Multilabeled Classification Approach To Find a Plant Source for Terpenoids

Dimitar Hristozov,*,† Johann Gasteiger,† and Fernando B. Da Costa†,‡

Computer-Chemie-Centrum und Institut für Organische Chemie, Universität Erlangen-Nürnberg, Nägelsbachstr. 25, D-91052 Erlangen, Germany, and Laboratório de Farmacognosia, Faculdade de Ciências Farmacêuticas de Ribeirão Preto, Universidade de São Paulo, Av. do Café s/no., 14040-903, Ribeirão Preto, SP, Brazil

Received May 23, 2007

Recently, we have built a classification model that is capable of assigning a given sesquiterpene lactone (STL) into exactly one tribe of the plant family Asteraceae from which the STL has been isolated. Although many plant species are able to biosynthesize a set of peculiar compounds, the occurrence of the same secondary metabolites in more than one tribe of Asteraceae is frequent. Building on our previous work, in this paper, we explore the possibility of assigning an STL to more than one tribe (class) simultaneously. When an object may belong to more than one class simultaneously, it is called multilabeled. In this work, we present a general overview of the techniques available to examine multilabeled data. The problem of evaluating the performance of a multilabeled classifier is discussed. Two particular multilabeled classification methods—cross-training with support vector machines (ct-SVM) and multilabeled *k*-nearest neighbors (M_L-*k*NN)—were applied to the classification of the STLs into seven tribes from the plant family Asteraceae. The results are compared to a single-label classification and are analyzed from a chemotaxonomic point of view. The multilabeled approach allowed us to (1) model the reality as closely as possible, (2) improve our understanding of the relationship between the secondary metabolite profiles of different Asteraceae tribes, and (3) significantly decrease the number of plant sources to be considered for finding a certain STL. The presented classification models are useful for the targeted collection of plants with the objective of finding plant sources of natural compounds that are biologically active or possess other specific properties of interest.

1. INTRODUCTION

Natural products of plant origin have a crucial role in different areas of research. These compounds belong to different chemical classes (alkaloids, phenolics, terpenoids, etc.) and have chemically diverse and complex structures. Because of these complex structures, many of the natural compounds are hard to synthesize.[1] Natural products have a wealth of applications. Some of them are used as drugs, while others possess important biological properties or are used as dietary supplements, as dyes, flavoring agents, or ingredients in the cosmetics industry.[2] Currently, both academia and industry are interested in finding different plant sources of such compounds.[3] However, finding specific chemical compounds in tens of thousands of plant species can be compared to finding a needle in a haystack.

A frequently used strategy for the targeted collection of plants is the chemotaxonomic approach.[4] It consists of selecting plants that are likely to produce the desired chemical compounds based on the relationship between the taxonomic classification in the plant kingdom and the secondary metabolism of the plants. For example, if one is interested in a special type of terpenoids—the sesquiterpene lactones (STLs)—the plant family of choice is Asteraceae, the sunflower family. Within this family, several subgroups of STL structures with special substitution patterns occur in one or more of the ten Asteraceae tribes.[5,6] However, plants in each of the Asteraceae tribes synthesize STLs, which are somewhat typical for the particular tribe.[7] Thus, the STLs are often used as taxonomic markers to characterize or cluster taxa (tribes, subtribes, genera, species, etc.).[5] Moreover, many of the STLs show biological activities, such as lactopicrin, the bitter principle from chicory (*Cichorium intybus*) roots, and the antiphlogistic compound matricin from chamomile (*Matricaria chamomilla*), among others (see Chart 1).

The aforementioned characteristics of the STLs have always attracted the attention of many scientists, making the STLs one of the most explored classes of terpenoids. Therefore, a method that helps in the identification of possible plant sources for a specific STL has multiple applications. As can be seen from Chart 1, the co-occurrence (or overlap) of STLs across different tribes in Asteraceae is not uncommon. Therefore, a technique that is capable of assigning an STL to more than one tribe is preferable. Such a technique can predict more than one possible source and thus provides an option for selecting the most advantageous plant source. Thus, it may significantly reduce the costs associated with the collection of plant material.

One technique capable of handling data that may belong to more than one class is the so-called "collaborative filtering" method,[8] which has been recently applied to a family of biological targets.[9] Another approach is based on the so-called "multilabeled classification" method. In the following, we first give a general overview of multilabeled classification. Subsequently, a short survey of the literature

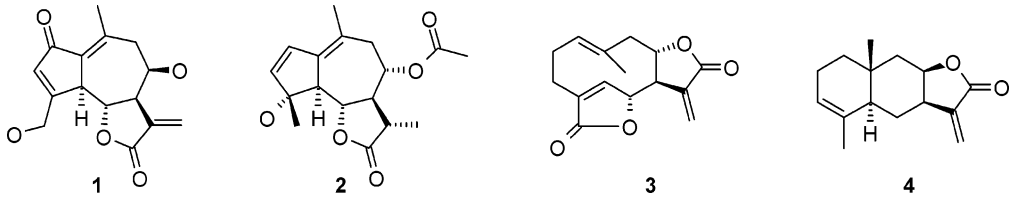
* Corresponding author phone: +49 9131 815668; e-mail: Dimitar.Hristozov@chemie.uni-erlangen.de.

† Computer-Chemie-Centrum und Institut für Organische Chemie, Universität Erlangen-Nürnberg.

‡ Laboratório de Farmacognosia, Faculdade de Ciências Farmacêuticas de Ribeirão Preto, Universidade de São Paulo.

10.1021/ci700175m CCC: $40.75
Published on Web 12/11/2007

Chart 1. Biologically Active STLs of Asteraceae: Lactucin (**1**, Tribe Lactuceae), Matricin (**2**, Tribe Anthemidae), Isabelin (**3**, Tribes Anthemidae and Heliantheae) and Isoallantolactone (**4**, Tribes Anthemidae, Heliantheae, and Inuleae)



on multilabeled classification is presented. In the final part of the Introduction, some measures suitable for assessing the performance of multilabeled classification are presented. Two concrete multilabeled classification implementations are described in the Materials and Methods section. The results of their application to the problem of finding the Asteraceae tribe(s) in which a particular STL can be found are presented in the Results section. The interpretation of the obtained classification models, from a chemotaxonomic point of view, is presented in the Discussion section.

1.1. Multilabeled Classification. Classification is the task of assigning objects to a prespecified set of classes. In traditional classification tasks, these classes are mutually exclusive by definition. Various learning methods have been developed to address such problems.[10] With any of these methods, errors occur when the classes overlap in the selected feature space (see Figure 1).

However, in some classification tasks, the assumption of mutual exclusiveness of the classes is violated by definition. Thus, for example, in text categorization, a document may belong to multiple genres;[11 12] a biologically active compound may exhibit more than one activity; in biochemistry, a gene may have multiple functions, yielding multiple labels;[13] a plant secondary metabolite may appear in more than one taxa[5,7] (tribe, subtribe, genus, etc.). When an object may belong to more than one class, it is called multilabeled (see Figure 2).

The most common approach to examine multilabeled data is to assign each object to the class it is most likely to belong, by some perhaps subjective criterion. For example, one may assign a co-occurring STL as belonging only to the tribe from which it was most frequently reported. Following this approach, we recently built a *k*-nearest neighbors (*k*NN) classifier with good performance.[14] However, it has been shown that the decision borders in this case are additionally blurred by STLs that simultaneously occur in several tribes.

Another possible method for handling multilabeled data is to ignore the multilabeled instances when training the classifier. This will help the underlying method in finding the decision borders between classes. Serious drawbacks of this approach include the following: (a) available data are discarded; (b) predictions for multilabeled samples are likely to be incorrect or, at best, incomplete; and (c) important and characteristic chemical groups are not considered.

A straightforward approach is to consider all objects with multiple labels as a new class (i.e., the "mixed" class). An important limitation is that the data belonging to multiple classes are usually too sparse to build useful models. Although ∼10% of STLs in our data set belong to more than one tribe (cf. Table 1 in section 2, Material and Methods), many combined classes are extremely small (containing less than five STLs) which effectively prevents building any useful classification model for such "combined" classes.

Boutell et al.[15] have introduced the concept of "cross-training". In this approach, the multilabeled data are used more than once when training the classifier (i.e., using each multilabeled object as a positive example for each of the classes to which it belongs). Figure 3 shows the decision borders obtained by cross-training in the two-class scenario. The multilabeled instances (marked by a star and a circle in the figure) are considered to belong to the "star" class when training a classifier for this class and to the "circle" class when a classifier for the circle class is built. They are not used as a negative example for either the star or circle classes. The central area between the decision borders (dashed curves) belongs to both classes simultaneously. When classifying a pattern in this area, both classifiers are expected to classify it as an instance of each class. In such a case, the pattern will obtain multiple labels (star and circle).

