nature machine intelligence

Article

https://doi.org/10.1038/s42256-022-00577-2

Joint structural annotation of small molecules using liquid chromatography retention order and tandem mass spectrometry data

Received: 23 February 2022

Accepted: 31 October 2022

Published online: 19 December 2022

Check for updates

Eric Bach [1] ✉, Emma L. Schymanski [2] & Juho Rousu [1] ✉

Structural annotation of small molecules in biological samples remains a key bottleneck in untargeted metabolomics, despite rapid progress in predictive methods and tools during the past decade. Liquid chromatography–tandem mass spectrometry, one of the most widely used analysis platforms, can detect thousands of molecules in a sample, the vast majority of which remain unidentified even with best-of-class methods. Here we present LC-MS^{2}Struct, a machine learning framework for structural annotation of small-molecule data arising from liquid chromatography–tandem mass spectrometry (LC-MS^2) measurements. LC-MS^{2}Struct jointly predicts the annotations for a set of mass spectrometry features in a sample, using a novel structured prediction model trained to optimally combine the output of state-of-the-art MS^2 scorers and observed retention orders. We evaluate our method on a dataset covering all publicly available reversed-phase LC-MS^2 data in the MassBank reference database, including 4,327 molecules measured using 18 different LC conditions from 16 contributors, greatly expanding the chemical analytical space covered in previous multi-MS^2 scorer evaluations. LC-MS^{2}Struct obtains significantly higher annotation accuracy than earlier methods and improves the annotation accuracy of state-of-the-art MS^2 scorers by up to 106%. The use of stereochemistry-aware molecular fingerprints improves prediction performance, which highlights limitations in existing approaches and has strong implications for future computational LC-MS^2 developments.

Structural annotation of small molecules in biological samples is a key bottleneck in various research fields including biomedicine, biotechnology, drug discovery and environmental sciences. Samples in untargeted metabolomics studies typically contain thousands of different molecules, the vast majority of which remain unidentified[1–3]. Liquid chromatography–tandem mass spectrometry (LC-MS^2) is one of the most widely used analysis platforms[4], as it allows for high-throughput screening, is highly sensitive and is applicable to a wide range of molecules. In LC-MS^2, molecules are first separated by their different physicochemical interactions between the mobile and stationary phase of the column in the liquid chromatographic system, resulting in retention time (RT) differences. Subsequently, they are separated

[1]Department of Computer Science, Aalto University, Espoo, Finland. [2]Luxembourg Centre for Systems Biomedicine (LCSB), University of Luxembourg, Belvaux, Luxembourg. ✉e-mail: eric.bach@aalto.fi; juho.rousu@aalto.fi

according to their mass-to-charge ratio in a mass analyser (MS^1). Finally, the molecular ions are isolated and fragmented in the tandem mass spectrometer (MS^2).

For each ion, the recorded fragments and their intensities constitute the MS^2 spectrum, which contains information about the substructures in the molecule and serves as a basis for annotation efforts. In typical untargeted LC-MS^2 workflows, thousands of MS features (MS^1, MS^2, RT) arise from a single sample. The goal of structural annotation is to associate each feature with a candidate molecular structure, for further downstream interpretation.

In recent years, many powerful methods[5,6] to predict structural annotations for MS^2 spectra have been developed[7–18]. In general, these methods find candidate molecular structures potentially associated with the MS feature, for example, by querying molecules with a certain mass from a structure database such as Human Metabolome Database (HMDB)[19] or PubChem[20] and subsequently computing a match score between each candidate and the MS^2 spectrum. The highest-scoring candidate is typically considered as the structure annotation of a given MS^2. Currently, even the best-of-class methods only reach an annotation accuracy of around 40% (ref. [17]) in evaluations when searching large candidate sets such as those retrieved from PubChem. Therefore, in practice, a ranked list of molecular structures is provided to the user (for example, the top-20 structures). This level of performance is still a considerable hindrance in metabolomics and other fields.

Interestingly, RT information remains underutilized in automated approaches for structure annotation based on MS^2, despite RTs being readily available in all LC-MS^2 pipelines and generally recognized as contributing valuable information[21,22]. An explanation is that a molecule generally has different RTs under different LC conditions (mobile phase, column composition and so on)[23,24]. Typically, the RT information is used for post-processing of candidate lists, for example, by comparing measured and reference standard RTs[3,24]. This approach, however, is limited by the availability of experimentally determined RTs of reference standards. RT prediction models[24,25], however, allow the prediction of RTs based solely on the molecular structure of the candidate, and have been successfully applied to aid structure annotation[11,26–29]. However, such prediction models generally have to be calibrated to the specific LC configuration[3], requiring at least some amount of target LC reference standard RT data to be available[21,29,30]. Recently, the idea of predicting retention orders (ROs), that is, the order in which two molecules elute from the LC column, has been explored[31–34]. ROs are largely preserved within a family of LC systems (for example, reversed-phase or hydrophilic interaction LC systems). Therefore, RO predictors can be trained using a diverse set of RT reference data, and applied to out-of-dataset LC set-ups[31]. Integration of MS^2- and RO-based scores using probabilistic graphical models improved the annotation performance in LC-MS^2 experiments[34].

Another somewhat neglected aspect in automated annotation pipelines is the treatment of stereochemistry, that is, the different three-dimensional (3D) variants of the molecules. The general assumption has been that LC-MS^2 data do not contain sufficient information to separate stereoisomers in samples[5,24]. As a result, MS^2 scorers typically disregard the stereochemical information in the candidate structures and often output the same matching for different stereoisomers (compare refs. [7,17]). However, stereoisomers that vary in their double-bond orientation (for example, *cis*–*trans* or *E*–*Z* isomerism) may have different shapes and thus exhibit different fragmentation and/or interactions with the LC system. Thus, ignoring stereochemistry in candidate processing may disregard LC-relevant stereochemical information. Furthermore, it is known that certain stereochemical configurations occur more frequently than others in nature and hence in the reference databases. Making use of such information can potentially improve annotation performance.

In this Article, we set out to provide a new perspective on jointly using MS^2 and RO combined with stereochemistry-aware molecular features for the structure annotation of LC-MS^2 data. We present a novel machine learning framework called LC-MS^2Struct, which learns to optimally combine the MS^2 and RO information for the accurate annotation of a sequence of MS features. LC-MS^2Struct relies on the structured support vector machine (SSVM)[35] and max-margin Markov network[36] frameworks. In contrast to the previous work of ref. [34], our framework does not require a separately learned RO prediction model. Instead, it optimizes the SSVM parameters such that the score margin between correct and any other sequence of annotations is maximized. This way, LC-MS^2Struct learns to optimally use the RO information from a set of LC-MS^2 experiments. We trained and evaluated LC-MS^2Struct on all available reversed-phase LC data from MassBank[37], including a combined total of 4,327 molecules from 18 different LC configurations, hence reaching a high level of measurement diversity in the model evaluation. Our framework is compared with three other approaches: RT filtering, log*P* predictions[11] and RO predictions[34]. LC-MS^2Struct can be combined with any MS^2 scorer, and is demonstrated with the CFM-ID[9,18], MetFrag[7,11] and SIRIUS[8,17] tools. The use of chirality encoding circular molecular fingerprints[38] in the predictive model allows to distinguish and rank different stereoisomers based on the observed ROs.

Overview of LC-MS^2Struct

Input and output

We consider a typical data setting in untargeted LC-MS^2-based experiments, after pre-processing such as chromatographic peak picking and alignment (Fig. 1a). Such data comprise a sequence of MS features, here indexed by σ. Each feature consists of MS^1 information (for example, mass, adduct and isotope pattern), LC retention time (RT) t_σ and an MS^2 spectrum x_σ. We assume that a set of candidate molecules $\mathcal{C}_\sigma$ is associated with each MS feature σ. Such a set can be, for example, generated from a structure database (for example, PubChem[20], ChemSpider[39] or PubChemLite[40]) based on the ion's mass, a suspect list or an in silico molecule generator (for example, SmiLib v2.0[41,42]). We furthermore require that for MS^2 spectrum x_σ, a matching score $\theta(x_\sigma, m)$ with its candidates $m \in \mathcal{C}_\sigma$ is pre-computed using an in silico tool, such as CFM-ID[9,18], MetFrag[11] or SIRIUS[8,17]. LC-MS^2Struct predicts a score for MS feature σ and each associated candidate $m \in \mathcal{C}_\sigma$ based on a sequence of spectra $\mathbf{x} = (x_\sigma)_{\sigma=1}^{L}$, of length L, and the ROs derived from the observed RTs $\mathbf{t} = (t_\sigma)_{\sigma=1}^{L}$. These scores are used to rank the molecular candidates associated with the MS features (Fig. 1b).

Candidate ranking using max-marginals

We define a fully connected graph $G = (V, E)$ capturing the MS features and modelling their dependencies (Fig. 1c), where V represents the set of nodes and E the set of edges. Each node $\sigma \in V$ corresponds to an MS feature, and is associated with the pre-computed MS^2 matching scores $\theta(x_\sigma, m)$ between the MS^2 spectrum x_σ and all molecular candidates $m \in \mathcal{C}_\sigma$. The graph G contains an edge $(\sigma, \tau) \in E$ for each MS feature pair. A scoring function F is defined predicting a compatibility score between a sequence of molecular structure assignments $\mathbf{y} = (y_\sigma)_{\sigma=1}^{L}$ in the label-space $\Sigma = \mathcal{C}_1 \times \ldots \times \mathcal{C}_L$ and the observed data:

$$F(\mathbf{y} \mid \mathbf{x}, \mathbf{t}, \mathbf{w}, G) = \underbrace{\frac{1}{|V|}\sum_{\sigma \in V} \theta(x_\sigma, y_\sigma)}_{\text{Node scores: } MS^2 \text{ information}} + \underbrace{\frac{1}{|E|}\sum_{(\sigma,\tau)\in E} f((t_\sigma, t_\tau), (y_\sigma, y_\tau) \mid \mathbf{w})}_{\text{Edge scores: RO information}}, \quad (1)$$

where the function f outputs an edge score (Fig. 1d) expressing the agreement between the observed and the predicted RO, for each candidate assignment pair (y_σ, y_τ) given the observed RTs (t_σ, t_τ). The function f is parameterized by the vector $\mathbf{w}$, which is trained specifically for each MS^2 scorer (see next section). Using the compatibility score function F (equation (1)), we compute the max-marginal scores[43] for each candidate and MS feature, defined for a candidate $m \in \mathcal{C}_\sigma$ and MS

https://doi.org/10.1038/s42256-022-00577-2

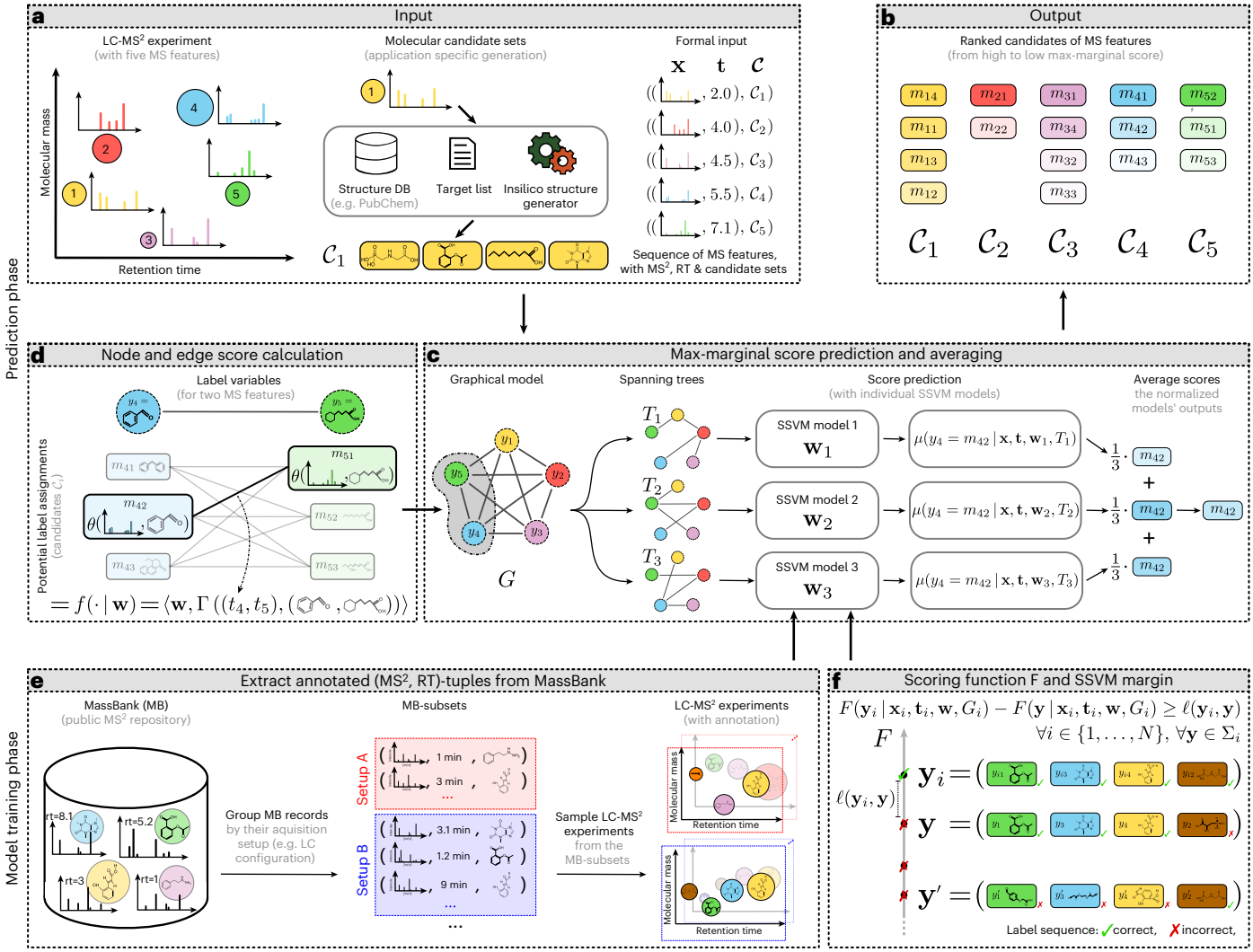


Fig. 1 | Overview of the LC-MS²Struct workflow. a, Input to LC-MS²Struct during the application phase. The LC-MS² experiment results in a set of (MS², RT)-tuples. The MS information is used to generate a molecular candidate set for each MS feature. **b**, The output of LC-MS²Struct is the ranked molecular candidates for each MS feature. **c**, A fully connected graph G models the pairwise dependency between the MS features. Using a set of random spanning trees T_k and SSVM, we predict the max-marginal scores for each candidate used for the ranking. **d**, The MS² and RO information is used to score the nodes and edges in the graph G. **e**, To train the SSVM models and evaluate LC-MS²Struct, we extract MS² spectra and RTs from MassBank. We group the MassBank records such that their experimental set-ups are matching, simulating LC-MS² experiments. **f**, Main objective optimized during the SSVM training, where $\mathbf{y}_i \in \Sigma_i$ is the ground-truth label sequence of example i and $\mathbf{y}, \mathbf{y}' \in \Sigma_i$ are further possible label sequences.

feature σ as the maximum compatibility score that a candidate assignment $\bar{\mathbf{y}} \in \Sigma$ with $\bar{y}_\sigma = m$ can reach:

$$\mu(y_\sigma = m \mid \mathbf{x}, \mathbf{t}, \mathbf{w}, G) = \max_{\{\bar{\mathbf{y}} \in \Sigma \,:\, \bar{y}_\sigma = m\}} F(\bar{\mathbf{y}} \mid \mathbf{x}, \mathbf{t}, \mathbf{w}, G).$$

We use μ to rank the molecular candidates[34]. However, for general graphs G, the max-marginal inference problem (MMAP) is intractable. Therefore, we approximate the MMAP problem by performing the inference on tree-like graphs T_k randomly sampled from G (Fig. 1c), for which exact inference is feasible[43,44]. Here, k indexes the individual spanning trees. Subsequently, we average the max-marginal scores $\mu(y_\sigma = m \mid \mathbf{x}_i, \mathbf{t}_i, \mathbf{w}_k, T_k)$ over a set of trees $\mathbf{T}$, an approach that performed well for practical applications[34,45,46]. Thereby i indexes the individual training MS² spectra and RT sequences. For each spanning tree T_k, we apply a separately trained SSVM model $\mathbf{w}_k$ to increase the diversity of the predictions.

