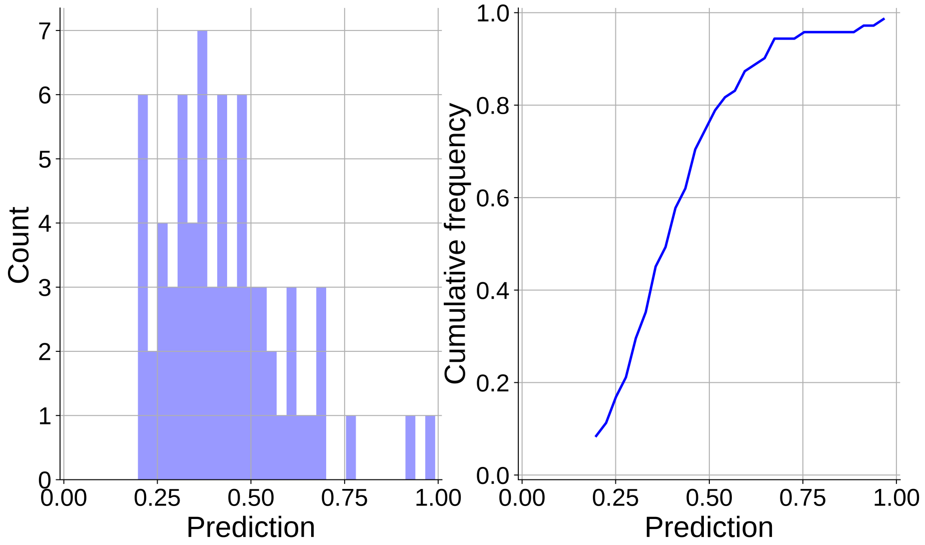

Figure S3: Score histogram (left) and cumulative distribution function (right) of the 71 invalid combinations for carbamazepine.

S4.2 Additional invalid coformer pairs for carbamazepine

Besides the four invalid cocrystal combinations that were already in our data set for carbamazepine, we found an additional 67 eligible couples according to the procedures mentioned in section S1. The models were again trained on all data points except for the carbamazepine combinations, and the prediction values for all invalid pairs are shown in Figure S3. Clearly, most combinations (approx. 78%) are scored below the 0.5 threshold, and are therefore assumed to be non-existing. Although our set of invalid cocrystals is to a certain degree artificial and lacks real experimental evidence, its usefulness is thus demonstrated by Figure S3, as well as by the high precision values (and therefore small occurence of false positives) presented in Figure S2.

S5 Coformer clustering

The clustering of the coformers was done with Ward's hierarchical clustering method [23], according to the procedures described in Ref. 1. The adjacency matrix of the coformer network (see section S1), which essentially contains all determined cocrystals in the CSD, is transformed into a similarity matrix, where coformer similarity for a pair of coformers i and j is defined with the Jaccard index [24]:

$$s_{i,j} = \frac{n_i \cap n_j}{n_i \cup n_j}, \tag{1}$$

where the neighbors n_i of coformer i can be found from the set of nodes (or coformers) N as:

$$n_i = \{j \epsilon N | A_{i,j} = 1\}. \tag{2}$$

For the purpose of hierarchical clustering, the similarity matrix was converted into dissimilarity matrix ($d_{i,j} = 1 - s_{i,j}$) and was resized to include only the 75 most popular coformers. These were determined based on their degree ($= |n_i|$) in the adjacency matrix. Ward's clustering method works in an agglomerative fashion, repeatedly merging coformers or clusters thereof that are least dissimilar (or closest/most similar) in larger clusters. The method starts by placing all coformers in separate clusters or *singletons*, which were subsequently agglomerated into clusters. The distance to a cluster p containing multiple coformers is calculated as:

$$d(p,q) = \sqrt{\frac{|q|+|s|}{|q|+|s|+|t|}d(q,s)^2 + \frac{|q|+|t|}{|q|+|s|+|t|}d(q,t)^2 - \frac{|q|}{|q|+|s|+|t|}d(s,t)^2} \tag{3}$$

where p is the cluster as a result from merging clusters s and t, and q is one of the remaining clusters. The distances at which two clusters were joined were recorded and illustrated as a dendrogram (Figure 4).