The advantage of the cross-training is that it allows most of the binary classifiers, which output real-valued scores, to be turned into multilabeled classifiers. This is done by applying some predefined criteria (cf. subsection 2.3.1 in the Materials and Methods section) to transform the real-valued scores to labels. The cross-training does overcome the

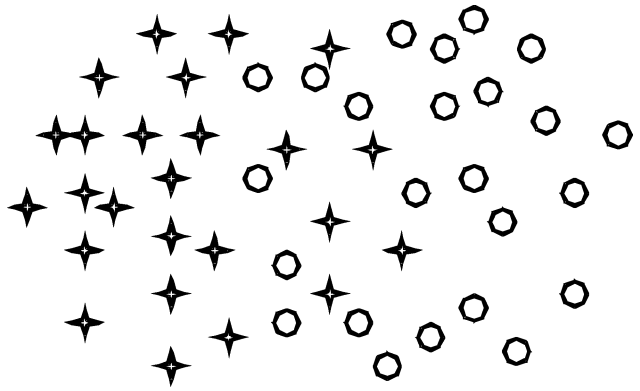


Figure 1. Typical classification problem: the two classes contain instances that are difficult to separate in the feature space.

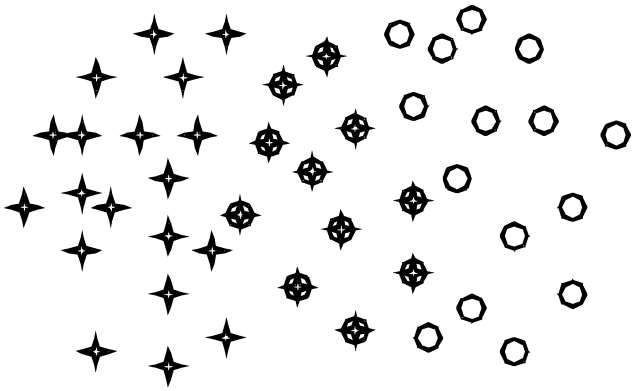


Figure 2. Multilabeled classification problem: the data marked with both symbols belong to both base classes simultaneously.

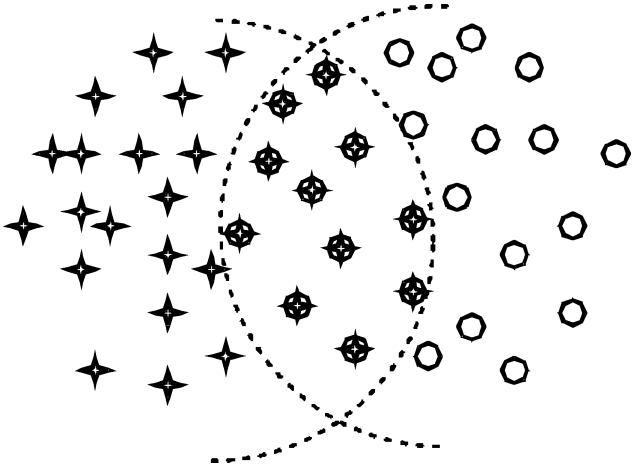


Figure 3. Decision borders obtained via cross-training. The area between the dashed lines belongs to both the "star" and "circle" classes.

problem of the sparse multilabeled data, because no "combined" classes are created. An inherited limitation is that it is based on the one-against-all approach, which may lead to highly unbalanced classifiers.[16] In a chemical context, cross-training using multiple logistic regression as a classifier was applied[17] to classify the compounds into different toxic modes of actions.

1.2. Literature on Multilabeled Classification. The literature on multilabeled classification is generally related to text classification. The first approach that addresses this problem was reported by Schapire and Singer[12] and is called BoosTexter. It is an extended version of the popular ensemble learning method AdaBoost.[18] Following this work, multi-labeled learning has attracted more attention. McCallum[11] proposed a Bayesian approach to multilabeled document classification. Elisseeff and Weston[19] proposed a kernel method for multilabeled classification, based on a special cost function ("ranking loss") and the corresponding margin for multilabeled models. The popular C4.5 decision tree[20] has been adapted[13] to handle multilabeled data through modifying the definition of entropy. Two probabilistic generative models for multilabeled text called Parametric Mixture Models (PMM1 and PMM2) have been proposed.[21] Alternating decision trees[22] also have been extended to the multilabeled case.[23] As already mentioned, Boutel et al.[15] applied multilabeled learning to scene classification. They decomposed the multilabeled problem into multiple independent binary classification problems—one per each base class—and utilized support vector machines (SVMs) as a classifier. Recently, Zhang and Zhou[24] proposed an extended version of the *k*-nearest neighbors (*k*NN) algorithm that is capable of handling multilabeled data. Their approach is based on the label sets of the *k*NN algorithm and utilizes the maximum a posteriori (MAP) principle to determine the label set of a new instance. Both the prior and the posterior probabilities are required and can be estimated from the training set. In addition, the method is capable of ranking each of the base class labels. The proposed multilabeled *k*NN, which is called M_L-*k*NN, was successfully applied to a yeast gene functional data set, yielding a performance that is comparable to an SVM-based method.

1.3. Assessing the Performance of a Multilabeled Classification. An important question related to multilabeled classification is how to assess the model performance. The measures usually used in a single-label classification include precision, recall, accuracy, and F-measure.[10] In a multilabeled case, the evaluation of the model performance is more complicated, because a result can be fully correct, partly correct, or fully wrong. As a simple example, let us consider an object that belongs to two (c_1 and c_2) out of four (c_1, c_2, c_3, and c_4) possible classes. All the following outcomes are possible: (1) c_1 and c_2 is correct; (2) c_1 is partly correct; (3) c_1 and c_3 is partly correct; (4) c_1, c_3 and c_4 is partly correct; (5) c_3 and c_4 is wrong. All these results differ in their degrees of correctness. Given a data set D containing m instances with Q possible classes, for each pattern x, let Y_x be the set of truth labels and P_x be the set of predicted labels. The easiest way to access the accuracy of a multilabeled classification is to use the Hamming loss (HL(D)), which is defined as

$$\mathrm{HL}(D) = \frac{1}{m}\sum_{i=1}^{m}\left(\frac{1}{Q}\right)|P_{x_i} \varnothing Y_{x_i}| \qquad (1)$$

where $\varnothing$ represents the symmetric difference between the two sets. The smaller the value of HL(D), the better the classifier performance. When $|Y_{x_i}| = 1$ for all instances, a multilabeled system is, in fact, a multiclass single-labeled one and the Hamming loss is $2/Q$ times the loss of the usual classification error.

Another score was proposed by Boutell et al.[15] They introduced the so-called "α-evaluation", which is a generalized version of the Jaccard's similarity metric.[25] In addition to the aforementioned definitions, let $M_x = Y_x - P_x$ (missed labels) and $F_x = P_x - Y_x$ (false positive labels). As a result, the prediction of each instance is scored according to

$$\mathrm{score}(x) = \left(1 - \frac{|\beta M_x + \gamma F_x|}{|Y_x \cup P_x|}\right)^{\alpha} \quad (\text{for } \alpha \geq 0,\ 0 \leq \beta, \gamma \leq 1, \beta = 1 | \gamma = 1) \qquad (2)$$

The constraints on β and γ are chosen to constrain the score to be non-negative. These parameters allow false positives and misses to be penalized differently, allowing customization of the measure to the task at hand. Setting $\beta = \gamma = 1$ yields a simpler formula:

$$\mathrm{score}(x) = \left(\frac{|Y_x \cap P_x|}{|Y_x \cup P_x|}\right)^{\alpha} \qquad (\text{for } \alpha \geq 0) \qquad (3)$$

The parameter α is called the "forgiveness rate", because it reflects how much to forgive errors made in predicting labels. Small values of α are more aggressive, and larger values are conservative (penalizing errors more severely). Using this score, the authors[15] defined the recall and precision of multilabeled classes as well as the accuracy on a given testing set. The multilabeled accuracy on a data set D of size m is defined as

$$\mathrm{accuracy}_{\mathrm{ML}}(D) = \frac{1}{m}\sum_{i=1}^{m}\mathrm{score}(x_i) \qquad (4)$$

In addition to the aforementioned definitions, let $H_x^c = 1$ if $c \in Y_x$ and $c \in P_x$ ("hit" label), $H_x^c = 0$ otherwise. Analogously, let $\tilde{Y}_x^c = 1$ if $c \in Y_x$, $\tilde{Y}_x^c = 0$ otherwise; and let

$\tilde{P}_x^c = 1$ if $c \in P_x$, let $\tilde{P}_x^c = 0$ otherwise. Consequently, base-class recall and precision on a data set D are defined as follows:

$$\text{recall}_c = \frac{\sum_{x \in D} H_x^c}{\sum_{x \in D} \tilde{Y}_x^c} \tag{5}$$

$$\text{precision}_c = \frac{\sum_{x \in D} H_x^c}{\sum_{x \in D} \tilde{P}_x^c} \tag{6}$$

The accuracy$_{ML}$ (eq 4), in addition to the base-class recall (eq 5) and precision (eq 6), allow a comparison between multilabeled and single-labeled classification schemes.