Joint annotation using SSVMs

We propose to tackle the joint assignment of candidate labels $\mathbf{y} \in \Sigma$ to the sequence of MS features of a LC-MS² experiment through structured prediction, a family of machine learning methods generally used to annotate sequences or networks[35,46,47]. In our model, the structure is given by the observed RO of the MS feature pairs (y_σ, y_τ), which provides additional information on the correct candidate labels y_σ and y_τ. Given a set of annotated LC-MS² experiments extracted from MassBank[37] (Fig. 1e), we train an SSVM[35] model $\mathbf{w}$ predicting the edge scores. SSVM models can be optimized using the max-margin principle[35]. In a nutshell, given a set of ground-truth-annotated MS feature sequences, the model parameters $\mathbf{w}$ are optimized such that the correct label sequence $\mathbf{y}_i \in \Sigma_i$, that is, the structure annotations for all MS features in an LC-MS² experiment, scores higher than any other possible label sequence assignment $\mathbf{y} \in \Sigma_i$ (Fig. 1f).

Results

Extracting training data from MassBank

Ground-truth-annotated MS² spectra and RTs were extracted from MassBank[37], a public online database for MS² data. Each individual MassBank record typically provides a rich set of meta-information (Supplementary Table 1), such as the chromatographic and MS

https://doi.org/10.1038/s42256-022-00577-2

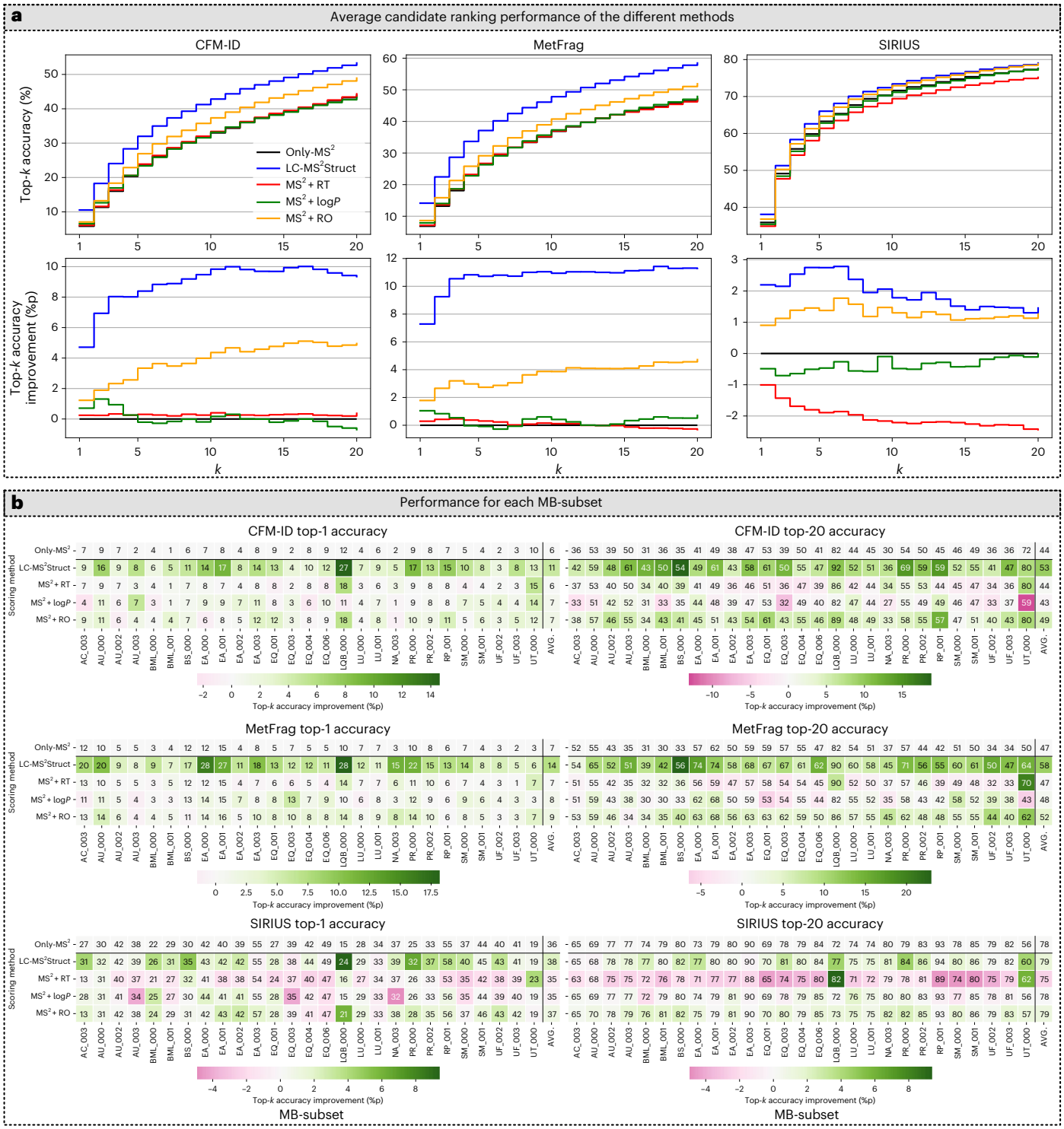


Fig. 2 | Different approaches to combine MS² and RT information. **a**, Comparison of the performance, measured by top-*k* accuracy, for the different ranking approaches combining MS² and RT information. The results shown are averaged accuracies over 350 sample MS feature sequences (LC-MS² experiments). **b**, Average top-*k* accuracies per MB-subset, rounded to full integers. The colour encodes the performance improvement in percentage units (%p) of each score integration method compared with only-MS².

conditions as well as molecular structure annotations. For training the SSVM model of LC-MS²Struct, the MassBank data were processed such that the experimental conditions were consistent within each MS feature set, that is, with identical LC set-up and MS configuration as in a typical LC-MS² experiment, to ensure comparable RT, RO and MS² data. We developed a Python package 'massbank2db' that can process MassBank records and group them into consistent MS feature sets, which we denote as MassBank subsets (MB-subsets). For our experiments, we sampled sequences of MS features from the MB-subsets to simulate real LC-MS² experiments where the signals of multiple unknown compounds are measured under consistent experimental set-ups. Figure 1e illustrates the grouping and LC-MS² sampling process. Two collections of MassBank data were considered: ALLDATA and the ONLYSTEREO subset.

https://doi.org/10.1038/s42256-022-00577-2

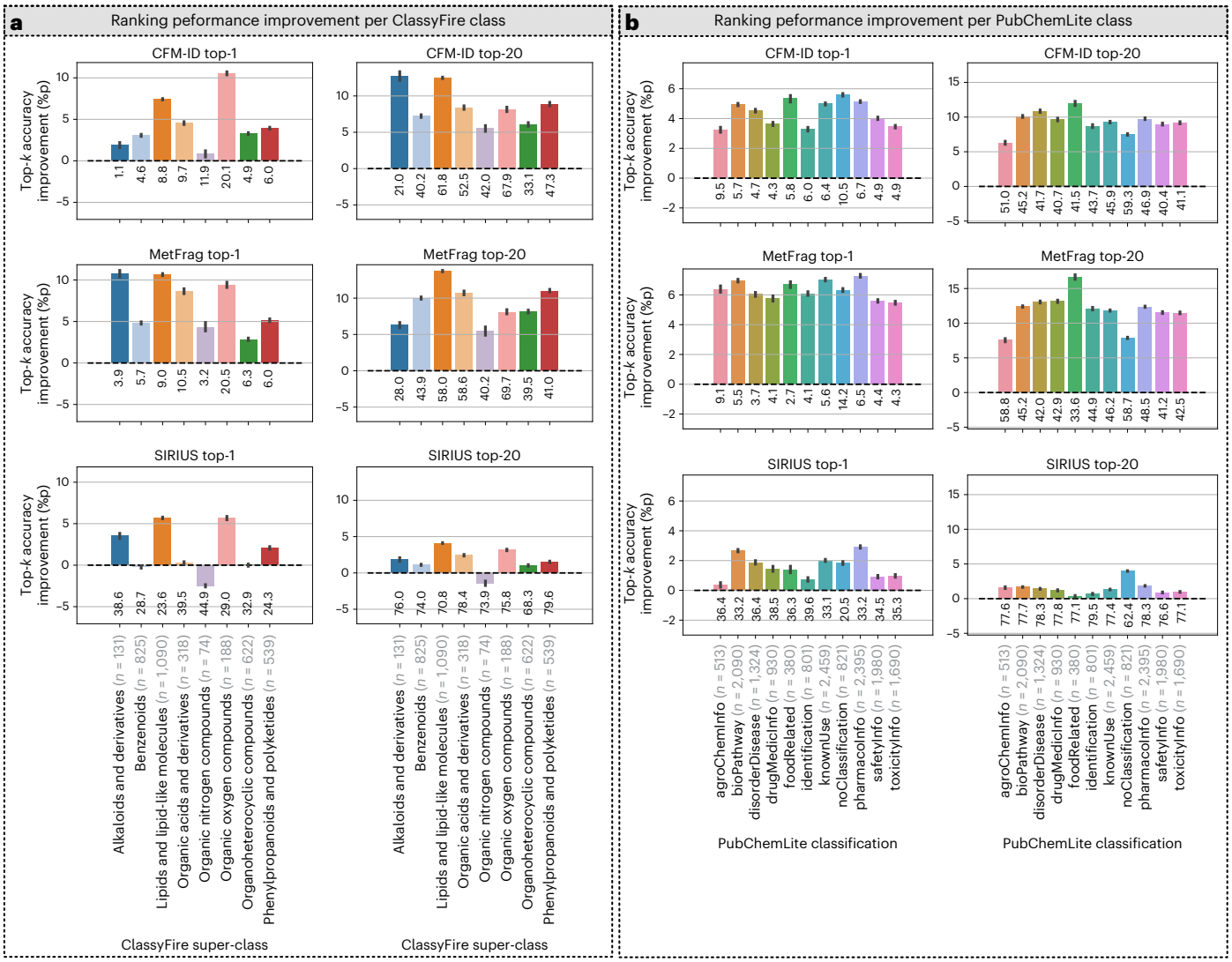


Fig. 3 | Performance gain by LC-MS²Struct across molecular classes. The ranking performance (top-*k*) improvement of LC-MS²Struct compared with only-MS² (baseline). The data are presented as mean values (50 samples) and the error bars show the 95% confidence interval of the mean estimate (1,000 bootstrapping samples). The top-*k* accuracies (%) under the bars show the only-MS² performance. For each molecular class, the number of unique molecular structures in the class is denoted in the *x*-axis label (*n*). **a**, Molecular classification using the ClassyFire[51] framework (class level). **b**, PubChemLite[40] annotation classification system. Molecules not present in PubChemLite are summarized under the 'noClassification' category. Note that in PubChemLite, a molecule can belong to multiple categories.

Comparison of LC-MS²Struct with other approaches

In the first experiment, we compare LC-MS²Struct with previous approaches for candidate ranking either using only-MS² or additionally using RT or RO information. Only-MS² uses the MS² spectrum information to rank the molecular candidates and serves as baseline; MS² + RO (ref. [34]) uses a ranking support vector machine (RankSVM)[48,49] to predict the ROs of candidate pairs and a probabilistic inference model to combine the ROs with MS² scores; MS² + RT uses predicted RTs to remove false-positive molecule structures from the candidate set, ordered by their MS² score, by comparing the predicted and observed RT; MS² + log*P* is an approach introduced by ref. [11], which uses the observed RT to predict the XlogP3 value[50] of the unknown compound and compares it with the candidates' XlogP3 values extracted from PubChem to refine the initial ranking based on the MS² scores. The RO-based methods (LC-MS²Struct and MS² + RO) were trained using the RTs from all available MB-subsets, ensuring that no test molecular structure (based on InChIKey first block, that is, the structural skeleton) was used for the model training (structure disjoint). For the RT-based approaches (MS² + RT and MS² + log*P*), the respective predictors were trained in a structure disjoint fashion using only the RT data available for that MB-subset. For the experiment, all MB-subsets with more than 75 (MS², RT)-tuples from the ALLDATA data set-up were used (Supplementary Table 2), as the RT-based approaches require LC-system-specific RT training data. The ranking performance was computed for each LC-MS² experiment within a particular MB-subset. The candidate molecules are identified by their InChIKey first block (the structural skeleton); hence, no stereoisomers are in the candidate sets.

Each candidate ranking approach was evaluated with three MS² scorers: CFM-ID 4.0[18], MetFrag[11] and SIRIUS[17]. For LC-MS²Struct, we use stereochemistry-aware molecular fingerprints (3D) to represent the candidates.

Figure 2a shows the average ranking performance (top-*k* accuracy) across 350 LC-MS² experiments, each encompassing about 50 (MS², RT)-tuples (Methods). LC-MS²Struct is the best-performing method combined with any of the three MS² scorers. For CFM-ID and MetFrag, LC-MS²Struct provides 4.7 and 7.3 percentage unit increases over the only-MS² for the top-1 accuracy, corresponding to 80.8% and 106% performance gain, respectively. In our setting, that translates to

https://doi.org/10.1038/s42256-022-00577-2

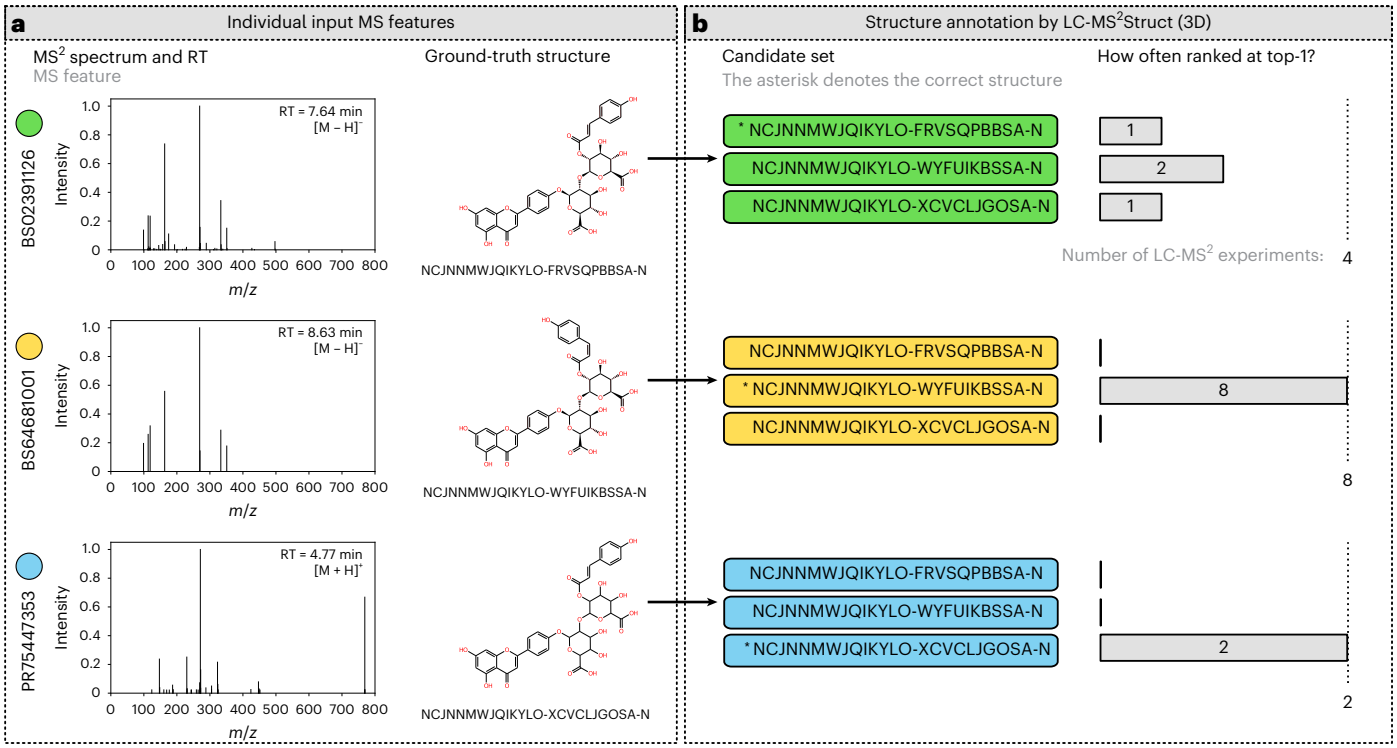


Fig. 4 | Application of LC-MS2Struct to annotate stereoisomers. Post-hoc analysis of the stereoisomer annotation using LC-MS^{2}Struct for three (MS^2, RT)-tuples from our MassBank data associated with the same 2D skeleton (InChIKey first block). In our evaluation, all three MS features were analysed multiple times in different contexts (BS02391126 in four, BS64681001 in eight and PR75447353 in two LC-MS^2 experiments). **a**, MS features with their ground-truth annotations. Two of the spectra (starting with BS) were measured under the same LC condition (MB-subset 'BS_000'), demonstrating the separation of *E*/*Z* isomers on LC columns. **b**, The candidate sets of the three features are identical (defined by the molecular formula $C_{36}H_{32}O_{19}$) and contain only three structures. For 12 out of the 14 LC-MS^2 experiments, LC-MS^{2}Struct predicts the correct *E*/*Z* isomer.