S6 Experimental procedures for cocrystal synthesis and characterization

RS-ketoprofen (50 mg; TCI Europe NV, $>$ 98% pure) and carbamazepine (51 mg; Aldrich, $\leq$ 100% pure) were ground in the presence of 40 μL acetonitrile (*i.e.* liquid-assisted grinding, LAG) for 30 minutes at 25 Hz with a Retsch MM 400. The same procedure was followed for S-ketoprofen (48 mg; Sigma Aldrich, 99% pure) and carbamazepine (53 mg). White powders were harvested, which were subsequently analyzed by powder X-ray diffraction (PXRD). For powder diffraction analysis, samples were sealed in a 0.3 mm soda lime glass capillary. The diffractograms were measured in capillary mode on a Panalytical Empyrean diffractometer with CuKα radiation from a sealed LFF tube and a PIXcel3D 1x1 detector. The powder patterns of the obtained mixtures (shown in Figures S4 and S5) were different from their constituents and known polymorphs in the CSD, indicating the formation of two new phases. Both patterns are extremely similar, indicating that the racemic and enantiopure cocrystal are likely to have quasi-identical unit cell parameters.

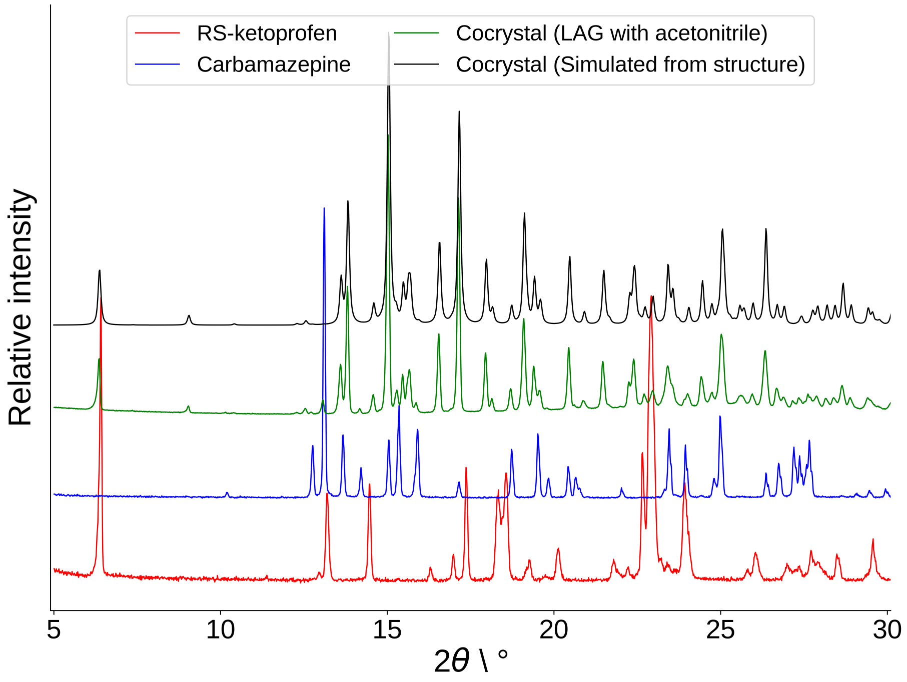

Figure S4: Powder diffraction patterns of RS-ketoprofen (red), carbamazepine (blue), and the co-ground powder (green). The simulated PXRD pattern of the cocrystal structure (black) confirms that the phase obtained by LAG is the cocrystal.