All the measures discussed so far are based on the actual labels assigned to an instance. However, in most cases, the learning method will produce a ranking function, $f(x,\cdot)$, which, for a given instance x, will order the labels in Ψ (where $\Psi = \{1, 2, ..., Q\}$ is the complete set of labels (i.e., in this study, this set contains the seven Asteraceae tribes)). That is, a label (i.e., class, l_1) is considered to be ranked higher than l_2 if $f(x,l_1) > f(x,l_2)$. Based on this ranking function, different performance measures can be defined. The first measure is called one-error:

$$\text{one - err}_D(f) = \frac{1}{m} \sum_{i=1}^{m} H(x_i) \tag{7a}$$

where

$$H(x_i) = \begin{cases} 0 & (\text{if } \arg\max_{k \in \Psi_x} f(x_i,k) \in Y_{x_i}) \\ 1 & (\text{otherwise}) \end{cases} \tag{7b}$$

The smaller the value of this measure, the better the performance. For single-label classification problems, the one-error measure is identical to the ordinary classification error.

The second ranking-based measure is called coverage and is defined as

$$\text{coverage}_S(f) = \frac{1}{m} \sum_{i=1}^{m} |C(x_i)| - 1 \tag{8a}$$

where

$$C(x_i) = \{l | f(x_i,l) \geq f(x_i,l_i'),\ l \in \Psi\} \tag{8b}$$

and

$$l_i' = \arg\min_{k \in Y_x} f(x_i,k) \tag{8c}$$

It measures how far we need, in average, to go down the list of labels to cover all possible labels assigned to an instance. The smaller its value, the better the performance.

The last ranking-based measure to be introduced is the average precision (denoted as ave_prec$_D$ here), which was originally used in information retrieval systems.[26] Nevertheless, it can be used to measure the effectiveness of the label rankings:

$$\text{ave\_prec}_D = \frac{1}{m} \sum_{i=1}^{m} \frac{1}{|Y_{x_i}|} R(x_i) \tag{9a}$$

where

$$R(x_i) = \sum_{k \in Y_{x_i}} \frac{|\{l | f(x_i,l) \geq f(x_i,k),\ l \in Y_{x_i}\}|}{|\{l | f(x_i,l) \geq f(x_i,k),\ l \in \Psi\}|} \tag{9b}$$

The average precision evaluates the average fraction of labels, ranked above a particular label $l \in Y_{x_i}$, which are actually in Y_{x_i}. When the value of the average precision is equal to one, the system achieves the perfect performance. The larger its value, the better the performance.

We have briefly introduced the concept of multilabeled classification and different methods to assess its performance. In the remainder of this paper, we first describe two particular multilabeled classification techniques in more detail and then, afterward, show their application to the classification of natural products from plants. The objectives of the present work are (1) to introduce the concepts of multilabeled classification; (2) to build and to evaluate a multilabeled classification model that is capable of relating STLs from Asteraceae to the tribe(s) from which they originate, taking into account skeletal types and substitutional patterns; (3) to compare two different methods for multilabeled classification (SVM-based cross-training and multilabeled k-nearest neighbors (M$_L$-kNN)); and (4) to interpret the results from a chemotaxonomic point of view.

2. MATERIALS AND METHODS

2.1. Datasets. The dataset given in the reported work of Hristozov et al.,[14] which consisted of 921 STLs, was used. Table 1 gives the distribution of the dataset ($N = 921$) into the seven corresponding Asteraceae tribes, the abbreviations used in this paper for these tribes, and the respective number of structures present in the training, validation, and test sets. All structures were assigned to their corresponding tribe(s) according to the current taxonomic classification.[27] For each STL, all reported sources were checked. When a structure was reported in more than one tribe, it was assigned to each of these tribes (i.e., it has multiple labels). The occurrence of such cases is given in parentheses in Table 1.

The complete data set of 921 STLs was split semirandomly (the corresponding distribution of STLs in the tribes was preserved) in three subsets: the training data was ~70% of the dataset, whereas the validation and test set data each represented ~15% of the structures (cf. Table 1).

2.2. Structure Representation. All STLs were represented by their RDF codes,[28] which were calculated using the three-dimensional (3D) structures. Single, low-energy 3D conformations were generated for the STLs from their two-dimensional (2D) constitution using CORINA.[29,30] The RDF codes were calculated according to the following equation:

$$g(r) = \sum_{i=1}^{N-1} \sum_{j>i}^{N} A_i A_j \exp[-B(r - r_{ij})^2] \tag{10}$$

where N is the number of atoms in a molecule; A_i and A_j are

Table 1. Overview of the Dataset Used in This Study (Comprised of 921 Structures of STLs and the Respective Tribes from Which the STLs Were Isolated)[a]

		Training Set		Validation Set		Test Set	
tribe	abbreviation	single	multilabeled	single	multilabeled	single	multilabeled
Anthemideae	ANT	109	26	20	10	25	5
Cardueae	CAR	41	26	10	4	11	2
Eupatorieae	EUP	123	36	27	8	29	7
Heliantheae	HLT	178	51	41	11	41	6
Inuleae	INU	29	21	5	4	5	7
Lactuceae	LAC	30	11	6	3	6	4
Vernonieae	VER	51	18	11	2	12	2
total (921)		561	80[b]	120	18[b]	129	13[b]

[a] Single-labeled compounds occur in one tribe only; multilabeled compounds occur in several tribes. [b] The number is smaller than the sum of the corresponding column because a multilabeled STL is counted toward each individual tribe.

properties associated with the atoms *i* and *j*, respectively; r_{ij} represents the distance between atoms *i* and *j*; and *B* is a smoothing factor. The aforementioned formula was applied, with the property A set, to the atomic number of the considered atom and 64-dimensional RDF codes were calculated using the descriptor calculation package ADRIANA.Code.[31] The function $g(r)$ was defined in the interval of 2.0−9.0 Å.

Note that the STLs used in this study possess a certain degree of conformational flexibility. However, unlike the classification of compounds according to their biological activity, where a change in the conformation may render a compound inactive, we were more concerned with the skeletal types and substitution features, which may help us to identify a plausible plant source for a given STL. Nevertheless, the STLs are ultimately formed into specific enzymes pockets and utilization of a 3D descriptor is relevant, as supported by our previous experience.[14]

2.3. Classification Methods. 2.3.1. Cross-Training with Support Vector Machines (ct-SVMs). A detailed description of the SVM learning technique is outside of the scope of this article. Several comprehensive texts on this subject exist,[16,32] as well as practical guides.[33] In this work, SVM classifiers with radial basis function (RBF) kernel were used. All calculations were performed in R,[34] using the package e1071.[35] The "one-against-all" strategy, as described in the Introduction, was applied, resulting in seven binary classifiers—one for each tribe of Asteraceae. Each binary classifier was separately optimized for optimal performance as suggested,[33] via 10-fold cross-validation and with the validation set (cf. Table 1). Three testing criteria to transform the SVM scores into labels were used:[15]

(1) P-criterion, in which the test data are labeled by all of the classes corresponding to positive SVM scores; if no score is positive, the pattern is labeled as "unknown".

(2) T-criterion, which is similar to the P-criterion but uses the Closed World Assumption (CWA), in which all examples belong to at least one of the *Q* classes; if all SVM scores are negative, the pattern is labeled to the SVM that produces the top (less-negative) score.

(3) C-criterion, in which the decision is dependent on the closeness between the top SVM scores, regardless of their sign; how close two scores must be can be determined via cross-validation, on a hold-out set or using the MAP principle. In this work, the second of the aforementioned approaches, based on a hold-out set, was used. The validation set (cf. section 2.1, Datasets) was used to determine how close two scores must be. This resulted in a value of 0.1 for the acceptable difference between two scores.

2.3.2. Multilabeled *k*NN (M_L-*k*NN) Method. This method belongs to the family of "lazy" learners. It is memory-based and requires no model to be fitted.[36] All training examples are stored in the memory, and the prediction of a new pattern is made by finding its *k*-nearest neighbors, in terms of some predefined distance measure and averaging the values of their known class. In this study, the measure was the Euclidian distance, which is the most commonly used one.

However, the commonly used *k*-NN method assigns an instance to exactly one class. As mentioned in the Introduction, Zhang and Zhou[24] described a modified version of the algorithm, which can handle multilabeled data (M_L-*k*NN). In this work, the M_L-*k*NN method was implemented as an R[34] script, following the algorithm described in the original article. When compared to the standard *k*-NN classification, the main difference is that, instead of selecting the class for a new instance by majority vote, the M_L-*k*NN uses the MAP principle, which combines prior and posterior probabilities, calculated from the training set, to assign labels to an instance as well as to rank the labels. It is worth noting that, although it is not discussed by the authors,[24] as occurred in the SVM approach, there are cases where the M_L-*k*NN method failed to assign an instance to any one of the known classes; therefore, in this work, the same three testing criteria as described in Section 2.3.1 (P-, T-, and C-criteria) were applied to the label rankings, outputted by M_L-*k*NN (see section 3, Results). The best threshold used with the C-criterion (0.05) and the best value of *k* (5) were determined to give the best performance when applied to the validation set.