2.4 and 3.7 additional annotations at the top rank, respectively (out of approximately 50). The performance improvement increases for larger *k*, reaching as far as 9.3 and 11.3 percentage units for the top-20, which means 4.7 and 5.7 additional correct structures, respectively, in the top-20. For SIRIUS, the improvements are more modest, on average around 2 percentage units for top-1 to top-20. This might be explained by the higher baseline performance of SIRIUS. Nevertheless, SIRIUS can be improved for particular MB-subsets (see Fig. 2b and the discussion in the next section).

The runner-up score integration method is MS^2 + RO, which also makes use of predicted ROs. For CFM-ID and MetFrag, it leads to about one-third to one-half of the performance gain of LC-MS^{2}Struct. The approaches relying on RTs, either by candidate filtering (MS^2 + RT) or through log*P* prediction (MS^2 + log*P*), lead to only minor improvements for MetFrag and CFM-ID, but none for SIRIUS, for which MS^2 + RT even leads to a decrease in ranking performance by about 2 percentage units. An explanation for this is that the filtering approach removes on average 4.7% of the correct candidates, which leads to false-negative predictions.

The performance gain by using either RO or RT varies between the MB-subsets with differing LC-MS^2 set-ups (Supplementary Table 3) and compound class compositions (Extended Data Fig. 1). We illustrate these differences in Fig. 2b. Applying LC-MS^{2}Struct improves the ranking performance in almost all MB-subsets, including the SIRIUS MS^2-scorer (some very slight decreases were observed in some SIRIUS scored sets). This is in stark contrast to the RT-based approaches (MS^2 + RT and MS^2 + log*P*), which often lead to less accurate rankings, especially for SIRIUS. Furthermore, as seen already in the average results (Fig. 2a), the benefit of LC-MS^{2}Struct depends on the MS^2 base scorer. For example, the top-1 accuracy of the subsets 'AC_003' and 'NA_003' can be greatly improved for MetFrag but show little improvement for CFM-ID. Both datasets are natural-product toxins, which are perhaps poorly explained by the bond-disconnection approach of MetFrag. In contrast, for 'RP_001' and 'UF_003', the largest improvements (top-1) can be reached for CFM-ID. The RT-filtering approach (MS^2 + RT) performs particularly well for 'LQB_000' and 'UT_000'. These subsets mostly contain lipids and lipid-like molecules (Extended Data Fig. 1).

Since the RT prediction models are trained using only data from the respective MB-subsets, more accurate models may be reached for less heterogeneous subsets of molecules. Hence, the RT filtering could work well in such cases[26].

Performance for different compound classifications

Next we investigate how LC-MS^{2}Struct can improve the annotation across different categories in two molecule classification systems, ClassyFire[51] and PubChemLite[40]. Figure 3 shows the average top-1 and top-20 accuracy improvement of LC-MS^{2}Struct over the only-MS^2 baseline for each ClassyFire super-class and PubChemLite annotation category. For ClassyFire (Fig. 3a), the ranking performance improvement for the different super-classes depends on the MS^2 scorer. For example, the top-1 accuracy of 'Alkaloids and derivatives' can be improved by 10.8 percentage units for MetFrag, but improves much less for CFM-ID and SIRIUS (1.9 and 3.5 percentage units, respectively). For 'Organic oxygen compounds', in contrast, the top-1 accuracy improves by about 10 percentage units when using both CFM-ID and MetFrag, whereas only half that improvement is observed for SIRIUS. This suggests that the CFM-ID results may be improved with the inclusion of more 'Alkaloids and derivatives'. In addition, the 'Alkaloids and derivatives', 'Organic acids and derivatives' and 'Organic nitrogen compounds' appear less well explained by MetFrag (perhaps with more rearrangements, or

https://doi.org/10.1038/s42256-022-00577-2

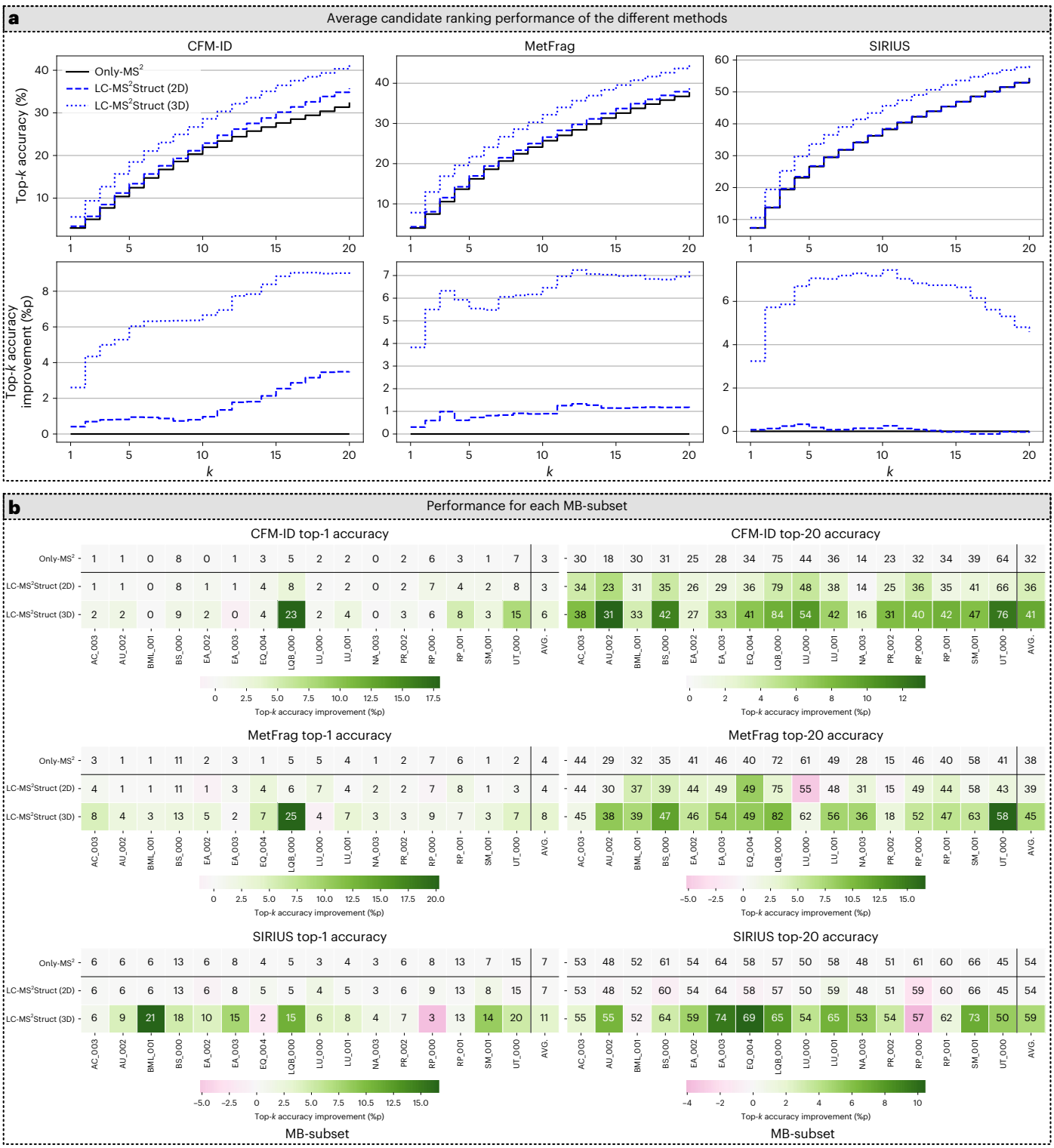


Fig. 5 | Using LC-MS²Struct to identify stereoisomers. a, Comparison of the performance, measured by top-*k* accuracy, of LC-MS²Struct using either 2D (no stereochemistry) or 3D (with stereochemistry) molecular fingerprints in the ONLYSTEREO setting. The results shown are averaged accuracies over 94 sample MS feature sequences (LC-MS² experiments). **b**, Average top-*k* accuracies per MB-subset rounded to full integers. The colour encodes the performance improvement in percentage units (%p) of each score integration method compared with only-MS².

less distinguishable spectra), such that the improvement from the RO approach is more apparent. For SIRIUS, 'Lipids and lipid-like molecules' as well as 'Organic oxygen compounds' benefit the most from LC-MS²Struct in top-1 (both improving by 5.7 percentage units) and top-20 (4.1 and 3.2 percentage units, respectively). In general, for 'Lipid and lipid-like molecules', LC-MS²Struct seems to achieve the best improvement (top-1 and top-20) over all MS² scorers. However, depending on the MS² scorer, this improvement distributes differently across the lipid sub-classes (Extended Data Fig. 2), such as 'Fatty acyls', 'Prenol lipids' or 'Sphingolipids'.

For the PubChemLite classification (Fig. 3b), we also see that the MS^2 scorers benefit differently from LC-MS^{2}Struct. The improvement is generally close to the average improvement of the respective MS^2 scorers and seems more equally distributed across the annotation categories.

For example for CFM-ID, the biggest top-1 improvements are in the 'foodRelated' and 'noClassification' categories. On the other hand, for SIRIUS the 'pharmacoInfo' and 'bioPathway' categories improve the most. MetFrag shows the most consistent performance improvement across the categories. 'agroChemInfo' benefits the least from LC-MS^{2}Struct (top-1 and top-20). A possible explanation could be that the molecules categorized as agrochemicals are mainly 'Benzenoids' (48.5%), 'Organoheterocyclic compounds' (25.9%) and 'Organic acids and derivatives' (11.6%). As shown in Fig. 3a, these three ClassyFire classes show low (CFM-ID and MetFrag) or practically no (SIRIUS) improvement when using ROs.

Annotation of stereoisomers

Finally, we study whether LC-MS^{2}Struct can annotate stereoisomers more accurately than MS^2 alone, considering differences between stereoisomers that vary in their double-bond orientation (for example, *cis*–*trans* or *E*–*Z* isomerism), which may potentially lead to differences in their LC behaviour (Fig. 4a). We consider candidate sets containing stereoisomers and evaluate LC-MS^{2}Struct only using MassBank records where the ground-truth structure has stereochemistry information provided, that is, where the InChIKey second block is not 'UHFFFAOYSA' (ONLYSTEREO data set-up; Methods). The molecular candidates are represented using two different molecular fingerprints: one that includes stereochemistry information (3D); and one that omits it (2D) (Methods). This allows us to assess the importance of stereochemistry-aware features for the structure annotation.

Figure 5a shows the ranking performance of LC-MS^{2}Struct using 2D and 3D fingerprints.

When looking into the top-1 performance of LC-MS^{2}Struct (3D) for the individual MS^2 scorers, we observe an improvement by 2.6, 3.8 and 3.2 percentage units for CFM-ID, MetFrag and SIRIUS, respectively. This translates to performance gains of 87.3%, 95.9% and 44.3%, respectively.

In general, LC-MS^{2}Struct improves the ranking for all three MS^2 scorers. The improvement, however, is notably larger when using stereochemistry-aware (3D) candidate features. Interestingly, a similar behaviour could be observed in the ALLDATA setting (Extended Data Fig. 3), even though the absolute performance improvements were smaller. This experiment demonstrates that LC-MS^{2}Struct can use RO information to improve the annotation of stereoisomers.

Discussion

LC-MS^{2}Struct is a novel approach for the integration of MS^2 and LC data for the structural annotation of small molecules. The method learns from the pairwise dependencies in the RO of MS features within similar LC configurations and can generalize across different, heterogeneous LC configurations. Furthermore, the use of stereochemistry-aware molecular fingerprints enables LC-MS^{2}Struct to annotate stereoisomers in LC-MS^2 experiments based on the observed ROs. Also, our novel processing pipeline to group all (MS^2, RT) data from MassBank into subsets of homogeneous LC-MS^2 conditions, which is implemented and made available in the 'massbank2db'[52] Python package will, we believe, make MassBank more accessible to other researchers and hence lower the bar of entry to computational metabolomics research.

Our experiments demonstrate that LC-MS^{2}Struct annotates small molecules with an accuracy far superior to more traditional RT filtering and log*P*-based approaches, and also markedly better than previous methods that rely on ROs. In particular, compared with ref. [34], which used a graphical model as a post-hoc integration tool of MS^2 scores and RO predictions, the benefits of learning the parameters of the graphical model are clear. All three studied MS^2 scorers could be improved by LC-MS^{2}Struct, including the best-of-class SIRIUS, for which improvements have generally been hard to come by due to its already high baseline accuracy. Our results show the superiority of stereochemistry-aware molecular features for the structure annotation of LC-MS^2 data. Remarkably, this was the case not only for the annotation of stereoisomers but also for candidates distinguished by only their 2D structure. This result could be relevant for improving structural annotations in ion mobility separation–mass spectrometry with collision-cross-section measurements.

Our examples indicated that LC-MS^{2}Struct separates candidates with varying double-bond stereochemistry, that is, *E*/*Z* and *cis*/*trans* isomers (see, for example, Fig. 4). However, there were very few examples of double-bond and/or chiral isomers measured on the same LC system in our dataset, which makes it difficult to quantify this effect, or interrogate these further until more such data are publicly available. Furthermore, as non-chiral LC cannot distinguish stereoisomers that differ only in their chiral centres, the development of more selective stereochemistry-aware molecular features, ignoring the chiral annotations, might be beneficial. We also note that the direct modelling of a node score (MS^2 information) predictor in the SSVM would be possible. However, as the MS^2 scorers used here are already relatively mature and well known in the community, we have left this research line open for future efforts.