Approximately 10 mg of the co-ground powder with racemic ketoprofen was dissolved in 1 mL of methanol and left to slowly evaporate, yielding colourless block-like crystals suitable for single-crystal X-ray diffraction after 4 days. Single crystals of the enantiopure cocrystal were obtained by slow evaporating a solution containing 45 mg of a 1:1 mixture of carbamazepine and S-ketoprofen (95% pure, Fluorochem), and 1 mL of ethanol:ethyl acetate (3:7 v/v). The latter were found together with crystals of pure carbamazepine and a glue-like liquid.

The structures of the two cocrystals are presented in Figure S6 and S7. The simulated powder diffraction patterns of these structures, corrected for thermal expansion at room temperature, are also displayed in Figures S4 and S5, confirming that the phases obtained by LAG are indeed the discovered cocrystals. Hydrogen bonding details (Tables S5 and S6) and the crystallographic data (Table S7) of the cocrystal structures are presented below.

* : Reflections were measured on a Bruker D8 Quest diffractometer with sealed tube and Triumph monochromator ($\lambda = 0.71073$ Å). Software package used for the intensity integration was Saint (v8.38a, Bruker AXS Inc., Madison, Wisconsin, USA). Absorption correction was performed with SADABS [25]. The structures were solved with direct methods using SHELXT [26].

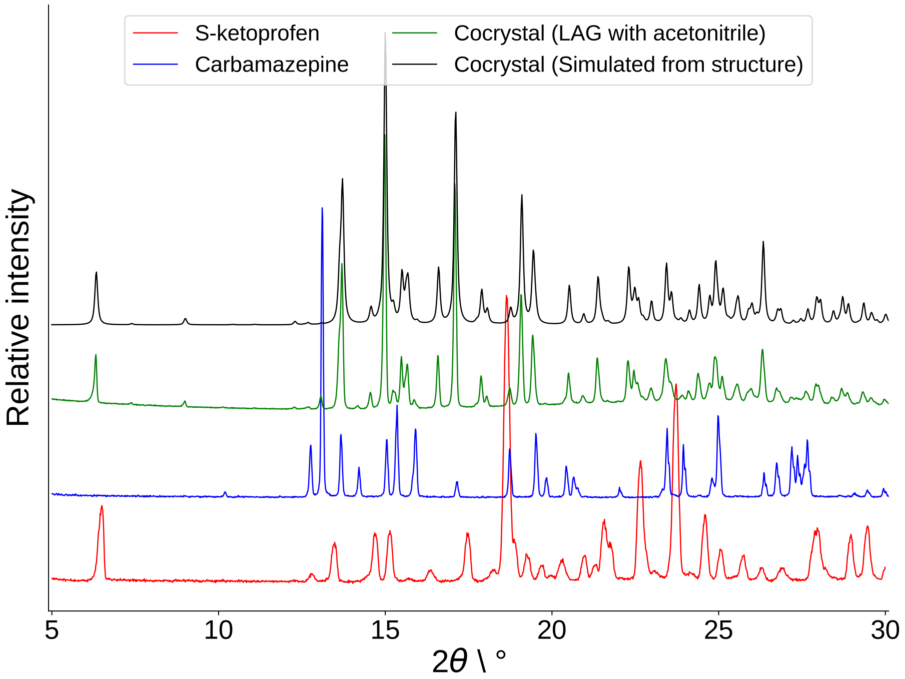

Figure S5: Powder diffraction patterns of S-ketoprofen (red), carbamazepine (blue), and the co-ground powder (green). Comparison of the experimental pattern to that of the structure (black) confirms that obtained material is the binary cocrystal.

Table S5: H-bonding details of the racemic cocrystal (p2033a). Besides the interaction between the acid and amide groups on the coformers, involving one hydrogen (H21A) of the amide group, the ketone group of ketoprofen (C13 = O03) forms a hydrogen bond to the second hydrogen (H21B) on the amide.