2.4. Model Validation and Performance Measures. The proposed models using cross-training SVM and M_L-*k*NN were validated using a test set that consisted of 142 STLs (cf. Table 1). This set was not used in the model building phase (i.e., in determining the best model parameters). The test set was submitted through the final models and the performance was assessed only after the final models had been built. In addition, using the full set of 921 STLs and the settings previously described, the performance of the two methods was assessed by means of ten times 10-fold cross-validation.

An overview of the multilabeled classification measures has already been presented in the Introduction. In this work, for the two models (ct-SVM and M_L-*k*NN), six different performance measures, arranged in two distinct groups, were

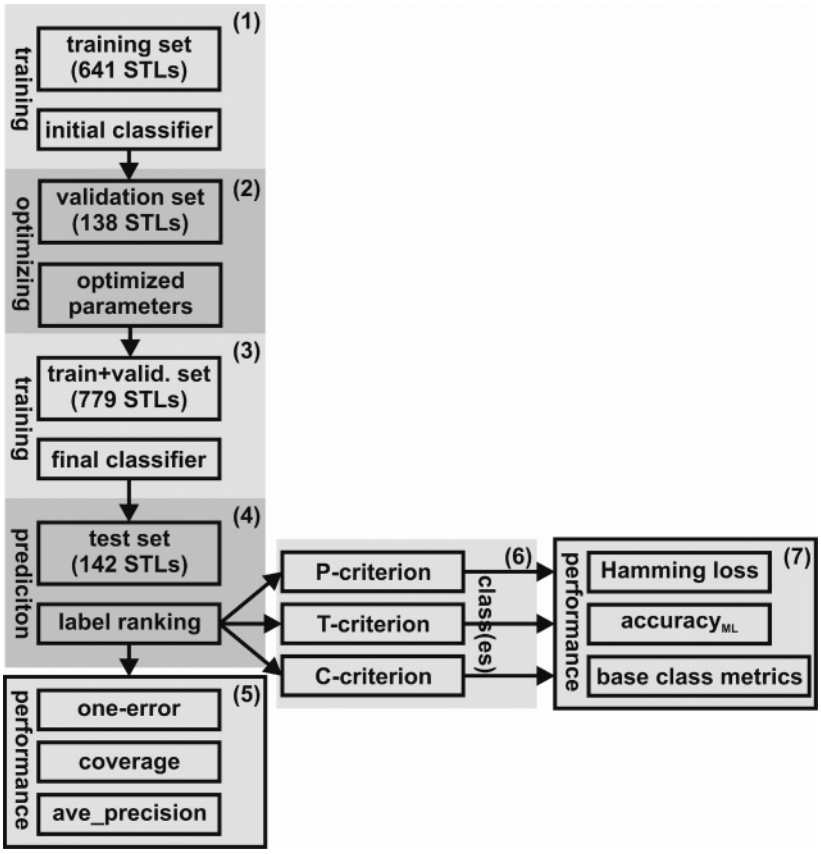


Figure 4. Workflow for obtaining a multilabeled classifier and assessing its performance.

Table 2. Overall Performance Measures and Base-Class Recall and Precision for the ct-SVM Model Applied to the Test Set[a]

	P-Criterion[c] Hamming loss = 0.054, Accuracy$_{ML(\alpha=1)}$ = 0.838		T-Criterion Hamming loss = 0.080, Accuracy$_{ML(\alpha=1)}$ = 0.746		C-Criterion[d] Hamming loss = 0.081 (0.094), Accuracy$_{ML(\alpha=1)}$ = 0.753 (0.709)	
tribe[b]	recall	precision	recall	precision	recall	precision
ANT (30)	0.880	0.786	0.867	0.684	0.867	0.667
CAR (13)	0.727	0.800	0.692	0.750	0.769	0.667
EUP (36)	0.808	0.875	0.667	0.800	0.667	0.800
HLT (47)	0.974	0.860	0.830	0.830	0.872	0.837
INU (12)	0.333	0.750	0.333	0.667	0.333	0.667
LAC (10)	0.571	1.000	0.500	0.714	0.500	0.625
VER (14)	0.750	1.000	0.714	0.833	0.714	0.833

[a] The measures are based on the classes (tribes) to which an STL was assigned under the corresponding criterion (P-, T-, and C-criteria). [b] The number of STLs from the test set belonging to each tribe (including multilabeled STLs) is given in parentheses. [c] Thirty STLs (ca. 21%) were not assigned to any of the seven Asteraceae tribes. [d] The mean values of the overall performance measures obtained with ten times 10-fold cross-validation under C-criterion are given in parentheses.

used: Hamming loss (eq 1), accuracy$_{ML(\alpha=1)}$ (eq 4), and the base class metrics (recall and precision) (eqs 5 and 6, respectively) represent three of the measures; these measures require that a set of labels be assigned to each instance, and, therefore, they are associated with the three testing criteria (P-, T-, and C-criterion; see Section 2.3.1). The other three measures are one-error, coverage, and average precision (eqs 7, 8, and 9, respectively). They are calculated based solely on the real-valued scores that have been output by the classifier and, as such, are independent of the testing criteria. The overall workflow is outlined in Figure 4.

Using the training set, an initial classifier (ct-SVM or M_L-*k*NN) is built (step 1). In step 2, the classifier parameters, as well as the threshold for the C-criterion, are optimized, to give the best performance on the validation set. Afterward, the training and validation sets are merged and a new classifier is built (step 3), using the parameters obtained in step 2. The final classifier obtained is applied to the test set in step 4. For each instance (structure of an STL), the base classes (tribes of Asteraceae) are ranked. Based on this ranking, the classifier performance can be assessed with the corresponding measures, according to step 5. By utilizing the different criteria (step 6) described in section 2.3.1, any STL can be assigned to the corresponding tribe(s). Based on this assignment, the classifier performance can be assessed by the additional measures, as shown in step 7.

3. RESULTS

3.1. Cross-Training SVM (ct-SVM) Models. Table 2 gives an overview of the performance, based on the actually assigned classes (tribes). The base-class metrics recall and precision, which have been calculated according to eqs 5

Table 3. Overall Performance Measures and Base-Class Recall and Precision for the M_L-*k*NN Model with $k = 5$ Applied on the Test Set[a]

	P-Criterion[c] Hamming Loss = 0.081; $Accuracy_{ML(\alpha=1)} = 0.754$		T-Criterion Hamming Loss = 0.096; $Accuracy_{ML(\alpha=1)} = 0.695$		C-Criterion[d] Hamming Loss = 0.098 (0.121); $Accuracy_{ML(\alpha=1)} = 0.698$ (0.618)	
class (tribe)[b]	recall	precision	recall	precision	recall	precision
ANT (30)	0.769	0.800	0.767	0.742	0.767	0.742
CAR (13)	0.545	0.667	0.615	0.667	0.615	0.667
EUP (36)	0.538	0.824	0.556	0.714	0.639	0.657
HLT (47)	0.927	0.776	0.894	0.737	0.915	0.729
INU (12)	0.250	0.667	0.167	0.667	0.167	0.667
LAC (10)	0.625	0.833	0.500	0.833	0.500	0.833
VER (14)	0.750	0.857	0.571	0.667	0.571	0.615

[a] The measures are based on the classes (tribes) to which an STL was assigned under the corresponding criterion (P-, T-, and C-criteria). [b] The number of STLs from the test set belonging to each tribe (including multilabeled STLs) is given in parentheses. [c] Thirty three STLs (ca. 23%) were not assigned to any of the seven Asteraceae tribes. [d] The mean values of the overall performance measures obtained with ten times 10-fold cross-validation under C-criterion are given in parentheses.

Table 4. Performance Measures for the ct-SVM and M_L-*k*NN ($k = 5$) Models Applied on the Test Set[a]

measure	ct-SVM	M_L-KNN	short description
one-error	0.204 (0.238)	0.268 (0.332)	Gives the ratio of the number of STLs that were not found in the top-ranked tribe to the total number of STLs. Bound between 0 and 1. The smaller the value, the better the performance.
coverage	1.563 (1.824)	1.711 (1.938)	Indicates to how many tribes an STL must be assigned on average to ensure that all true tribes are included in the prediction. Bound between 1 and 7 (the number of classes). The smaller the value, the better the performance.
average precision	0.876 (0.840)	0.832 (0.788)	Shows how often the true tribes are top-ranked. Bound between 0 and 1. The larger the value, the better the performance

[a] The mean values obtained with ten times 10-fold cross-validation are given in parentheses. The measures are based on the label rankings produced by each method

and 6, respectively, are also given. Under the P-criterion, 30 STLs (ca. 21%) were not assigned to any of the seven classes.