Methods

Notation

We use the following notation to describe LC-MS^{2}Struct:

Sequence of spectra	$\mathbf{x} = (x_1, \ldots, x_L)$	with $x_\sigma \in \mathcal{X}$
Sequence of retention times	$\mathbf{t} = (t_1, \ldots, t_L)$	with $t_\sigma \in \mathbb{R}_{\geq 0}$
Sequence of candidate sets	$\mathcal{C} = (\mathcal{C}_1, \ldots, \mathcal{C}_L)$	with $\mathcal{C}_\sigma \subseteq \mathcal{Y}$
Sequence of labels	$\mathbf{y} = (y_1, \ldots, y_L) \in \Sigma$	with $y_\sigma \in \mathcal{Y}$
Candidate assignment space	$\Sigma = \mathcal{C}_1 \times \ldots \times \mathcal{C}_L$,	

where $\mathcal{X}$ and $\mathcal{Y}$ denote the MS^2 spectra and the molecular structure space, respectively, $\mathcal{C}$ denotes a candidate set that is a subset of all possible molecular structures, and $A \times B$ denotes the cross product of two sets A and B. For the purpose of model training and evaluation, we assume a dataset with ground-truth-labelled MS feature sequences: $\mathcal{D} = \{((\mathbf{x}_i, \mathbf{t}_i), \mathcal{C}_i, \mathbf{y}_i)\}_{i=1}^N$, where N denotes the total number of sequences. We use $i, j \in \mathbb{N}_{\geq 0}$ to index MS feature sequences and $\sigma, \tau \in \mathbb{N}_{\geq 0}$ as indices for individual MS features within a sequence, for example, $x_{i\sigma}$ denotes the MS^2 spectrum at index σ in the sequence i. The length of a sequence of MS features is denoted with L. We denote the ground-truth labels (candidate assignment) of sequence i with $\mathbf{y}_i$ and any labelling with $\mathbf{y}$. Both, $\mathbf{y}_i$ and $\mathbf{y}$ are in Σ_i. We use y to denote the candidate label variable, whereas m denotes a particular molecular structure. For example, $y_\sigma = m$ means that we assign the molecular structure m as label to the MS feature σ.

Graphical model for joint annotation of MS features

We consider the molecular annotation problem for the output of LC-MS^2, which means assigning a molecular structure to each MS feature, as a structured prediction problem[35,46,47], relying on a graphical model representation of the sets of MS features arising from an LC-MS^2 experiment. For each MS feature σ, we want to predict a label y_σ from a fixed and finite candidate (label) set $\mathcal{C}_\sigma$. We model the observed ROs between each MS feature pair (σ, τ) within an LC-MS^2 experiment, as pairwise dependencies of the features. We define an undirected graph $G = (V, E)$ with the vertex set V containing a node σ for each MS feature and the edge set E containing an edge for each MS feature pair $E = \{(\sigma, \tau) \mid \sigma, \tau \in V, \sigma \neq \tau\}$ (compare Fig. 1a,c). The resulting graph is complete with an edge between all pairs of nodes. This allows us to make use of arbitrary pairwise dependencies, instead of limiting to,

say, adjacent RTs. This modelling choice was previously shown to be beneficial by ref. [34]. Here we extend that approach by learning from the pairwise dependencies to optimize joint annotation accuracy, which leads to markedly improved annotation accuracy.

For learning, we define a scoring function F that, given the input MS feature sequences $(\mathbf{x}, \mathbf{t})$ and its corresponding sequence of candidate sets $\mathcal{C}$, computes a compatibility score between the measured data and any possible sequence of labels $\mathbf{y} \in \Sigma$:

$$F(\mathbf{y} \mid \mathbf{x}, \mathbf{t}, \mathbf{w}, G) = \frac{1}{|V|} \sum_{\sigma \in V} \theta(x_\sigma, y_\sigma) + \frac{1}{|E|} \sum_{(\sigma,\tau) \in E} \langle \mathbf{w}, \mathbf{\Gamma}(\mathbf{t}^{\sigma\tau}, \mathbf{y}^{\sigma\tau}) \rangle, \tag{2}$$

where $\theta : \mathcal{X} \times \mathcal{Y} \to (0, 1]$ is a function returning an MS^2 matching score between the spectrum x_σ and a candidate $y_\sigma \in \mathcal{C}_\sigma$, $\langle \cdot, \cdot \rangle$ denotes the inner product, and $\mathbf{w}$ is a model weight vector to predict the RO matching score, based on the joint-feature vector $\mathbf{\Gamma} : \mathbb{R}_{\geq 0} \times \mathbb{R}_{\geq 0} \times \mathcal{Y} \times \mathcal{Y} \to \mathcal{F}$ between the observed RO derived from $\mathbf{t}^{\sigma\tau} = (t_\sigma, t_\tau)$ and a pair of molecular candidates $\mathbf{y}^{\sigma\tau} = (y_\sigma, y_\tau)$.

Equation (2) consists of two parts: (1) a score computed over the nodes in G capturing the MS^2 information; and (2) a score expressing the agreement of observed and predicted RO computed over the edge set. We assume that the node scores are pre-computed by a MS^2 scorer such as CFM-ID[18], MetFrag[11] or SIRIUS[17]. The node scores are normalized to (0, 1] within each candidate set $\mathcal{C}_\sigma$. The edge scores are predicted for each edge (σ, τ) using the model $\mathbf{w}$ and the joint-feature vector $\mathbf{\Gamma}$:

$$\begin{aligned} f(\mathbf{t}^{\sigma\tau}, \mathbf{y}^{\sigma\tau} \mid \mathbf{w}) &= \langle \mathbf{w}, \mathbf{\Gamma}(\mathbf{t}^{\sigma\tau}, \mathbf{y}^{\sigma\tau}) \rangle \\ &= \langle \mathbf{w}, \operatorname{sign}(t_\sigma - t_\tau)\,(\phi(y_\sigma) - \phi(y_\tau)) \rangle \\ &= \operatorname{sign}(t_\sigma - t_\tau) \langle \mathbf{w}, \phi(y_\sigma) - \phi(y_\tau) \rangle, \end{aligned} \tag{3}$$

with $\phi : \mathcal{Y} \to \mathcal{F}_\mathcal{Y}$ being a function embedding a molecular structure into a feature space. The edge prediction function (3) will produce a height edge score, if the observed RO (that is, $\operatorname{sign}(t_\sigma - t_\tau)$) agrees with the predicted one.

Using the compatibility score function (2), the predicted joint annotation for $(\mathbf{x}, \mathbf{t})$ corresponds to the the highest-scoring label sequence $\hat{\mathbf{y}} \in \Sigma : \hat{\mathbf{y}} = \arg\max_{\bar{\mathbf{y}} \in \Sigma} F(\bar{\mathbf{y}} \mid \mathbf{x}, \mathbf{t}, \mathbf{w}, G)$. In practice, however, instead of predicting only the best label sequence, it can be useful to rank the molecular candidates $m \in \mathcal{C}_\sigma$ for each MS feature σ. This is because for state-of-the-art MS^2 scorers, the annotation accuracy in the top-20 candidate list is typically much higher than for the highest-ranked candidate (top-1).

Our framework provides candidate rankings by solving the following problem for each MS feature σ and $m \in \mathcal{C}_\sigma$:

$$\mu(y_\sigma = m \mid \mathbf{x}, \mathbf{t}, \mathbf{w}, G) = \max_{\{\bar{\mathbf{y}} \in \Sigma \,:\, \bar{y}_\sigma = m\}} F(\bar{\mathbf{y}} \mid \mathbf{x}, \mathbf{t}, \mathbf{w}, G). \tag{4}$$

Problem (4) returns a max-marginal μ score for each candidate m. That is, the maximum compatibility score any label sequence $\bar{\mathbf{y}} \in \Sigma$ with $\bar{y}_\sigma = m$ can achieve. One can interpret equation (2) as the log-space representation of a unnormalized Markov random field probability distribution over $\mathbf{y}$ associated with an undirected graphical model G (ref. [44]).

Feasible inference using random spanning trees

For general graphs G, the maximum a posterior inference problem (that is, finding the highest-scoring label sequence $\mathbf{y}$ given an MS feature sequence) is an $\mathcal{NP}$-hard problem[53,54]. The max-marginals inference (MMAP), needed for the candidate ranking, is an even harder problem which is $\mathcal{NP}^{\mathcal{PP}}$ complete[54]. However, efficient inference approaches have been developed. In particular, if G is tree-like, we can efficiently compute the max-marginals using dynamic programming and the max-product algorithm[43,44]. Such tree-based approximations have shown to be successful in various practical applications[34,45,46].

Here, we follow the work by ref. [34] and sample a set of random spanning trees (RST) $\mathbf{T} = \{T_k\}_{k=1}^K$ from G, whereby K denotes the size of the RST sample. Each tree T_k has the same node set V as G, but an edge set $E(T) \subseteq E$, with $|E(T)| = L - 1$, ensuring that T is a single connected component and cycle free. We follow the sampling procedure used by ref. [34]. Given the RST set $\mathbf{T}$, we compute the averaged max-marginals to rank the molecular candidates[34]:

$$\bar{\mu}(y_\sigma = m \mid \mathbf{x}, \mathbf{t}, \mathbf{w}, \mathbf{T}) = \frac{1}{K} \sum_{k=1}^{K} \left(\mu(y_\sigma = m \mid \mathbf{x}, \mathbf{t}, \mathbf{w}, T_k) - \max_{\bar{\mathbf{y}} \in \Sigma} F(\bar{\mathbf{y}} \mid \mathbf{x}, \mathbf{t}, \mathbf{w}, T_k) \right), \tag{5}$$

where we subtract the maximum compatibility score from the marginal values corresponding to the individual trees to normalize the marginals before averaging[34]. This normalization value can be efficiently computed given the max-marginals μ. In our experiments, we train K individual models $(\mathbf{w}_k)$ and associate them with the trees T_k to increase the diversity. The influence of the number of SSVM models on the prediction performance is shown in Extended Data Fig. 4.

The SSVM model

To train the model parameters $\mathbf{w}$ (equation (2)), we implemented a variant of the SSVM[35,36]. Its primal optimization problem is given as[55]:

$$\begin{aligned} \min_{\mathbf{w}, \boldsymbol{\xi}} \quad & \frac{1}{2} \| \mathbf{w} \|^2 + \frac{C}{N} \sum_{i=1}^{N} \xi_i \\ \text{st.} \quad & F(\mathbf{y}_i \mid \mathbf{x}_i, \mathbf{t}_i, \mathbf{w}, G_i) - F(\mathbf{y} \mid \mathbf{x}_i, \mathbf{t}_i, \mathbf{w}, G_i) \geq \ell(\mathbf{y}_i, \mathbf{y}) - \xi_i \\ & \forall i \in \{1, \ldots, N\},\ \forall \mathbf{y} \in \Sigma_i, \end{aligned} \tag{6}$$

where $C > 0$ is the regularization parameter, $\xi_i \geq 0$ is the slack variable for example i and $\ell : \Sigma_i \times \Sigma_i \to \mathbb{R}_{\geq 0}$ is a function capturing the loss between two label sequences. The constraint set definition (st.) of problem (6) leads to a parameter vector $\mathbf{w}$ that is trained according to the max-margin principle[35,36,47], that is, the score $F(\mathbf{y}_i)$ of the correct label should be greater than the score $F(\mathbf{y})$ of any other label sequence by at least the specified margin $\ell(\mathbf{y}_i, \mathbf{y})$. Note that in the SSVM problem (6), a different graph $G_i = (V_i, E_i)$ can be associated with each training example i, allowing, for example, to process sequences of different length.

We solve (6) in its dual formulation and use the Frank–Wolfe algorithm[56] following the recent work by ref. [55]. In the Supplementary Information, we derive the dual problem and demonstrate how to solve it efficiently using the Frank–Wolfe algorithm and RST approximations for G_i. Optimizing the dual problem enables us to use non-linear kernel functions $\lambda : \mathcal{Y} \times \mathcal{Y} \to \mathbb{R}_{\geq 0}$ measuring the similarity between the molecular structures associated with the label sequences.

The label loss function ℓ is defined as follows:

$$\ell(\mathbf{y}_i, \mathbf{y}) = \frac{1}{|V_i|} \sum_{\sigma=1}^{L} (1 - \lambda(y_{i\sigma}, y_\sigma))$$

and satisfies $\ell(\mathbf{y}, \mathbf{y}) = 0$ (a required property[55]), if λ is a normalized kernel, which holds true in our experiments (we used the MinMax kernel[57]).

Pre-processing pipeline for raw MassBank records

Extended Data Fig. 5 illustrates our MassBank pre-processing pipeline implemented in the Python package 'massbank2db'[52]. First, the MassBank records text files were parsed and the MS^2 spectrum, ground-truth annotation, RT and meta-information extracted. Records with missing MS^2, RT or annotation were discarded. We use the MB 2020.11 release for our experiments.

Subsequently, we grouped the MassBank records into subsets (denoted as MB-subsets) where the (MS^2, RT)-tuples were measured under the same LC and MS conditions.

Supplementary Table 1 summarizes the grouping criteria. In the next step, we used the InChIKey[58] identifier in MassBank to retrieve the SMILES[59] representation from PubChem[20] (1 February 2021), rather than using the contributor-supplied SMILES. This ensures a consistent SMILES source for the molecular candidates and ground-truth annotations.

Three more filtering steps were performed before creating the final database, to remove records: (1) if the ground-truth exact mass deviated too far (>20 ppm) from the calculated exact mass based on the precursor mass-per-charge and adduct type; (2) if the subset contained <50 unique molecular structures; (3) if they were potential isobars (see pull-request #152 in the MassBank GitHub repository, https://github.com/MassBank/MassBank-data/pull/152).

Supplementary Table 3 summarizes the LC-MS^2 meta-information for all generated MB-subsets.

Generating the molecular candidate sets

We used SIRIUS[8,17] to generate the molecular candidate sets. For each MassBank record, the ground-truth molecular formula was used by SIRIUS to collect the candidate structures from PubChem[20]. The candidate sets generated by SIRIUS contain a single stereoisomer per candidate, identified by their InChIKey first block (structural skeleton). To study the ability of LC-MS^{2}Struct to annotate the stereochemical variant of the molecules, we enriched the SIRIUS candidates sets with stereoisomers, using the InChIKey first block of each candidate to search PubChem (1 February 2021) for stereoisomers. The additional molecules were then added to the candidate sets.

Pre-computing the MS^2 matching scores

For each MB-subset, MS^2 spectra with identical adduct type (for example, $[M + H]^+$) and ground-truth molecular structure were aggregated. Depending on the MS^2 scorer, we either merged the MS^2 into a single spectrum (CFM-ID and MetFrag) following the strategy by ref. [11] or we provided the MS^2 spectra separately (SIRIUS). For the spectra merging, we used the 'mzClust_hclust' function of the xcms package[60], which first combines all MS^2 spectra's peaks into a single peaklist and subsequently merges peaks based on a mass error threshold.

To compute the CFM-ID (v4.0.7) MS^2 matching score, we first predicted the in silico MS^2 spectra for all molecular candidate structures based on their isomeric SMILES representation using the pre-trained CFM-ID models (Metlin 2019 MSML) by ref. [18]. We merged the three in silico spectra predicted by CFM-ID for different collision energies and compared them with the merged MassBank spectrum using the modified cosine similarity[61] implemented in the matchms[62] (v0.9.2) Python library. For MetFrag (v2.4.5), the MS^2 matching scores were calculated using the FragmenterScore feature based on the isomeric SMILES representation of the candidates. For SIRIUS, the required fragmentation trees are computed using the ground-truth molecular formula of each MassBank spectrum. SIRIUS uses canonical SMILES and hence does not encode stereochemical information (which is absent in the canonical SMILES). Therefore, we used the same SIRIUS MS^2 matching score for all stereoisomers sharing the same InChIKey first block.