Donor	— H	....Acceptor	ARU	D - H	H...A	D...A	D - H...A
O01	–H01	..O20A	1555.02	0.972(17)	1.598(17)	2.5572(14)	168.7(13)
N21A	–H21A	..O02	1555.01	0.88	2.14	2.9823(16)	161
N21A	–H21B	..O03	4565.01	0.88	2.32	3.1454(15)	156

Table S6: H-bonding details of the enantiopure cocrystal (p2059a).

Donor	— H	....Acceptor	ARU	D - H	H...A	D...A	D - H...A
O03	–H03	..O04	1555.04	0.97(4)	1.60(4)	2.557(3)	172(4)
O06	–H06	..O01	1555.03	0.81(5)	1.82(5)	2.613(3)	168(4)
N11	–H11A	..O02	1555.01	0.89(4)	2.12(4)	2.995(3)	168(3)
N11	–H11B	..O08	1555.02	0.91(3)	2.24(3)	3.120(3)	162(3)
N12	–H12A	..O07	1555.02	0.88(4)	2.06(4)	2.928(3)	168(3)
N12	–H12B	..O05	1454.01	0.89(3)	2.30(4)	3.141(3)	157(3)
C74	–H74C	..O02	2656.01	0.98	2.60	3.489(4)	151

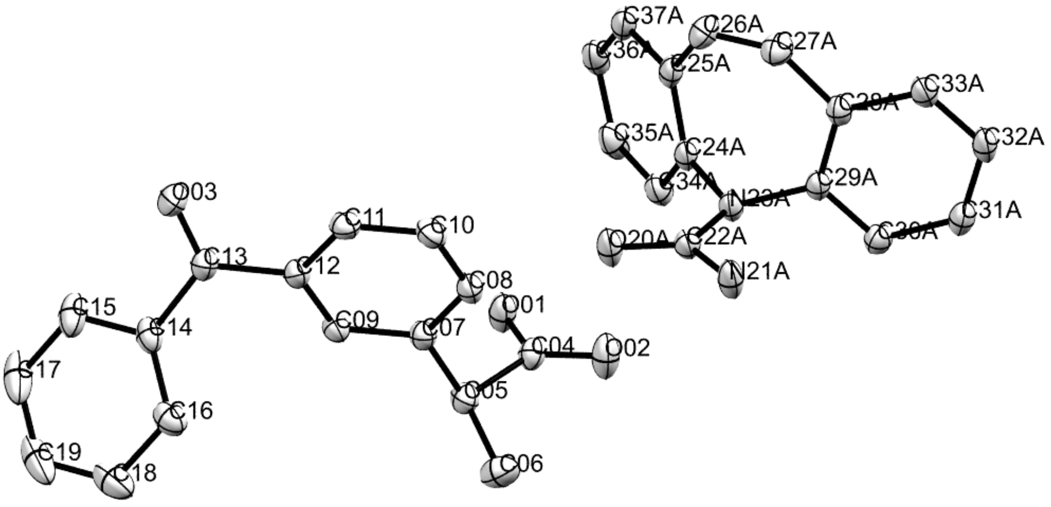

Figure S6: ORTEP plot of the racemic cocrystal (p2033a). As the structure is disordered, only the major conformation ('A') of carbamazepine is shown.