3.2. M_L-*k*NN Models. Table 3 gives an overview of the M_L-*k*NN performance, based on the actually assigned classes (tribes). In addition, the base-class metrics recall and precision, which have been calculated according to eqs 5 and 6, respectively, are also given. Under the P-criterion, 33 STLs (*ca*. 23%) were not assigned to any of the seven classes.

3.3. Measures Based on the Label Rankings. The classification performances of the two multilabeled classifiers, based on the label rankings (one-error, coverage, and average precision), which have been applied to the test set, and the mean values obtained with ten times 10-fold cross-validation are shown in Table 4.

3.4. Comparison to a Single-Labeled Classifier. As a baseline, Table 5 compares the accuracy of the *k*-NN single-labeled classifier as we have previously reported[14] with the accuracy of the two multilabeled methods under the C-criterion.

4. DISCUSSION

The main objective of this work was to build and evaluate a multilabeled classification model (Figure 4) that is capable of assigning STLs from Asteraceae species to their corresponding tribes, based on skeletal types and substitutional patterns. Such a model can further be used for targeted collection of plants with the goal of isolating specific STL. To accomplish our goal, two multilabeled classification techniques were applied: a cross-training with support vector machine as binary classifiers (ct-SVM) and a modified

Table 5. Accuracy of a Single-Labeled *k*-nearest Neighbor Classifier with $k = 1$ (cf. Hristozov et al.[14]) and That of the Two Multilabeled Methods under C-Criterion

	1-NN[a]	SVM	M_L-*k*NN
accuracy	0.722	0.753	0.698

[a] Single-labeled *k*-nearest neighbor (*k*NN) with $k = 1$ (see Hristozov et al.[14]).

version of the *k*-nearest neighbor (M_L-*k*NN) method. Both methods are able to assign a sample to more than one class and proved to be appropriate for the building of models with good performance.

4.1. Comparison of the Classification Methods. 4.1.1. Based on the Actual Tribes to Which an STL Has Been Assigned (Tables 2 and 3). The following discussion is based on the results under C-criterion (i.e., the last two columns in Tables 2 and 3). However, the same trend is observed with the other two criteria.

Starting with the overall measures (Hamming loss and the multilabeled accuracy), one can see that both methods performed well on the test set. Remember that, for a classifier that performs perfectly, the Hamming loss (eq 1) is equal to zero. Therefore, the low values obtained for Hamming loss (0.081 for ct-SVM and 0.098 for M_L-*k*NN) show good performance. On the other hand, the multilabeled accuracy (eq 4) is equal to one when the underlying classifier performs perfectly and zero when all predictions are wrong. Once again, the multilabeled accuracy values obtained (0.753 for ct-SVM and 0.698 M_L-*k*NN) show good performance. According to both metrics, the ct-SVM method performs better than the M_L-*k*NN method. The superiority of the ct-SVM is further confirmed when the cross-validated results

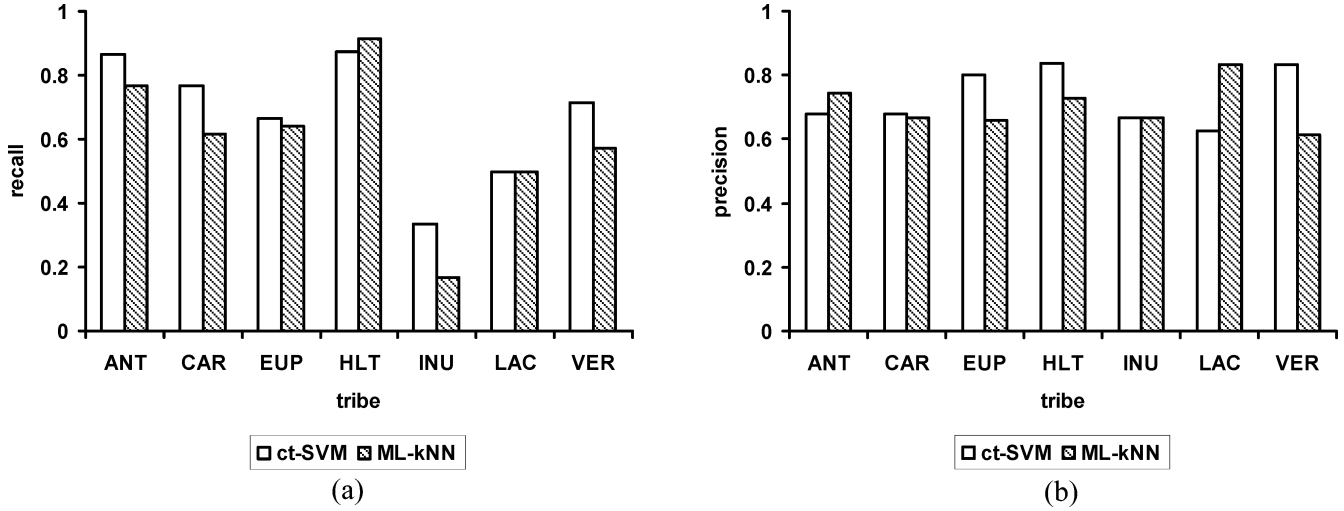


Figure 5. Base-class recall and precision under C-criterion (cf. Tables 2 and 3) for cross-training with support vector machines (ct-SVM) and for multi-labeled *k*-nearest neighbors (ML-*k*NN).

(given in parentheses in Tables 2 and 3) are considered. The cross-validated ct-SVM results are pretty close to those obtained with a single test set. The performance of the M_L-*k*NN method evaluated with ten times 10-fold cross-validation, at the same time, deteriorates more, in comparison to the M_L-*k*NN model evaluated on a single test set. Thus, the ct-SVM model is more robust, with regard to different training and test sets.

Figure 5 compares the performance of the ct-SVM and M_L-*k*NN methods, relative to the base classes under C-criterion (cf. Tables 2 and 3). With regard to recall (Figure 5a), the ct-SVM model performed better for all base classes, with the exception of HLT. The recall of a base class, as calculated according to eq 5, measures the fraction of STLs correctly predicted as belonging to the corresponding tribe (base class). That is, a recall of 1 will show that, indeed, all STLs isolated from a given tribe (including the multilabeled cases) were predicted as belonging to at least that tribe. If we consider, for example, the tribe Anthemideae (ANT), there were 30 STLs in the test set. The ct-SVM under C-criterion achieved a recall of 0.867 (cf. Table 2). This means that ~87% of the 30 STLs in the test set (26 STLs) obtained at least an ANT label (i.e., were classified (at least partially) correctly). Therefore, from the base class recall values presented in Tables 2 and 3 and depicted in Figure 5a, both multilabeled classification methods clearly performed well for almost all base classes (tribes), with the exception of the tribe Inuleae (INU), for which relatively low recall values were obtained. Both methods produced the same recall values for LAC, whereas the M_L-*k*NN method gave a slightly better recall for HLT. The ct-SVM method achieved higher recall for ANT, CAR, EUP, INU, and VER.