For all three MS^2 scorers, we normalized the MS^2 matching scores to the range [0, 1] separately for each candidate set. For the machine-learning-based scorers (CFM-ID and SIRIUS), the matching scores of the candidates associated with a particular MassBank record using in evaluation were predicted using models that did include its ground-truth structure (determined by InChIKey first block).

If a MS^2 scorer failed on a MassBank record, we assigned a constant MS^2 score to each candidate.

Molecular feature representations

Extended connectivity fingerprints with function classes (FCFP)[38] were used to represent molecular structures in our experiments. We employed RDKit (v2021.03.1) to generate counting FCFP fingerprints. The fingerprints were computed based on the isomeric SMILES, using the parameter 'useChirality' to generate fingerprints that either encoded stereochemistry (3D) or not (2D). To define the set of substructures in the fingerprint vector, we first generated all possible substructures, using a FCFP radius of two, based on a set of 50,000 randomly sampled molecular candidates associated with our training data, and all the ground-truth training structures, resulting in 6,925 (3D) and 6,236 (2D) substructures. We used 3D FCFP fingerprints in our experiments, except for the experiments focusing on the annotation of stereoisomers, where we used both 2D and 3D fingerprints for comparison. We used the MinMax kernel[57] to compute the similarity between the molecules.

Computing molecular categories

For the analysis of the ranking performance for different molecular categories, we used two classification systems, ClassyFire[51], which classifies molecules according to their structure, and PubChemLite[40], which classifies molecules according to information available for ten exposomics-relevant categories. For ClassyFire, we used the 'classyfireR' R package to retrieve the classification for each ground-truth molecular structure in our dataset. For PubChemLite, the classification categories were retrieved via InChIKey first block matching of each molecular structure; if it was not found in PubChemLite, the category 'noClassification' was assigned.

Training and evaluation data set-ups

We considered only MassBank data that have been analysed using an LC reversed-phase (RP) column. We removed molecules from the data if their measured RT was less than three times the estimated column dead-time[63], as we considered such molecules to be non-retaining.

We considered two separate data set-ups. The first one, denoted by ALLDATA, used all available MassBank data to train and evaluate LC-MS^{2}Struct. This set-up was used to compare the different candidate ranking approaches as well as to investigate the performance across various molecular classes. The second set-up, denoted by ONLYSTEREO, used MassBank records where the ground-truth molecular structure contains stereochemical information, that is, where the InChIKey second block is not 'UHFFFAOYSA'. This set-up was used in the experiments regarding the ability of LC-MS^{2}Struct to distinguish stereochemistry. In the training, we additionally used MassBank records that appear only without stereochemical information in our candidate sets, identified by the InChIKey second block equal to 'UHFFFAOYSA' in PubChem. The number of available training and evaluation (MS^2, RT)-tuples per MB-subset are summarized in Supplementary Table 2.

For each MB-subset, we sampled a set of LC-MS^2 experiments, that is (MS^2, RT)-tuple sequences, from the available evaluation data. The number of LC-MS^2 experiments (n below) depended on the number of available (MS^2, RT)-tuples (Supplementary Table 2) as follows:

$$n = \begin{cases} 0 & \text{if } |\mathcal{D}| < 30 \\ 1 & \text{else if } 30 \leq |\mathcal{D}| \leq 75 \\ 15 & \text{else if } 76 \leq |\mathcal{D}| \leq 250 \\ \left\lfloor \frac{|\mathcal{D}|}{50} \right\rfloor & \text{else.} \end{cases}$$

where $\mathcal{D}$ is a set of (MS^2, RT)-tuples with ground-truth annotation and molecular candidate sets associated with an MB-subset. If there are fewer than 30 (MS^2, RT)-tuples available, we do not generate an evaluation LC-MS^2 experiment from the corresponding MB-subset. On the basis of this sampling scheme, we obtained 354 and 94 LC-MS^2 experiments for ALLDATA and ONLYSTEREO, respectively, for our evaluation (Supplementary Table 2).

We trained eight ($K = 8$) separate SSVM models $\mathbf{w}_k$ for each evaluation LC-MS^2 experiment. For each SSVM, model we first generated a

set containing the (MS^2, RT)-tuples from all MB-subsets. Then, we removed all tuples whose ground-truth molecular structure, determined by the InChIKey first block, was in the respective evaluation LC-MS^2 experiment. Lastly, we randomly sampled LC-MS^2 experiments from the training tuples, within their respective MB-subset, with a length randomly chosen from 4 to (maximum) 32 (see also Fig. 1e) and an RST T_{ik} assigned for each MS feature sequence i. In total, 768 LC-MS^2 training experiments were generated for each SSVM model. To speed up the model training, we restricted the candidate set size $|\mathcal{C}_{i\sigma}|$ of each training MS feature σ to maximum 75 candidate structures by random subsampling. We ensure that the correct candidate is included in the subsample. Each SSVM model $\mathbf{w}_k$ was applied to the evaluation LC-MS^2 experiment, associated with different RSTs T_k, and the averaged max-marginal scores where used for the final candidate ranking (see equation (5) and Fig. 1c).

SSVM hyperparameter optimization

The SSVM regularization parameter C was optimized for each training set separately using grid search and evaluation on a random validation set sampled from the training data's (MS^2, RT)-tuples (33%). A set of LC-MS^2 experiments was generated from the validation set and used to determine the normalized discounted cumulative gain (NDCG)[64] for each C value. The regularization parameter with the highest NDCG value was chosen to train the final model. We used the scikit-learn[65] (v0.24.1) Python package to compute the NDCG value, taking into account ranks up until 10 (NDCG@10) and defined the relevance for each candidate to be 1 if it is the correct one and 0 otherwise. To reduce the training time, we searched the optimal C^* only for SSVM model $k = 0$ and used C^* for the other models with $k > 0$.

Ranking performance evaluation

We computed the ranking performance (top-k accuracy) for a given LC-MS^2 experiment using the tie-breaking strategy described in ref. [8]: if a ranking method assigns an identical score to a set of n molecular candidates, then all accuracies at the ordinal ranks k at which one of these candidates is found are increased by $1/n$. We computed a candidate score (that is, only-MS^2, LC-MS^2Struct and so on) for each molecular structure in the candidate set (identified by PubChem CID). Depending on the data set-up (Supplementary Table 4), we first collapse the candidates by InChIKey first block (ALLDATA, method comparison and molecule category analysis) or full InChIKey (ONLYSTEREO stereochemistry prediction), assigning the maximum candidate score for each InChIKey first block or InChIKey group, respectively. Subsequently, we compute the top-k accuracy based on the collapsed candidate sets.

For the performance analysis of individual molecule categories, either ClassyFire[51] or PubChemLite[40] classes, we first computed the rank of the correct molecular structure for each (MS^2, RT)-tuple of each LC-MS^2 evaluation experiment based on only-MS^2 and LC-MS^2Struct scores. Subsequently, we computed the top-k accuracy for each molecule category, associated with at least 50 unique ground-truth molecular structures (based on InChIKey first block). As a ground-truth structure can appear multiple times in our dataset, we generate 50 random samples, each containing only one example per unique structure, and computed the averaged top-k accuracy.

Comparison of LC-MS^2Struct with other approaches

We compared LC-MS^2Struct with three different approaches to integrate MS^2 and RT information, namely RT filtering, log$P$ prediction and RO prediction.

For RT filtering (MS^2 + RT), we followed ref. [26], which used the relative error $\epsilon = \frac{|\hat{t}-t_\sigma|}{t_\sigma}$, between the predicted ($\hat{t}$) and observed (t_σ) RT. We set the filtering threshold to the 95% quantile of the relative RT prediction errors estimated from the RT model's training data, following refs. [27,29]. We used scikit-learn's[65] (v0.24.1) implementation of the support vector regression[66] with radial basis function kernel for the RT prediction. For support vector regression, we use the same 196 features, computed using RDKit (v2021.03.1), as in ref. [25].

For log$P$ prediction (MS^2 + log$P$), we followed ref. [11], which assigned a weighted sum of an MS^2 and log$P$ score $s = \beta s_{MS^2}(m) + (1-\beta)s_{\log P}(m)$ to each candidate $m \in \mathcal{C}_\sigma$, and used it rank the set of molecular candidates. The log$P$ score is given by $s_{\log P}(m) = \frac{1}{\delta\sqrt{2\pi}} \exp\left(-\frac{(\log P_m - \log P_\sigma)^2}{2\delta^2}\right)$ where $\log P_m$ is the predicted XlogP3[50] extracted from PubChem[20] for candidate m, and $\log P_\sigma = a\ t_\sigma + b$ is the XlogP3 value of the unknown compound, associated with MS feature σ, predicted based on its measured RT t_σ. The parameters a and b of the linear regression model were determined using a set of RT and XlogP3 tuples associated with the LC system. As in ref. [11], we set $\delta = 1.5$ and set β such that it optimizes the top-1 candidate ranking accuracy, calculated from a set of 25 randomly generated training LC-MS^2 experiments.

For RO prediction (MS^2 + RO), we used the approach by ref. [34], which relies on a RankSVM implementation in the Python library ROSVM[31,67] (v0.5.0). We used counting 'substructure' fingerprints calculated using CDK (v2.5)[68] and the MinMax kernel[57]. The MS^2 matching scores and predicted ROs were used to compute max-marginal ranking scores using the framework by ref. [34]. We used the author's implementation in version 0.2.3[69]. The hyper-parameters β and k of the model were optimized for each evaluation LC-MS^2 experiment separately using the respective training data. To estimate β, we generated 25 LC-MS^2 experiments from the training data and selected the β that maximized the Top20AUC[34] ranking performance. The sigmoid parameter k was estimated using Platt's method[70] calibrated using RankSVM's training data. We used 128 random spanning trees per evaluation LC-MS^2 experiment to compute the averaged max-marginals.

For the experiments comparing the different methods, we used all LC-MS^2 experiments generated, except the ones from the MB-subsets 'CE_001', 'ET_002', 'KW_000' and 'RP_000' (Supplementary Table 2). For those subsets, the evaluation LC-MS^2 experiment contains all available (MS^2, RT)-tuples, leaving no LC-system-specific data to train the RT (MS^2 + RT) or log$P$ (MS^2 + log$P$) prediction models. The RT and log$P$ prediction models are trained in a structure disjoint fashion using the RT data of the particular MB-subset associated with the evaluation LC-MS^2. The RO prediction model used by MS^2 + RO is trained structure disjoint as well, but using the RTs of all MB-subsets.

Reporting summary

Further information on research design is available in the Nature Portfolio Reporting Summary linked to this article.

Data availability

All data used in our experiments are available online[71] (https://zenodo.org/record/5854661). The candidate rankings of all LC-MS^2 experiments are available online: ALLDATA[72] (https://zenodo.org/record/6451016) and ONLYSTEREO[73] (https://zenodo.org/record/6037629). Source data are provided with this paper.

Code availability

The source code developed for this study is available on GitHub, including the implementation of LC-MS^2Struct[74] (v2.13.0; https://github.com/aalto-ics-kepaco/msms_rt_ssvm); scripts to run the experiments[75] (https://github.com/aalto-ics-kepaco/lcms2struct_exp); and the library implementing the MassBank pre-processing[52] (v0.9.0; https://github.com/bachi55/massbank2db). The candidate fingerprints were computed by the ROSVM Python library[67] (v0.5.0; https://github.com/bachi55/rosvm) using RDKit (2021.03.1). The SSVM library uses the max-marginal inference solver implemented by ref. [34] (v0.2.3; https://github.com/aalto-ics-kepaco/msms_rt_score_integration).

References

1. da Silva, R. R., Dorrestein, P. C. & Quinn, R. A. Illuminating the dark matter in metabolomics. *Proc. Natl Acad. Sci. USA* **112**, 12549–12550 (2015).
2. Aksenov, A. A., da Silva, R., Knight, R., Lopes, N. P. & Dorrestein, P. C. Global chemical analysis of biology by mass spectrometry. *Nat. Rev. Chem.* **1**, 0054 (2017).
3. Blaženović, I. et al. Structure annotation of all mass spectra in untargeted metabolomics. *Anal. Chem.* **91**, 2155–2162 (2019).
4. Blaženović, I., Kind, T., Ji, J. & Fiehn, O. Software tools and approaches for compound identification of LC-MS/MS data in metabolomics. *Metabolites* **8**, 31 (2018).
5. Schymanski, E. L. et al. Critical assessment of small molecule identification 2016: automated methods. *J. Cheminform.* **9**, 22 (2017).
6. Nguyen, D. H., Nguyen, C. H. & Mamitsuka, H. Recent advances and prospects of computational methods for metabolite identification: a review with emphasis on machine learning approaches. *Brief. Bioinform.* **20**, 2028–2043 (2019).
7. Wolf, S., Schmidt, S., Müller-Hannemann, M. & Neumann, S. In silico fragmentation for computer assisted identification of metabolite mass spectra. *BMC Bioinform.* **11**, 1–12 (2010).
8. Dührkop, K., Shen, H., Meusel, M., Rousu, J. & Böcker, S. Searching molecular structure databases with tandem mass spectra using CSI:FingerID. *Proc. Natl Acad. Sci. USA* **112**, 12580–12585 (2015).
9. Allen, F., Greiner, R. & Wishart, D. Competitive fragmentation modeling of ESI-MS/MS spectra for putative metabolite identification. *Metabolomics* **11**, 98–110 (2015).
10. Brouard, C. et al. Fast metabolite identification with input output kernel regression. *Bioinformatics* **32**, i28–i36 (2016).
11. Ruttkies, C., Schymanski, E. L., Wolf, S., Hollender, J. & Neumann, S. MetFrag relaunched: incorporating strategies beyond in silico fragmentation. *J. Cheminform.* **8**, 3 (2016).
12. Brouard, C., Bach, E., Böcker, S. & Rousu, J. Magnitude-preserving ranking for structured outputs. In *Proc. Ninth Asian Conference on Machine Learning, Proc. Machine Learning Research* Vol. 77 (eds Zhang, M.-L. & Noh, Y.-K.) 407–422 (PMLR, 2017); http://proceedings.mlr.press/v77/brouard17a.html
13. Nguyen, D. H., Nguyen, C. H. & Mamitsuka, H. Simple: sparse interaction model over peaks of molecules for fast, interpretable metabolite identification from tandem mass spectra. *Bioinformatics* **34**, i323–i332 (2018).
14. Li, Y., Kuhn, M., Gavin, A.-C. & Bork, P. Identification of metabolites from tandem mass spectra with a machine learning approach utilizing structural features. *Bioinformatics* **36**, 1213–1218 (2019).
15. Ruttkies, C., Neumann, S. & Posch, S. Improving MetFrag with statistical learning of fragment annotations. *BMC Bioinform.* **20**, 376 (2019).
16. Nguyen, D. H., Nguyen, C. H. & Mamitsuka, H. ADAPTIVE: learning data-dependent, concIse molecular vectors for fast, accurate metabolite identification from tandem mass spectra. *Bioinformatics* **35**, i164–i172 (2019).
17. Dührkop, K. et al. SIRIUS 4: a rapid tool for turning tandem mass spectra into metabolite structure information. *Nat. Methods* https://doi.org/10.1038/s41592-019-0344-8 (2019).
18. Wang, F. et al. CFM-ID 4.0: nore accurate ESI-MS/MS spectral prediction and compound identification. *Anal. Chem.* https://doi.org/10.1021/acs.analchem.1c01465 (2021).
19. Wishart, D. S. et al. HMDB 4.0: the Human Metabolome Database for 2018. *Nucleic Acids Res.* **46**, D608–D617 (2017).
20. Kim, S. et al. PubChem in 2021: new data content and improved web interfaces. *Nucleic Acids Res.* **49**, D1388–D1395 (2020).
21. Stanstrup, J., Neumann, S. & Vrhovšek, U. PredRet: prediction of retention time by direct mapping between multiple chromatographic systems. *Anal. Chem.* **87**, 9421–9428 (2015).
22. Low, D. Y. et al. Data sharing in predret for accurate prediction of retention time: application to plant food bioactive compounds. *Food Chem.* **357**, 129757 (2021).
23. Fanali, S., Haddad, P., Poole, C. & Lloyd, D. *Liquid Chromatography: Fundamentals and Instrumentation* (Handbooks in Separation Science, Elsevier Science, 2013).
24. Witting, M. & Böcker, S. Current status of retention time prediction in metabolite identification. *J. Sep. Sci.* **43**, 1746–1754 (2020).
25. Bouwmeester, R., Martens, L. & Degroeve, S. Comprehensive and empirical evaluation of machine learning algorithms for small molecule LC retention time prediction. *Anal. Chem.* **91**, 3694–3703 (2019).
26. Aicheler, F. et al. Retention time prediction improves identification in nontargeted lipidomics approaches. *Anal. Chem.* **87**, 7698–7704 (2015).
27. Samaraweera, M. A., Hall, L. M., Hill, D. W. & Grant, D. F. Evaluation of an artificial neural network retention index model for chemical structure identification in nontargeted metabolomics. *Anal. Chem.* **90**, 12752–12760 (2018).
28. Bonini, P., Kind, T., Tsugawa, H., Barupal, D. K. & Fiehn, O. Retip: retention time prediction for compound annotation in untargeted metabolomics. *Anal. Chem.* https://doi.org/10.1021/acs.analchem.9b05765 (2020).
29. Yang, Q., Ji, H., Lu, H. & Zhang, Z. Prediction of liquid chromatographic retention time with graph neural networks to assist in small molecule identification. *Anal. Chem.* https://doi.org/10.1021/acs.analchem.0c04071 (2021).
30. Bouwmeester, R., Martens, L. & Degroeve, S. Generalized calibration across liquid chromatography setups for generic prediction of small-molecule retention times. *Anal. Chem.* **92**, 6571–6578 (2020).
31. Bach, E., Szedmak, S., Brouard, C., Böcker, S. & Rousu, J. Liquid-chromatography retention order prediction for metabolite identification. *Bioinformatics* **34**, i875–i883 (2018).
32. Liu, J. J., Alipuly, A., Baczek, T., Wong, M. W. & Žuvela, P. Quantitative structure–retention relationships with non-linear programming for prediction of chromatographic elution order. *Int. J. Mol. Sci.* **20**, 3443 (2019).
33. Žuvela, P., Liu, J. J., Wong, M. W. & Baczek, T. Prediction of chromatographic elution order of analytical mixtures based on quantitative structure–retention relationships and multi-objective optimization. *Molecules* **25**, 3085 (2020).
34. Bach, E., Rogers, S., Williamson, J. & Rousu, J. Probabilistic framework for integration of mass spectrum and retention time information in small molecule identification. *Bioinformatics* **37**, 1724–1731 (2021).
35. Tsochantaridis, I., Joachims, T., Hofmann, T. & Altun, Y. Large margin methods for structured and interdependent output variables. *J. Mach. Learn. Res.* **6**, 1453–1484 (2005).
36. Taskar, B., Guestrin, C. & Koller, D. Max-margin Markov networks. *Adv. Neural Inf. Process. Syst.* 16, 25–32 (MIT, 2004).
37. Horai, H. et al. MassBank: a public repository for sharing mass spectral data for life sciences. *J. Mass Spectrom.* **45**, 703–714 (2010).
38. Rogers, D. & Hahn, M. Extended-connectivity fingerprints. *J. Chem. Inf. Model.* **50**, 742–754 (2010).
39. Pence, H. & Williams, A. ChemSpider: an online chemical information resource. *J. Chem. Educ.* **87**, 1123–1124 (2010).
40. Schymanski, E. L. et al. Empowering large chemical knowledge bases for exposomics: PubChemLite meets MetFrag. *J. Cheminform.* https://doi.org/10.21203/rs.3.rs-107432/v1 (2021).
41. Schüller, A., Schneider, G. & Byvatov, E. SmiLib: rapid assembly of combinatorial libraries in smiles notation. *QSAR Comb. Sci.* **22**, 719–721 (2003).