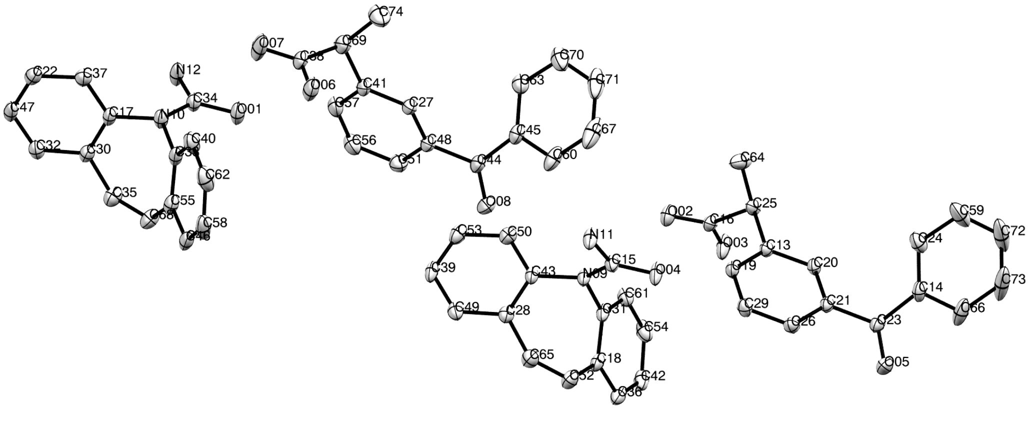

Figure S7: ORTEP plot of the enantiopure cocrystal (p2059a).

Table S7: Crystallographic data[*] of the cocrystals containing RS-ketoprofen and carbamazepine (racemic), and S-ketoprofen and carbamazepine (enantiopure).

Crystal data	Racemic (p2033a)	Enantiopure (p2059a)
CCDC no.	2010395	2010396
Chemical Formula	$C_{16}H_{14}O_3 \cdot C_{15}H_{12}N_2O$	$C_{16}H_{14}O_3 \cdot C_{15}H_{12}N_2O$
M_r	490.54	490.54
Crystal system, space group	Monoclinic, $P2_1/n$	Monoclinic, $P2_1$
Temperature (K)	150	150
a, b, c (Å)	16.9943 (7), 7.7147 (3), 19.6426 (8)	17.0106 (9), 7.7191 (5), 19.7334 (12)
β (°)	97.9814 (16)	98.375 (2)
V (Å^3)	2550.32 (18)	2563.5 (3)
Z	4	4
Radiation type	Mo$K\alpha$	Mo$K\alpha$
μ (mm^{-1})	0.09	0.09
Crystal size (mm)	0.33 × 0.30 × 0.10	0.39 × 0.26 × 0.08
Data collection		
Diffractometer	Bruker D8 Quest Apex3	
Absorption correction	Multi-scan *SADABS* 2016/2: Krause, L., Herbst-Irmer, R., Sheldrick G.M. & Stalke D., J. Appl. Cryst. 48 (2015) 3-10	
T_{min}, T_{max}	0.712, 0.746	0.715, 0.747
No. of measured, independent and observed $[I > 2\sigma(I)]$ reflections	37024, 6340, 5535	122742, 19508, 17490
R_{int}	0.024	0.036
$(\sin\theta/\lambda)_{\max}$(Å^{-1})	0.667	0.770
Refinement		
$R[F^2 > 2\sigma(F^2)], wR(F^2), S$	0.040, 0.110, 1.05	0.062, 0.168, 1.05
No. of reflections	6340	19508
No. of parameters	392	687
No. of restraints	48	1
H-atom treatment	H atoms treated by a mixture of independent and constrained refinement	
$\Delta\rho_{max}, \Delta\rho_{min}$ (eÅ^{-3})	0.26, -0.24	0.73, -0.33
Special remarks	Carbamazepine is disorded.	

Least-squares refinement was performed with SHELXL-2014 [27] against $|F_h^0|^2$ of all reflections. Non-hydrogen atoms were refined freely with anisotropic displacement parameters. Hydrogen atoms were placed on calculated positions or located in difference Fourier maps. All calculated hydrogen atoms were refined with a riding model.

References

[1] J. J. Devogelaer, H. Meekes, E. Vlieg, R. de Gelder, *Acta Crystallographica Section B: Structural Science Crystal Engineering and Materials* **2019**, *75*, 371–383.

[2] J.-J. Devogelaer, S. J. T. Brugman, H. Meekes, P. Tinnemans, E. Vlieg, R. d. Gelder, *CrystEngComm* **2019**, *21*, 6875–6885.