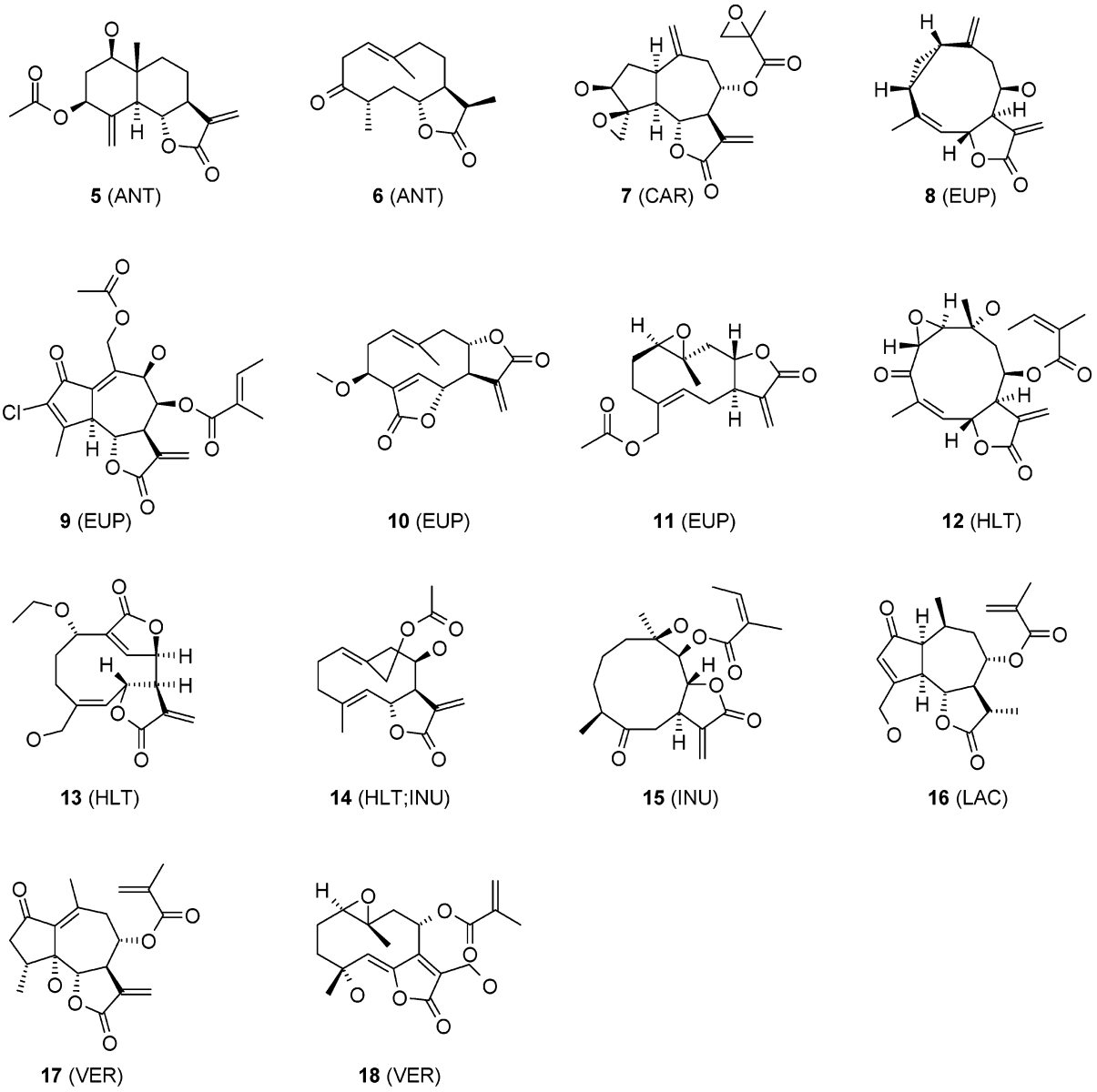
The base-class precision, calculated according to eq 6, on the other hand, shows the fraction of STLs predicted as belonging to a given tribe when they really belong to this tribe. That is, a perfectly performing classification method is expected to achieve a precision value of 1. Such precision will indicate that all STLs predicted as produced by a given tribe are actually found in at least this tribe. In other words, no false positive predictions are made. Let us consider again, as an example, the tribe Anthemideae (ANT). From all 142 STLs in the test set (cf. Table 1), 39 were predicted by ct-SVM as being synthesized at least by the plant species in ANT. From these 39 STLs, 26 were indeed isolated from plant species belonging to the tribe Anthemideae. By making the ratio (26/39), the resulting precision of 0.667 (cf. Table 2) is obtained. Examination of Figure 5b reveals that both the ct-SVM and ML-kNN methods produced similar base-class precisions (>0.6 in all cases). Therefore, both methods offer a moderate number of false positives and, thus, have good performance.

4.1.2. Based on the Label Rankings (Table 4). Examination of Table 4 confirms the good performance of both methods. Low one-error and coverage values were obtained, whereas the average precision was high. All metrics in this category favor the ct-SVM method, as can be observed in Table 4. This preference is once again enforced by the much lower loss of performance for the ct-SVM method when the cross-validated results are considered (observations similar to those presented previously in section 4.1.1 can be made). Combined with the better base-class recall, the aforementioned observations make the ct-SVM model the method of choice.

4.1.3. Comparison with Single-Labeled Classification (Table 5). It is worth noting that both multilabeled methods exhibit an accuracy that is similar to the accuracy of a single-labeled classification, as can be observed in Table 5. In the case of the ct-SVM method, the multilabeled model even outperformed the single-labeled *k*NN method. There are two different reasons for this result. First, the SVM method was not explored in our previous study,[14] and it may be more suited to this data set. The second possible reason is that the multilabeled ct-SVM classifier, based on cross-training, uses the multilabeled patterns more than once (cf. the Introduction). Subsequently, each of the individual binary classifiers is based on more data, compared to the single-labeled case, which may lead to better performance.

4.2. Comparison between the P-, T-, and C-Criterion (cf. Section 2.3.1.). Tables 2 and 3 shows that the use of the P-criterion produced the best performance metrics, compared to the T- and C-criteria. However, a noticeable aspect is that, under the P-criterion, both classification methods were not able to assign ~22% of the STLs to any of the seven tribes (see Tables 2 and 3).

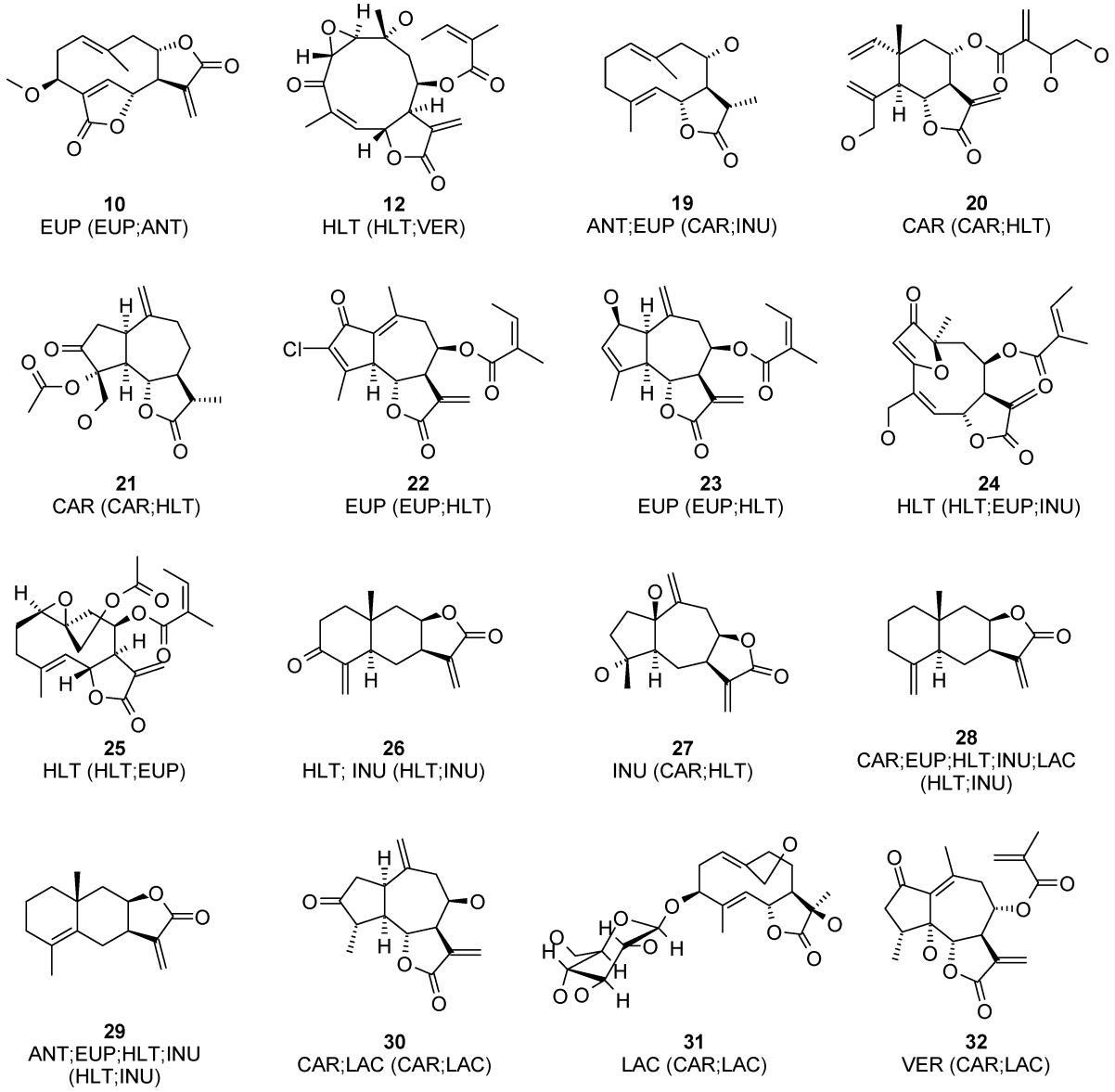
Chart 2. Structures of STLs from the Test Set That Both Classification Methods Did Not Assign to Any Tribe under P-Criterion, and the Corresponding Tribe(s) from Which They Have Been Isolated



Naturally, one would expect that there is a large intersection between the two sets of STLs, unassigned by the ct-SVM and M_L-*k*NN methods. However, there were 14 STLs unassigned to any of the tribes by both classification methods (see Chart 2), whereas the remainder (16 and 19 for the ct-SVM and Ml-kNN methods, respectively) were unique to the corresponding classification method. As expected, it indicates that, because of the singular nature of both classification methods, they have found somehow different decision boundaries between the classes. Because the multilabeled instances usually are set very close to the decision boundaries (cf. Figure 2), it was surprising that only 1 of the 14 rejected STLs was actually reported from more than one tribe (structure **14**, Chart 2). Nevertheless, each of the 14 structures is somewhat hard to classify into a single tribe. If skeletal types (or their subtypes) are analyzed alone, only two of the structures shown in Chart 2 (structures **8** and **13**) are considered as being unique to the tribe to which they belong. Based on the literature,[5] all of the remaining STLs from Chart 2 may appear in at least one more tribe. The exclusiveness of these structures in only one tribe, when it occurs, is solely attributed to certain peculiarities regarding specific substituents—or combinations of substituents—in their molecules, and we believe that such features were slightly difficult to capture using the P-criterion. It is the case, for example, of STLs **10**−**12** and **18**, which are typical for the tribes from which they originate,[5] but are not exclusive at all.