42. Schüller, A., Hähnke, V. & Schneider, G. SmiLib v2.0: a Java-based tool for rapid combinatorial library enumeration. *QSAR Comb. Sci.* **26**, 407–410 (2007).
43. Wainwright, M., Jaakkola, T. & Willsky, A. Tree consistency and bounds on the performance of the max-product algorithm and its generalizations. *Stat. Comput.* **14**, 143–166 (2004).
44. MacKay, D. J. *Information Theory, Inference and Learning Algorithms* (Cambridge Univ. Press, 2005).
45. Pletscher, P., Ong, C. S. & Buhmann, J. Spanning tree approximations for conditional random fields. In *Proc. Twelth International Conference on Artificial Intelligence and Statistics, Proc. Machine Learning Research* Vol. 5 (eds van Dyk, D. & Welling, M.) 408–415 (PMLR, 2009); http://proceedings.mlr.press/v5/pletscher09a.html
46. Su, H. & Rousu, J. Multilabel classification through random graph ensembles. *Mach. Learn.* **99**, 231–256 (2015).
47. Rousu, J., Saunders, C., Szedmak, S. & Shawe-Taylor, J. Kernel-based learning of hierarchical multilabel classification models. *J. Mach. Learn. Res.* **7**, 1601–1626 (2006).
48. Elisseeff, A. & Weston, J. A kernel method for multi-labelled classification. *Adv. Neural Inf. Process. Syst.* **14**, 681–687 (2002).
49. Joachims, T. Optimizing search engines using clickthrough data. In *Proc. Eighth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* 133–142 (ACM, 2002); https://doi.org/10.1145/775047.775067
50. Cheng, T. et al. Computation of octanol-water partition coefficients by guiding an additive model with knowledge. *J. Chem. Inf. Model.* **47**, 2140–2148 (2007).
51. Feunang, Y. D. et al. ClassyFire: automated chemical classification with a comprehensive, computable taxonomy. *J. Cheminform.* **8**, 61 (2016).
52. Bach, E. massbank2db: build a machine learning ready SQLite database from MassBank. *GitHub* https://github.com/bachi55/massbank2db (2022).
53. Gärtner, T. & Vembu, S. On structured output training: hard cases and an efficient alternative. *Mach. Learn.* **76**, 227–242 (2009).
54. Xue, Y., Li, Z., Ermon, S., Gomes, C. P. & Selman, B. Solving marginal map problems with NP oracles and parity constraints. *Adv. Neural Inf. Process. Syst.* **29**, 1135–1143 (2016).
55. Lacoste-Julien, S., Jaggi, M., Schmidt, M. & Pletscher, P. Block-coordinate Frank–Wolfe optimization for structural svms. In *International Conference on Machine Learning* 53–61 (PMLR, 2013).
56. Frank, M. & Wolfe, P. An algorithm for quadratic programming. *Nav. Res. Logist. Q.* **3**, 95–110 (1956).
57. Ralaivola, L., Swamidass, S. J., Saigo, H. & Baldi, P. Graph kernels for chemical informatics. *Neural Netw.* **18**, 1093–1110 (2005).
58. Heller, S. R., McNaught, A., Pletnev, I., Stein, S. & Tchekhovskoi, D. InChI, the IUPAC international chemical identifier. *J. Cheminform.* **7**, 23 (2015).
59. Weininger, D. SMILES, a chemical language and information system. 1. Introduction to methodology and encoding rules. *J. Chem. Inf. Comput. Sci.* **28**, 31–36 (1988).
60. Benton, H. P., Wong, D. M., Trauger, S. A. & Siuzdak, G. XCMS2: processing tandem mass spectrometry data for metabolite identification and structural characterization. *Anal. Chem.* **80**, 6382–6389 (2008).
61. Watrous, J. et al. Mass spectral molecular networking of living microbial colonies. *Proc. Natl Acad. Sci. USA* **109**, E1743–E1752 (2012).
62. Huber, F. et al. matchms—processing and similarity evaluation of mass spectrometry data. *J. Open Source Softw.* **5**, 2411 (2020).
63. Dolan, J. W. Column Dead Time as a Diagnostic Tool. *LCGC North America* **32**, 24–29 (2014).
64. Järvelin, K. & Kekäläinen, J. Cumulated gain-based evaluation of ir techniques. *ACM Trans. Inf. Syst.* **20**, 422–446 (2002).
65. Pedregosa, F. et al. scikit-learn: machine learning in Python. *J. Mach. Learn. Res.* **12**, 2825–2830 (2011).
66. Drucker, H., Burges, C. J., Kaufman, L., Smola, A. J. & Vapnik, V. Support vector regression machines. *Adv. Neural Inf. Process. Syst.* **9**, 155–161 (1997).
67. Bach, E. Retention order support vector machine (ROSVM) *GitHub* https://github.com/bachi55/rosvm (2022).
68. Willighagen, E. L. et al. The chemistry development kit (CDK) v2.0: atom typing, depiction, molecular formulas, and substructure searching. *J. Cheminform.* **9**, 33 (2017).
69. Bach, E. msmsrt_scorer: probabilistic framework for integration of mass spectrum and retention order information. *GitHub* https://github.com/aalto-ics-kepaco/msms_rt_score_integration (2021).
70. Platt, J. Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods. *Adv. Large Margin Classifiers* **10**, 61–74 (2000).
71. Bach, E. Dataset: 'Joint structural annotation of small molecules using liquid chromatography retention order and tandem mass spectrometry data'. *Zenodo* https://doi.org/10.5281/zenodo.5854661 (2022).
72. Bach, E. Result files (ALLDATA): 'Joint structural annotation of small molecules using liquid chromatography retention order and tandem mass spectrometry data with LC-MS^{2}Struct'. *Zenodo* https://doi.org/10.5281/zenodo.6451016 (2022).
73. Bach, E. Result files (ONLYSTEREO): 'Joint structural annotation of small molecules using liquid chromatography retention order and tandem mass spectrometry data'. *Zenodo* https://doi.org/10.5281/zenodo.6037629 (2022).
74. Bach, E. msms_rt_ssvm: implementation of the LC-MS^{2}Struct algorithm. *GitHub* https://github.com/aalto-ics-kepaco/msms_rt_ssvm (2022).
75. Bach, E. Experiments and figure generation for the LC-MS^{2}Struct evaluation. *GitHub* https://github.com/aalto-ics-kepaco/lcms2struct_exp (2022).

Acknowledgements

E.L.S. acknowledges discussions with G. Landrum (ETHZ) and E. Bolton (NCBI/NLM/NIH). We acknowledge CSC-IT Center for Science, Finland, and Aalto Science-IT infrastructure, Finland, for generous computational resources. E.B. thanks K. Dührkop for generating the SIRIUS candidate sets and predicting the SIRIUS MS^2 scores.

Author contributions

E.B. and J.R. designed the research. E.B. implemented the MassBank pre-processing. E.B. developed, implemented and evaluated the computational method. E.B., E.L.S. and J.R. interpreted the results. E.B., E.L.S. and J.R. wrote the manuscript.

Funding

Open Access funding provided by Aalto University. The work by E.B. and J.R. was partially supported by Academy of Finland grants 310107 (MACOME) and 334790 (MAGITICS). E.L.S. acknowledges funding support from the Luxembourg National Research Fund (FNR) for project A18/BM/12341006.

Competing interests

The authors declare no competing interests.

Additional information

Extended data is available for this paper at https://doi.org/10.1038/s42256-022-00577-2.

Supplementary information The online version contains supplementary material available at https://doi.org/10.1038/s42256-022-00577-2.

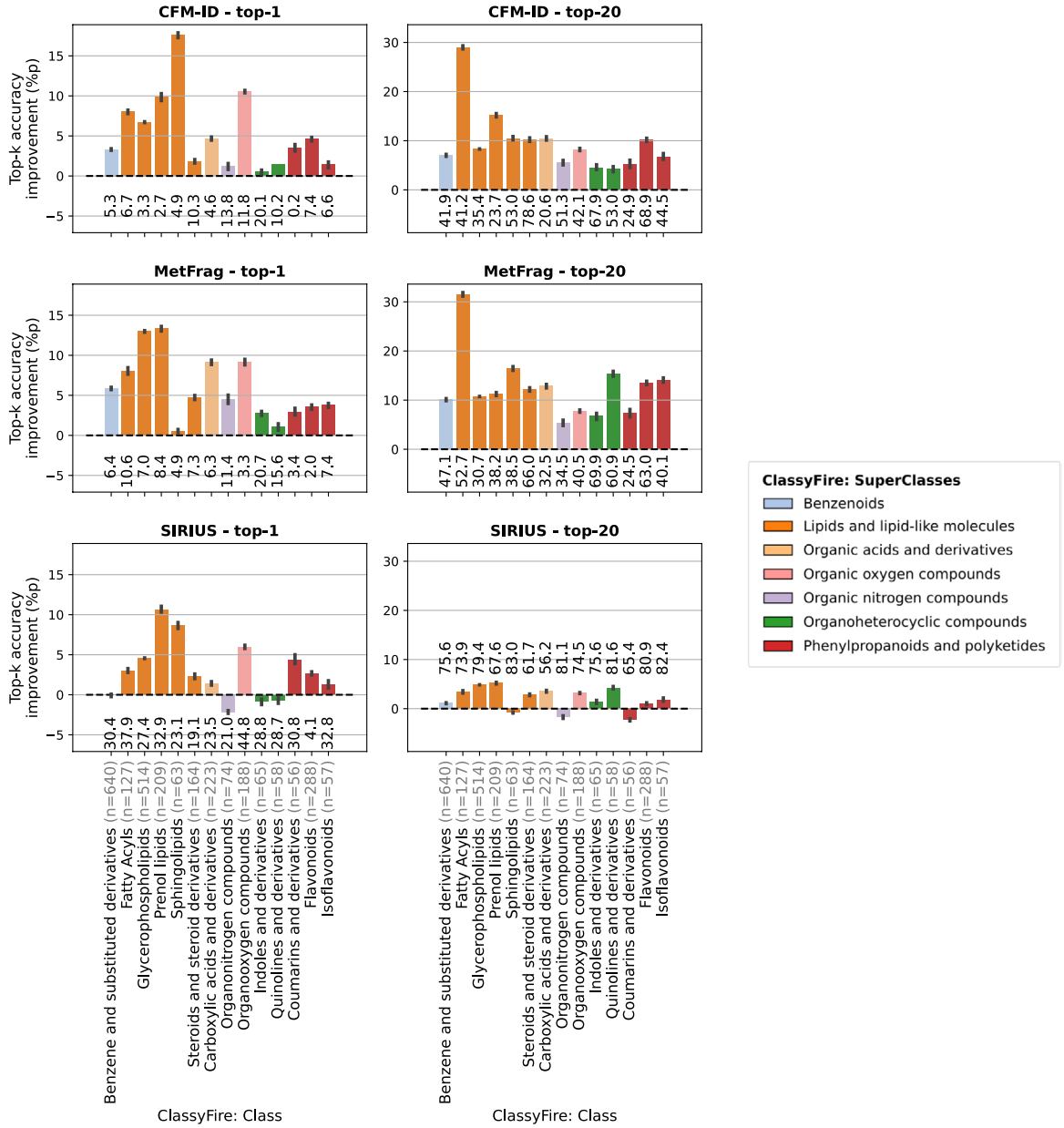
Correspondence and requests for materials should be addressed to Eric Bach or Juho Rousu.