[3] N. M. O'Boyle, M. Banck, C. A. James, C. Morley, T. Vandermeersch, G. R. Hutchison, *Journal of Cheminformatics* **2011**, *3*, 33.

[4] R. Todeschini, V. Consonni, R. Mannhold, H. Kubinyi, G. Folkers, *Molecular Descriptors for Chemoinformatics: Volume I: Alphabetical Listing / Volume II: Appendices, References*, Wiley, **2009**.

[5] D. Weininger, *Journal of Chemical Information and Computer Sciences* **1988**, *28*, 31–36.

[6] A. Mauri, V. Consonni, M. Pavan, R. Todeschini, *MATCH* **2006**, *56*, 237–248.

[7] J. L. Durant, B. A. Leland, D. R. Henry, J. G. Nourse, *Journal of Chemical Information and Computer Sciences* **2002**, *42*, 1273–1280.

[8] D. Rogers, M. Hahn, *Journal of Chemical Information and Modeling* **2010**, *50*, 742–754.

[9] D. K. Duvenaud, D. Maclaurin, J. Iparraguirre, R. Bombarell, T. Hirzel, A. Aspuru-Guzik, R. P. Adams in *Advances in Neural Information Processing Systems 28* (Eds.: C. Cortes, N. D. Lawrence, D. D. Lee, M. Sugiyama, R. Garnett), Curran Associates, Inc., **2015**, pp. 2224–2232.

[10] B. Ramsundar, "DeepChem", can be found under https://deepchem.io/.

[11] G. A. Landrum, "RDKit: Open Source Cheminformatics", can be found under http://www.rdkit.org.

[12] F. Chollet, "Keras", can be found under https://keras.io/.

[13] S. Ioffe, C. Szegedy, **2015**, arXiv preprint arXiv:1502.03167 [cs].

[14] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, R. Salakhutdinov, *Journal of Machine Learning Research* **2014**, *15*, 1929–1958.

[15] D. E. Rumelhart, G. E. Hinton, R. J. Williams, *Nature* **1986**, *323*, 533–536.

[16] H. Altae-Tran, B. Ramsundar, A. S. Pappu, V. Pande, *ACS Central Science* **2017**, *3*, 283–293.

[17] Y. Bengio, A. Courville, P. Vincent, *IEEE Transactions on Pattern Analysis and Machine Intelligence* **2013**, *35*, 1798–1828.

[18] D. P. Kingma, J. Ba, **2017**, arXiv preprint arXiv:1412.6980 [cs].

[19] B. Shahriari, K. Swersky, Z. Wang, R. P. Adams, N. de Freitas, *Proceedings of the IEEE* **2016**, *104*, 148–175.

[20] J. Bergstra, D. Yamins, D. D. Cox, **2012**, arXiv preprint arXiv:1209.5111 [cs].

[21] J. Bergstra, "Hyperopt: Distributed Hyperparameter Optimization", can be found under https://github.com/hyperopt/hyperopt.

[22] J. Bergstra, R. Bardenet, Y. Bengio, B. Kégl, *Proceedings of the 24th International Conference on Neural Information Processing Systems*, USA, **2011**, pp. 2546–2554.

[23] J. H. J. Ward, *Journal of the American Statistical Association* **1963**, *58*, 236–244.

[24] P. Jaccard, *New Phytologist* **1912**, *11*, 37–50.

[25] L. Krause, R. Herbst-Irmer, G. M. Sheldrick, D. Stalke, *Journal of Applied Crystallography* **2015**, *48*, 3–10.

[26] G. M. Sheldrick, *Acta Crystallographica Section A: Foundations and Advances* **2015**, *71*, 3–8.

[27] G. M. Sheldrick, *Acta Crystallographica Section C: Structural Chemistry* **2015**, *71*, 3–8.