Another possible explanation for the inability to assign certain STLs to any particular tribe is that these unassigned STLs are outliers, because of specific skeletal or substitution patterns. However, such an assumption is not supported by the structures in Chart 2. In addition, we have already attempted to account for outliers in this data set. Using principal component analysis (PCA), in concert with the Hotteling T^2 test, it was shown that no improvement in the classification accuracy is obtained, even when a significant amount of the test data was discarded.[14] Consequently, this result is more likely caused by the fact that these STLs lie

Chart 3. STLs from the Test Set Classified as Belonging to More than One Asteraceae Tribe by the ct-SVM under the C-Criterion



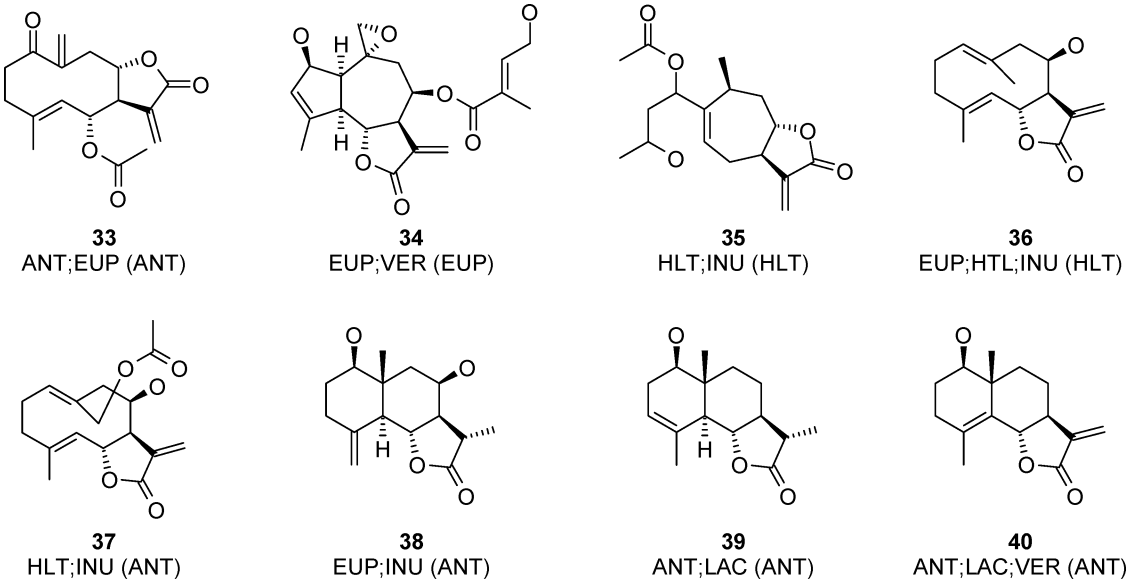
The tribe(s) from which the STLs have been reported is (are) given below the corresponding structures, and the tribes assigned by the classification are given in parentheses.

very close to the decision boundaries of the corresponding methods.

When examining the two remaining tested criteria (T- and C-criteria), the tendency of both classification methods (SVM and M_L-*k*NN) was to obtain slightly better results under the C-criterion. The difference appears marginal at a first glance (cf. Tables 2 and 3); nevertheless, from the data shown in Tables 2 and 3, it is difficult to evaluate how each classifier actually treated the multilabeled cases. Contrary to the expected behavior, the examination of the 13 multilabeled STLs in the test set reveals that the C-criterion does not improve their predictions but, instead, improves the overall performance. The reason is that some of the misclassified STLs under the T-criterion were assigned to more than one tribe under the C-criterion, thus making the prediction partially correct. By its nature, the C-criterion produces a higher number of multilabeled instances. Hence, while under the T-criterion, the ct-SVM method assigned multiple classes to 9 STLs, whereas under the C-criterion, this number was 16. The corresponding numbers for the M_L-*k*NN method were 7 and 17, respectively.

The purpose of our models is to assist in a targeted collection of plants. With this objective in mind, although the P-criterion produced the best performance metrics (cf. Tables 2 and 3), its inability to assign ~22% of the STLs to any of the considered tribes limits its use. Both T- and C-criteria are preferable. When choosing between T-criterion and C-criterion, one must consider two costs: (1) completely missing the correct tribe and (2) predicting more than one possible tribe for an STL, which has been reported from only one tribe. The C-criterion is the model of choice, when compared to the T-criterion, when the second cost is lower. With the objective of assisting in a targeted collection of plants, the second cost is lower, especially when the "wrong" tribes bear close relationships to the correct tribe. This is the case because only a small part of the Asteraceae plants have been studied so far, and it is likely that some of the STLs from our dataset will be found in additional tribes in the future. Thus, based on the discussion so far, we have identified the ct-SVM model in concert with the C-criterion

Chart 4. STLs from the Test Set Reported in More than One Asteraceae Tribe, and Their Subsequent Predictions by ct-SVM Using C-Criterion



The tribes from which the STL have been reported are below the structures, and the predicted tribe is given in parentheses. Only the eight multilabeled STLs not shown in Chart 3 are shown.

to be the best method for helping in the targeted collection of plants. In the following section, we present an analysis of the performance of the proposed method, ct-SVM, with the C-criterion, with regard to the chemical structures of the STLs, reported from more than one tribe.

4.3. Analyses of the ct-SVM Results under C-Criterion. Although the current taxonomic classification[27] of Asteraceae used to label the STLs in this work is mostly based on morphologic aspects, we obtained good classification models. Hence, taking into account only the C-criterion, the distribution of the STLs among the tribes clearly differs by a sufficient amount to allow a targeted plant collection. This observation is supported by the good base-class performance that is achieved using the ct-SVM method. However, another aspect to be considered is that, although all plant sources were checked for each single STL, there is a probability that, in the future, some of the currently single-labeled STLs may appear in more than one tribe. Because the main advantage of the proposed methodology is its ability to assign an STL to multiple tribes, we examined the STLs that were predicted to belong to more than one tribe under the C-criterion, using the ct-SVM method. Chart 3 gives the 16 STLs that were predicted to belong to more than one tribe.

First, we examine the case when a structure was reported from a single tribe—structures **10**, **12**, **20**−**25**, **27**, **31**, and **32** in Chart 3—and predicted to belong to two or more tribes. In most cases, the additional "new" tribe (or tribes) is meaningful, from a chemotaxonomic point of view. The "new" tribes usually possess closely related STLs. Therefore, the obtained results indicate that the chemical boundaries among such tribes are blurred. Consider, for example, STLs **22** and **23** from Chart 3, which actually belong to EUP. The predictions indicate that they belong to EUP as well as HLT. In this example, HLT (the "new" tribe) should not be considered as a completely wrong prediction for **22** and **23**, because, according to the structures in the dataset, closely related STLs also belong to HLT. Data from the literature also support this statement, because EUP and HLT have a strong chemical association to each other.[5−7] It should be emphasized that structures **10** and **12** appeared in both Chart 2 and Chart 3, and their exclusiveness in only one tribe has already been discussed previously, thus corroborating the current results under the C-criterion. The aforementioned explanation also is valid for the remaining STLs in Chart 3. The predicted tribes were completely wrong only for structures **27** and **32**.

The second interesting case is when a structure is reported from two or more tribes (STLs **19**, **26**, **28**−**30**) (i.e., they are multilabeled). In these examples, all tribe assignments were quite good. Totally correct assignments were achieved for structures **26** and **30**. Although STLs **28** and **29** have been assigned to only two tribes (out of five and four tribes, respectively) all of the assigned tribes were correct. The only completely wrongly assigned STL was **19**.

As shown in Chart 3, 5 of the 13 multilabeled STLs in the test set obtained more than one label from the classification (ct-SVM, C-criterion). Chart 4 shows the remaining 8 originally multilabeled STLs.

With the exception of STLs **37** and **38**, Chart 4 shows that all the multilabeled STLs were predicted partially correct. Interestingly, note that, if one excludes from this analysis STL **35**, whose skeleton is more restricted to HLT or INU, the skeletons of all the remaining STLs can actually be found in all 7 tribes, because they are of a widely occurring nature. For this reason, it was unexpected that only one tribe was assigned to each of these structures. The obtained results demonstrate that the classification in this case was based more on the substitutional features rather than skeletal types of the STLs. In regard to the focused collection of plants, such results, although missing possible plant sources, are useful when the substitutional features of the desired STL are of high importance.