Peer review information *Nature Machine Intelligence* thanks Nicola Zamboni and the other, anonymous, reviewer(s) for their contribution to the peer review of this work.

https://doi.org/10.1038/s42256-022-00577-2

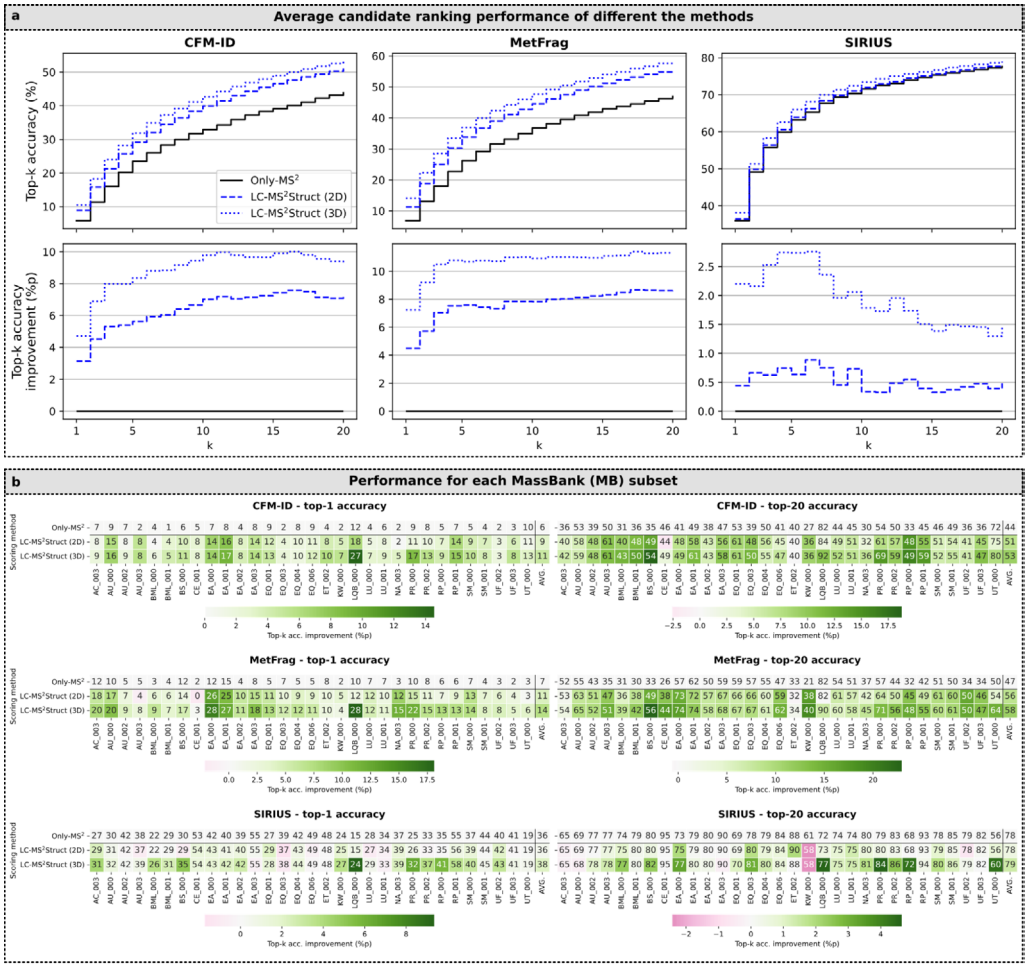
Extended Data Fig. 1 | Distribution of molecule classes in the MassBank (MB) subsets. Distribution of molecule classes in the MassBank (MB) subsets. ClassyFire SuperClass distribution [51] for each MB-subset studied in our experiments. Within each MB-subset, the label 'Other' is assigned to each SuperClass which makes up less than 2.5% of all molecules. The centre label represents the number of examples for the respective MB-subset.

https://doi.org/10.1038/s42256-022-00577-2



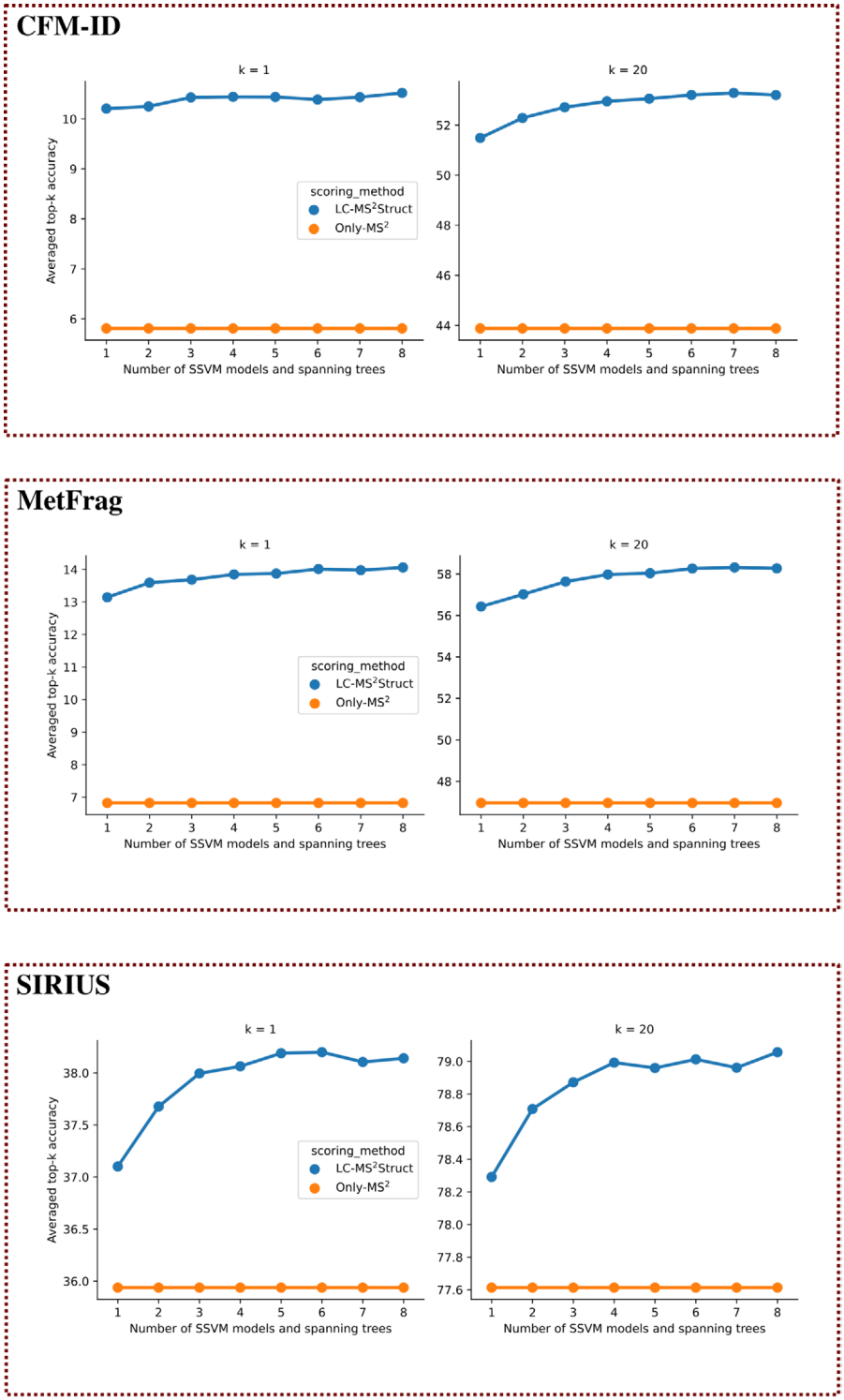
Extended Data Fig. 2 | Performance gain by LC-MS²Struct across ClassyFire Classes annotations. Ranking performance (top-*k*) improvement of LC-MS²Struct compared to Only-MS² (baseline) across ClassyFire Class-level annotations. The Classes (shown in the bars) are colour coded by SuperClasses (see legend). The data is presented as mean values (50 samples) and the error bars show the 95%-confidence interval of the mean estimate (1000 bootstrapping samples). The top-*k* accuracies (%) under the bars show the Only-MS² performance. For each molecule class, the number of unique molecular structures in the class is denoted in the x-axis label (n).

https://doi.org/10.1038/s42256-022-00577-2



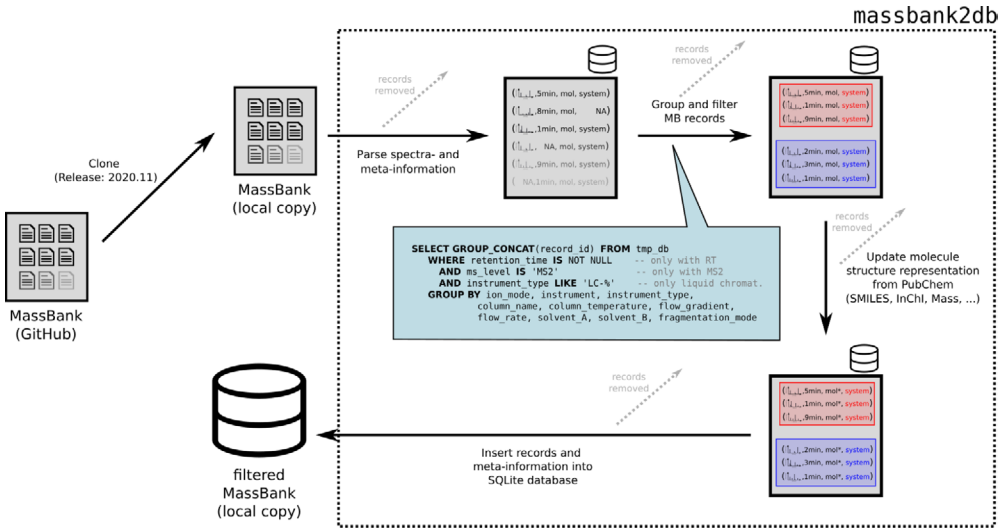
Extended Data Fig. 3 | Performance comparisons using 3D and 2D fingerprints in the ALLDATA setting. Using LC-MS²Struct with different molecule feature representations to identify the correct structure at the level of first InChIKey block (InChIKey-1). **a**: Comparison of the performance, measured by top-*k* accuracy, of LC-MS²Struct using either 2D (no stereochemistry) or 3D (with stereochemistry) molecular fingerprints in the ALLDATA setting. The results shown are averaged accuracies over 354 sample MS feature sequences (LC-MS² experiments). **b**: Average top-*k* accuracies per MassBank (MB) subset rounded to full integers. The colour encodes the performance improvement in percentage units (%p) of each score integration method compared to Only-MS².

https://doi.org/10.1038/s42256-022-00577-2



Extended Data Fig. 4 | Model performance for different number of SSVM models. Performance comparison of LC-MS^{2}Struct against using only-MS^2 information (Only-MS^2) for different number of SSVM models. The performance curves for the three MS^2-scorers are shown separately. The top-*k* accuracies shown are averaged accuracies over 354 sample MS feature sequences (LC-MS^2 experiments) from the ALLDATA setting.

https://doi.org/10.1038/s42256-022-00577-2



Extended Data Fig. 5 | Processing pipeline of the MassBank data. Processing pipeline of the MassBank data. Illustration of the processing pipeline to extract the training data from MassBank. The depicted workflow is implemented in the 'massbank2db' Python package [52].

nature portfolio

Corresponding author(s): Eric Bach, Juho Rousu

Last updated by author(s): 21.10.2022

Reporting Summary

Nature Portfolio wishes to improve the reproducibility of the work that we publish. This form provides structure for consistency and transparency in reporting. For further information on Nature Portfolio policies, see our Editorial Policies and the Editorial Policy Checklist.

Statistics

For all statistical analyses, confirm that the following items are present in the figure legend, table legend, main text, or Methods section.

n/a	Confirmed	
☐	☒	The exact sample size (*n*) for each experimental group/condition, given as a discrete number and unit of measurement
☒	☐	A statement on whether measurements were taken from distinct samples or whether the same sample was measured repeatedly
☒	☐	The statistical test(s) used AND whether they are one- or two-sided *Only common tests should be described solely by name; describe more complex techniques in the Methods section.*
☒	☐	A description of all covariates tested
☒	☐	A description of any assumptions or corrections, such as tests of normality and adjustment for multiple comparisons
☐	☒	A full description of the statistical parameters including central tendency (e.g. means) or other basic estimates (e.g. regression coefficient) AND variation (e.g. standard deviation) or associated estimates of uncertainty (e.g. confidence intervals)
☒	☐	For null hypothesis testing, the test statistic (e.g. *F*, *t*, *r*) with confidence intervals, effect sizes, degrees of freedom and *P* value noted *Give P values as exact values whenever suitable.*
☒	☐	For Bayesian analysis, information on the choice of priors and Markov chain Monte Carlo settings
☒	☐	For hierarchical and complex designs, identification of the appropriate level for tests and full reporting of outcomes
☒	☐	Estimates of effect sizes (e.g. Cohen's *d*, Pearson's *r*), indicating how they were calculated

Our web collection on statistics for biologists contains articles on many of the points above.

Software and code

Policy information about availability of computer code

Data collection	Code is available on Github: https://github.com/bachi55/massbank2db (DOI: https://doi.org/10.5281/zenodo.7029738)
Data analysis	The source code developed for this study is available on GitHub, including the implementation of LC-MS^{2}Struct (v2.13.0, https://github.com/aalto-ics-kepaco/msms_rt_ssvm); scripts to run the experiments (https://github.com/aalto-ics-kepaco/lcms2struct_exp); and the library implementing the MassBank pre-processing (v0.9.0, https://github.com/bachi55/massbank2db). The candidate fingerprints were computed by the ROSVM Python library (v0.5.0, https://github.com/bachi55/rosvm) using RDKit (2021.03.1). The SSVM library uses the max-marginal inference solver implemented by (v0.2.3, https://github.com/aalto-ics-kepaco/msms_rt_score_integration). Furthermore we used the scikit-learn package (v0.24.1).

For manuscripts utilizing custom algorithms or software that are central to the research but not yet described in published literature, software must be made available to editors and reviewers. We strongly encourage code deposition in a community repository (e.g. GitHub). See the Nature Portfolio guidelines for submitting code & software for further information.

Data

Policy information about availability of data

All manuscripts must include a data availability statement. This statement should provide the following information, where applicable:

- Accession codes, unique identifiers, or web links for publicly available datasets
- A description of any restrictions on data availability
- For clinical datasets or third party data, please ensure that the statement adheres to our policy

Our study analyses the MassBank release 2020.11 (https://github.com/MassBank/MassBank-data/releases/tag/2020.11). We provide all data used in our experiments in a format compatible with our framework on Zenodo (https://zenodo.org/record/5854661). The candidate rankings of all LC-MS² experiments are available on Zenodo as well: ALLDATA dataset (https://zenodo.org/record/6451016) and ONLYSTEREO (https://zenodo.org/record/6037629).

Human research participants

Policy information about studies involving human research participants and Sex and Gender in Research.

Reporting on sex and gender	*Use the terms sex (biological attribute) and gender (shaped by social and cultural circumstances) carefully in order to avoid confusing both terms. Indicate if findings apply to only one sex or gender; describe whether sex and gender were considered in study design whether sex and/or gender was determined based on self-reporting or assigned and methods used. Provide in the source data disaggregated sex and gender data where this information has been collected, and consent has been obtained for sharing of individual-level data; provide overall numbers in this Reporting Summary. Please state if this information has not been collected. Report sex- and gender-based analyses where performed, justify reasons for lack of sex- and gender-based analysis.*
Population characteristics	*Describe the covariate-relevant population characteristics of the human research participants (e.g. age, genotypic information, past and current diagnosis and treatment categories). If you filled out the behavioural & social sciences study design questions and have nothing to add here, write "See above."*
Recruitment	*Describe how participants were recruited. Outline any potential self-selection bias or other biases that may be present and how these are likely to impact results.*
Ethics oversight	*Identify the organization(s) that approved the study protocol.*

Note that full information on the approval of the study protocol must also be provided in the manuscript.

Field-specific reporting

Please select the one below that is the best fit for your research. If you are not sure, read the appropriate sections before making your selection.

☒ Life sciences ☐ Behavioural & social sciences ☐ Ecological, evolutionary & environmental sciences

For a reference copy of the document with all sections, see nature.com/documents/nr-reporting-summary-flat.pdf

Life sciences study design

All studies must disclose on these points even when the disclosure is negative.

Sample size	In our experiments we use 7716 (MS², RT) measurements, which represents all available data from MassBank (2020.11) passing our exclusion criteria (see respective section). In the context of small molecule structure annotation method development and evaluation, this dataset size can be considered "large". The dataset size is sufficient to evaluate the performance of a machine learning framework.
Data exclusions	We restrict to MassBank data that has been analyzed using a LC reversed phase (RP) column. We removed molecule from the data if their measure retention time (RT) was less then three times the estimated column dead-time. All exclusion criteria were pre-established.
Replication	The experiments can be replicated by using the datasets and code provided (see the respective availability statements).
Randomization	Randomly drawn structure disjoint training and testing folds were used to evaluate the methods.
Blinding	Not applicable. The chosen nested cross-validation schema ensures that the models' performance is evaluated in an unbiased manner.

Behavioural & social sciences study design

All studies must disclose on these points even when the disclosure is negative.

Study description	*Briefly describe the study type including whether data are quantitative, qualitative, or mixed-methods (e.g. qualitative cross-sectional, quantitative experimental, mixed-methods case study).*
Research sample	*State the research sample (e.g. Harvard university undergraduates, villagers in rural India) and provide relevant demographic information (e.g. age, sex) and indicate whether the sample is representative. Provide a rationale for the study sample chosen. For studies involving existing datasets, please describe the dataset and source.*
Sampling strategy	*Describe the sampling procedure (e.g. random, snowball, stratified, convenience). Describe the statistical methods that were used to predetermine sample size OR if no sample-size calculation was performed, describe how sample sizes were chosen and provide a rationale for why these sample sizes are sufficient. For qualitative data, please indicate whether data saturation was considered, and what criteria were used to decide that no further sampling was needed.*
Data collection	*Provide details about the data collection procedure, including the instruments or devices used to record the data (e.g. pen and paper, computer, eye tracker, video or audio equipment) whether anyone was present besides the participant(s) and the researcher, and whether the researcher was blind to experimental condition and/or the study hypothesis during data collection.*
Timing	*Indicate the start and stop dates of data collection. If there is a gap between collection periods, state the dates for each sample cohort.*
Data exclusions	*If no data were excluded from the analyses, state so OR if data were excluded, provide the exact number of exclusions and the rationale behind them, indicating whether exclusion criteria were pre-established.*
Non-participation	*State how many participants dropped out/declined participation and the reason(s) given OR provide response rate OR state that no participants dropped out/declined participation.*
Randomization	*If participants were not allocated into experimental groups, state so OR describe how participants were allocated to groups, and if allocation was not random, describe how covariates were controlled.*

Ecological, evolutionary & environmental sciences study design

All studies must disclose on these points even when the disclosure is negative.

Study description	*Briefly describe the study. For quantitative data include treatment factors and interactions, design structure (e.g. factorial, nested, hierarchical), nature and number of experimental units and replicates.*
Research sample	*Describe the research sample (e.g. a group of tagged Passer domesticus, all Stenocereus thurberi within Organ Pipe Cactus National Monument), and provide a rationale for the sample choice. When relevant, describe the organism taxa, source, sex, age range and any manipulations. State what population the sample is meant to represent when applicable. For studies involving existing datasets, describe the data and its source.*
Sampling strategy	*Note the sampling procedure. Describe the statistical methods that were used to predetermine sample size OR if no sample-size calculation was performed, describe how sample sizes were chosen and provide a rationale for why these sample sizes are sufficient.*
Data collection	*Describe the data collection procedure, including who recorded the data and how.*
Timing and spatial scale	*Indicate the start and stop dates of data collection, noting the frequency and periodicity of sampling and providing a rationale for these choices. If there is a gap between collection periods, state the dates for each sample cohort. Specify the spatial scale from which the data are taken*
Data exclusions	*If no data were excluded from the analyses, state so OR if data were excluded, describe the exclusions and the rationale behind them, indicating whether exclusion criteria were pre-established.*
Reproducibility	*Describe the measures taken to verify the reproducibility of experimental findings. For each experiment, note whether any attempts to repeat the experiment failed OR state that all attempts to repeat the experiment were successful.*
Randomization	*Describe how samples/organisms/participants were allocated into groups. If allocation was not random, describe how covariates were controlled. If this is not relevant to your study, explain why.*
Blinding	*Describe the extent of blinding used during data acquisition and analysis. If blinding was not possible, describe why OR explain why blinding was not relevant to your study.*

Did the study involve field work? ☐ Yes ☐ No

Field work, collection and transport

Field conditions	*Describe the study conditions for field work, providing relevant parameters (e.g. temperature, rainfall).*
Location	*State the location of the sampling or experiment, providing relevant parameters (e.g. latitude and longitude, elevation, water depth).*
Access & import/export	*Describe the efforts you have made to access habitats and to collect and import/export your samples in a responsible manner and in compliance with local, national and international laws, noting any permits that were obtained (give the name of the issuing authority, the date of issue, and any identifying information).*
Disturbance	*Describe any disturbance caused by the study and how it was minimized.*

Reporting for specific materials, systems and methods

We require information from authors about some types of materials, experimental systems and methods used in many studies. Here, indicate whether each material, system or method listed is relevant to your study. If you are not sure if a list item applies to your research, read the appropriate section before selecting a response.

Materials & experimental systems

n/a	Involved in the study
☒	☐ Antibodies
☒	☐ Eukaryotic cell lines
☒	☐ Palaeontology and archaeology
☒	☐ Animals and other organisms
☒	☐ Clinical data
☒	☐ Dual use research of concern

Methods

n/a	Involved in the study
☒	☐ ChIP-seq
☒	☐ Flow cytometry
☒	☐ MRI-based neuroimaging

Antibodies

Antibodies used	*Describe all antibodies used in the study; as applicable, provide supplier name, catalog number, clone name, and lot number.*
Validation	*Describe the validation of each primary antibody for the species and application, noting any validation statements on the manufacturer's website, relevant citations, antibody profiles in online databases, or data provided in the manuscript.*

Eukaryotic cell lines

Policy information about cell lines and Sex and Gender in Research

Cell line source(s)	*State the source of each cell line used and the sex of all primary cell lines and cells derived from human participants or vertebrate models.*
Authentication	*Describe the authentication procedures for each cell line used OR declare that none of the cell lines used were authenticated.*
Mycoplasma contamination	*Confirm that all cell lines tested negative for mycoplasma contamination OR describe the results of the testing for mycoplasma contamination OR declare that the cell lines were not tested for mycoplasma contamination.*
Commonly misidentified lines (See ICLAC register)	*Name any commonly misidentified cell lines used in the study and provide a rationale for their use.*

Palaeontology and Archaeology

Specimen provenance	*Provide provenance information for specimens and describe permits that were obtained for the work (including the name of the issuing authority, the date of issue, and any identifying information). Permits should encompass collection and, where applicable, export.*
Specimen deposition	*Indicate where the specimens have been deposited to permit free access by other researchers.*

Dating methods

If new dates are provided, describe how they were obtained (e.g. collection, storage, sample pretreatment and measurement), where they were obtained (i.e. lab name), the calibration program and the protocol for quality assurance OR state that no new dates are provided.

☐ Tick this box to confirm that the raw and calibrated dates are available in the paper or in Supplementary Information.

Ethics oversight

Identify the organization(s) that approved or provided guidance on the study protocol, OR state that no ethical approval or guidance was required and explain why not.

Note that full information on the approval of the study protocol must also be provided in the manuscript.

Animals and other research organisms

Policy information about studies involving animals; ARRIVE guidelines recommended for reporting animal research, and Sex and Gender in Research

Laboratory animals

For laboratory animals, report species, strain and age OR state that the study did not involve laboratory animals.

Wild animals

Provide details on animals observed in or captured in the field; report species and age where possible. Describe how animals were caught and transported and what happened to captive animals after the study (if killed, explain why and describe method; if released, say where and when) OR state that the study did not involve wild animals.

Reporting on sex

Indicate if findings apply to only one sex; describe whether sex was considered in study design, methods used for assigning sex. Provide data disaggregated for sex where this information has been collected in the source data as appropriate; provide overall numbers in this Reporting Summary. Please state if this information has not been collected. Report sex-based analyses where performed, justify reasons for lack of sex-based analysis.

Field-collected samples

For laboratory work with field-collected samples, describe all relevant parameters such as housing, maintenance, temperature, photoperiod and end-of-experiment protocol OR state that the study did not involve samples collected from the field.

Ethics oversight

Identify the organization(s) that approved or provided guidance on the study protocol, OR state that no ethical approval or guidance was required and explain why not.

Note that full information on the approval of the study protocol must also be provided in the manuscript.

Clinical data

Policy information about clinical studies

All manuscripts should comply with the ICMJE guidelines for publication of clinical research and a completed CONSORT checklist must be included with all submissions.

Clinical trial registration

Provide the trial registration number from ClinicalTrials.gov or an equivalent agency.

Study protocol

Note where the full trial protocol can be accessed OR if not available, explain why.

Data collection

Describe the settings and locales of data collection, noting the time periods of recruitment and data collection.

Outcomes

Describe how you pre-defined primary and secondary outcome measures and how you assessed these measures.

Dual use research of concern

Policy information about dual use research of concern

Hazards

Could the accidental, deliberate or reckless misuse of agents or technologies generated in the work, or the application of information presented in the manuscript, pose a threat to:

No	Yes	
☒	☐	Public health
☒	☐	National security
☒	☐	Crops and/or livestock
☒	☐	Ecosystems
☒	☐	Any other significant area

Experiments of concern

Does the work involve any of these experiments of concern:

No	Yes	
☒	☐	Demonstrate how to render a vaccine ineffective
☒	☐	Confer resistance to therapeutically useful antibiotics or antiviral agents
☒	☐	Enhance the virulence of a pathogen or render a nonpathogen virulent
☒	☐	Increase transmissibility of a pathogen
☒	☐	Alter the host range of a pathogen
☒	☐	Enable evasion of diagnostic/detection modalities
☒	☐	Enable the weaponization of a biological agent or toxin
☒	☐	Any other potentially harmful combination of experiments and agents

ChIP-seq

Data deposition

☐ Confirm that both raw and final processed data have been deposited in a public database such as GEO.

☐ Confirm that you have deposited or provided access to graph files (e.g. BED files) for the called peaks.

Data access links *May remain private before publication.*	*For "Initial submission" or "Revised version" documents, provide reviewer access links. For your "Final submission" document, provide a link to the deposited data.*
Files in database submission	*Provide a list of all files available in the database submission.*
Genome browser session (e.g. UCSC)	*Provide a link to an anonymized genome browser session for "Initial submission" and "Revised version" documents only, to enable peer review. Write "no longer applicable" for "Final submission" documents.*

Methodology

Replicates	*Describe the experimental replicates, specifying number, type and replicate agreement.*
Sequencing depth	*Describe the sequencing depth for each experiment, providing the total number of reads, uniquely mapped reads, length of reads and whether they were paired- or single-end.*
Antibodies	*Describe the antibodies used for the ChIP-seq experiments; as applicable, provide supplier name, catalog number, clone name, and lot number.*
Peak calling parameters	*Specify the command line program and parameters used for read mapping and peak calling, including the ChIP, control and index files used.*
Data quality	*Describe the methods used to ensure data quality in full detail, including how many peaks are at FDR 5% and above 5-fold enrichment.*
Software	*Describe the software used to collect and analyze the ChIP-seq data. For custom code that has been deposited into a community repository, provide accession details.*

Flow Cytometry

Plots

Confirm that:

☐ The axis labels state the marker and fluorochrome used (e.g. CD4-FITC).

☐ The axis scales are clearly visible. Include numbers along axes only for bottom left plot of group (a 'group' is an analysis of identical markers).

☐ All plots are contour plots with outliers or pseudocolor plots.

☐ A numerical value for number of cells or percentage (with statistics) is provided.

Methodology

Sample preparation	*Describe the sample preparation, detailing the biological source of the cells and any tissue processing steps used.*
Instrument	*Identify the instrument used for data collection, specifying make and model number.*

Software

Describe the software used to collect and analyze the flow cytometry data. For custom code that has been deposited into a community repository, provide accession details.

Cell population abundance

Describe the abundance of the relevant cell populations within post-sort fractions, providing details on the purity of the samples and how it was determined.

Gating strategy

Describe the gating strategy used for all relevant experiments, specifying the preliminary FSC/SSC gates of the starting cell population, indicating where boundaries between "positive" and "negative" staining cell populations are defined.

☐ Tick this box to confirm that a figure exemplifying the gating strategy is provided in the Supplementary Information.

Magnetic resonance imaging

Experimental design

Design type

Indicate task or resting state; event-related or block design.

Design specifications

Specify the number of blocks, trials or experimental units per session and/or subject, and specify the length of each trial or block (if trials are blocked) and interval between trials.

Behavioral performance measures

State number and/or type of variables recorded (e.g. correct button press, response time) and what statistics were used to establish that the subjects were performing the task as expected (e.g. mean, range, and/or standard deviation across subjects).

Acquisition

Imaging type(s)

Specify: functional, structural, diffusion, perfusion.

Field strength

Specify in Tesla

Sequence & imaging parameters

Specify the pulse sequence type (gradient echo, spin echo, etc.), imaging type (EPI, spiral, etc.), field of view, matrix size, slice thickness, orientation and TE/TR/flip angle.

Area of acquisition

State whether a whole brain scan was used OR define the area of acquisition, describing how the region was determined.

Diffusion MRI ☐ Used ☐ Not used

Preprocessing

Preprocessing software

Provide detail on software version and revision number and on specific parameters (model/functions, brain extraction, segmentation, smoothing kernel size, etc.).

Normalization

If data were normalized/standardized, describe the approach(es): specify linear or non-linear and define image types used for transformation OR indicate that data were not normalized and explain rationale for lack of normalization.

Normalization template

Describe the template used for normalization/transformation, specifying subject space or group standardized space (e.g. original Talairach, MNI305, ICBM152) OR indicate that the data were not normalized.

Noise and artifact removal

Describe your procedure(s) for artifact and structured noise removal, specifying motion parameters, tissue signals and physiological signals (heart rate, respiration).

Volume censoring

Define your software and/or method and criteria for volume censoring, and state the extent of such censoring.

Statistical modeling & inference

Model type and settings

Specify type (mass univariate, multivariate, RSA, predictive, etc.) and describe essential details of the model at the first and second levels (e.g. fixed, random or mixed effects; drift or auto-correlation).

Effect(s) tested

Define precise effect in terms of the task or stimulus conditions instead of psychological concepts and indicate whether ANOVA or factorial designs were used.

Specify type of analysis: ☐ Whole brain ☐ ROI-based ☐ Both

Statistic type for inference
(See Eklund et al. 2016)

Specify voxel-wise or cluster-wise and report all relevant parameters for cluster-wise methods.

Correction

Describe the type of correction and how it is obtained for multiple comparisons (e.g. FWE, FDR, permutation or Monte Carlo).

Models & analysis

n/a	Involved in the study
☐	☐ Functional and/or effective connectivity
☐	☐ Graph analysis
☐	☐ Multivariate modeling or predictive analysis

Functional and/or effective connectivity	*Report the measures of dependence used and the model details (e.g. Pearson correlation, partial correlation, mutual information).*
Graph analysis	*Report the dependent variable and connectivity measure, specifying weighted graph or binarized graph, subject- or group-level, and the global and/or node summaries used (e.g. clustering coefficient, efficiency, etc.).*
Multivariate modeling and predictive analysis	*Specify independent variables, features extraction and dimension reduction, model, training and evaluation metrics.*