An examination of the actual scores produced by the SVM classifier showed that, in most cases, the "true" labels obtained the highest scores, although the difference among them was higher than the used threshold.

5. CONCLUSION

We have presented a general overview of multilabeled classification, which is a machine learning technique that allows an object to be simultaneously classified into several classes. This technique has applications in various domains: text categorization, selectivity of a biologically active compounds, gene functions analysis, etc. Two multilabeled classification methods (cross-training with support vector machine as a classifier (ct-SVM) and multilabeled *k*-nearest neighbor (M_L-*k*NN)) have been successfully applied to the assignment of a special type of secondary metabolites—sesquiterpene lactones (STLs)—into the Asteraceae tribe(s) from which they were isolated. The utility of the proposed classification model for a targeted collection of plant material, with the objective of finding a particular natural compound, was shown. The SVM model yielded better results, outperforming both the M_L-*k*NN and the previously built single-labeled *k*-nearest neighbor (*k*-NN) classifier. With regard to the STLs that have been isolated from more than one tribe (i.e., the multilabeled STLs), both multilabeled methods (ct-SVM and M_L-*k*NN) performed reasonably and were able to assign such STLs at least partially correctly. Taking into account the three different testing criteria used to convert the real-valued classifier output to labels, the criterion of choice was the C-criterion. The C-criterion showed the best probability to label the STLs from the test set correctly. The simultaneous assignment of STLs to multiple Asteraceae tribes was exemplified and discussed. The handling of STLs, which appear in more than one tribe, was also shown and discussed. Both analyses demonstrated the value of the proposed methodology (1) to study the relationships between the secondary metabolism of the plant family Asteraceae and its current taxonomic classification and (2) to assist in the targeted collection of plant material, with the objective of isolating particular STLs.

ACKNOWLEDGMENT

F.B.C. is grateful to the Alexander von Humboldt-Foundation (Germany) for a Research Fellowship at the Computer-Chemie-Centrum.

REFERENCES AND NOTES

(1) Mann, J. *Chemical Aspects of Biosynthesis*; Oxford University Press: Oxford, U.K., 1994.
(2) Cordell, G. Natural Products in Drug Discovery—Creating a New Vision. *Phytochem. Rev.* **2002**, *1*, 261−273.
(3) Abel, U.; Koch, C.; Speitling, M.; Hansske, F. G. Modern Methods to Produce Natural-Product Libraries. *Curr. Opin. Chem. Biol.* **2002**, *6*, 453−458.
(4) Hostettmann, K.; Wolfender, J. L. The Search for Biologically Active Secondary Metabolites. *Pestic. Sci.* **1997**, *51*, 471−482.
(5) Seaman, F. C. Sesquiterpene Lactones As Taxonomic Characters in the Asteraceae. *Bot. Rev.* **1982**, *48*, 123−551.
(6) Alvarenga, S. A. V.; Ferreira, M. J. P.; Emerenciano, V. P.; Cabrol-Bass, D. Chemosystematic Studies of Natural Compounds Isolated From Asteraceae: Characterization of Tribes by Principal Component Analysis. *Chemometr. Intell. Lab.* **2001**, *56*, 27−37.
(7) Zdero, C.; Bohlmann, F. Systematics and Evolution Within the Composite, Seen With the Eyes of a Chemist. *Plant Syst. Evol.* **1990**, *171*, 1−14.
(8) Herlocker, J. L. Understanding and Improving Automated Collaborative Filtering Systems. Ph.D. Dissertation, University of Minnesota, Minneapolis, MN, 2000.
(9) Erhan, D.; LrHeureux, P. J.; Yue, S. Y.; Bengio, Y. Collaborative Filtering on a Family of Biological Targets. *J. Chem. Inf. Model.* **2006**, *46*, 626−635.
(10) Witten, I. H.; Eibe, F. *Data Mining: Practical Machine Learning Tools and Techniques*; Morgan Kaufmann: San Francisco, CA, 2000.
(11) McCallum, A. Multi-label text classification with a mixture model trained by EM. In *Proceedings of AAAI'99 Workshop on Text Learning*, 1999.
(12) Schapire, R. E.; Singer, Y. BoosTexter: A Boosting-Based System for Text Categorization. *Mach. Learn.* **2000**, *39*, 135−168.
(13) Clare, A.; King, R. D. Knowledge Discovery in Multi-Label Phenotype Data. In *Lecture Notes in Computer Science*; Raedt, L. D., Siebes, A., Eds.; Springer: Berlin, Germany, 2001; Vol. 2168, pp 42−53.
(14) Hristozov, D.; DaCosta, F. B.; Gasteiger, J. Sesquiterpene Lactones-Based Classification of the Family Asteraceae Using Neural Networks and *K*-Nearest Neighbors. *J. Chem. Inf. Model.* **2007**, *47*, 9−19.
(15) Boutell, M. R.; Luo, J.; Shen, X.; Brown, C. M. C. Learning Multi-Label Scene Classification. *Pattern. Recogn.* **2004**, *37*, 1757−1771.
(16) Schlokopf, B.; Smola, A. J. *Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond*, MIT Press: Cambridge, MA, 2001.
(17) Spycher, S.; Nendza, M.; Gasteiger, J. Comparison of Different Classification Methods Applied to a Mode of Toxic Action Data Set. *QSAR Combinat. Sci.* **2004**, *23*, 779−791.
(18) Freund, Y.; Schapire, R. E. A Decision-Theoretic Generalization of on-Line Learning and an Application to Boosting. *J. Comput. Syst. Sci.* **1997**, *55*, 119−139.
(19) Elisseeff, A.; Weston, J. A Kernel Method for Multi-Labelled Classification. In *Advances in Neural Information Processing Systems*; Dietterich, T. G., Becker, S., Ghahramani, Z., Eds.; MIT Press: Cambridge, MA, 2002; Vol. 14.
(20) Quinlan, J. R. *C4.5: Programs for Machine Learning*; Morgan Kauffman: San Mateo, CA, 1993.
(21) Ueda, N.; Saito, K. Parametric Mixture Models for Multi-Label Text. In *Advances in Neural Information Processing Systems*; Becker, S., Thrun, S., Obermayer, K., Eds.; MIT Press: Cambridge, MA, 2003; Vol. 15.
(22) Freund, Y.; Mason, L. The alternating decision tree learning algorithm. In *Proceedings of 16th International Conference on Machine Learning*; Morgan Kaufmann: San Francisco, CA, 1999; pp 124−133.
(23) De Comite, F.; Gilleron, R.; Tommasi, M. Learning Multi-Label Alternating Decision Trees From Texts and Data. In *Lecture Notes in Computer Science*; Perner, P., Rosenfeld, A., Eds.; Springer: Berlin, Germany, 2003; Vol. 2734, pp 35−49.
(24) Zhang, M.-L.; Zhou, Z.-H. A *k*-Nearest Neighbor Based Algorithm for Multi-label Classification. In *2005 IEEE International Conference on Granular Computing*, 2005; Vol. 2, pp 718−721.
(25) Gower, J. C.; Legendre, P. Metric and Euclidean Properties of Dissimilarity Coefficients. *J. Classif.* **1986**, *3*, 5−48.
(26) Salton, G. Developments in Automatic Text Retrieval. *Science* **1991**, *253*, 974−980.
(27) Bremer, K. Asteraceae: *Cladistics and Classification*; Timber Press: Portland, OR, 1994.
(28) Hemmer, M. C.; Steinhauer, V.; Gasteiger, J. Deriving the 3D Structure of Organic Molecules From Their Infrared Spectra. *Vib. Spectrosc.* **1999**, *19*, 151−164.
(29) Sadowski, J.; Gasteiger, J. From Atoms and Bonds to Three-Dimensional Atomic Coordinates: Automatic Model Builders. *Chem. Rev.* **1993**, *93*, 2567−2581.
(30) CORINA, version 3.2, Molecular Networks GmbH, Erlangen, Germany, http://www.molecular-networks.com (accessed June 2006).
(31) ADRIANA.Code, version 1.0, Molecular Networks GmbH, Erlangen, Germany, http://www.molecular-networks.com (accessed June 2006).
(32) Vapnik, V. N. Statistical Learning Theory; Wiley−Interscience: New York, 1998.
(33) Hsu, C.-W.; Chang, C.-C.; Lin, C.-J. *A Practical Guide to Support Vector Classification*; 2006. URL: http://www.csie.ntu.edu.tw/~cjlin/papers/guide/guide.pdf (accessed June 2006).
(34) R Development Core Team. R: A language and environment for statistical computing, Version 2.2.1, 2005. URL: http://www.r-project.org (accessed June 2006).
(35) Dimitriadou, E.; Hornik, K.; Leisch, F.; Meyer, D.; Weingessel, A. e1071: Misc functions of the department of statistics (e1071), TU Wien, 2005.
(36) Hastie, T.; Tibshirani, R.; Friedman, J. *The Elements of Statistical Learning*; Springer: New York, 2001.

CI700175M