Development of the Algorithm

The development process of the screening tool is schematically described in Figure S1. Initially, our team (with substantial input from SFQ and GES) empirically selected 22 candidate variables based on their known association with SDB from the 1280 variables as features. The 22 candidate variables included: the SaO_2% during sleep, age at time of study in years, based on start date of SHHS1 PSG recording; Gender as reported by Parent Cohort; Parent Cohort reported Diabetes Status; Neck circumference in centimeters. Questions included: How often do you snore? What is chance that you would doze off or fall asleep while in a car, while stopped for a few minutes in traffic? What is the chance that you would doze off or fall asleep while sitting inactive in a public place? What is the chance that you would doze off or fall asleep while sitting and talking to someone?

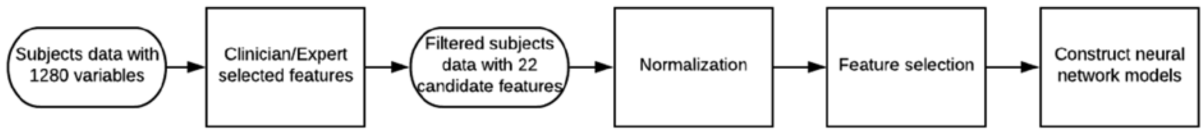

Figure S1—Development Process

Normalization

We used the Scikit-learn machine learning library to develop the screening tool.[17] The min-max normalization strategy was applied to normalize the 22 candidate features to the range [0, 1]. The following equations further clarify the normalization process:

$$X_{std} = \frac{X - \min(X)}{\max(X) - \min(X)} \qquad (2)$$

$$X_{scaled} = X_{std} * (\max - \min) + \min \qquad (3)$$

Here in (2), X is a 2D array storing all candidate features. The max(X) and min(X) are two 1D arrays with maximum values and minimum values of the features in the full dataset. In (3), the (max, min) is the normalized range of the candidate features. In this study, max is 1, min is 0.

Feature Selection

We employed the extremely randomized trees algorithm with model selection as the feature selection algorithm.[1,2] The extremely randomized trees algorithm is a tree-based ensemble method to build 10 total randomized trees.[2] The importance weights of each feature were computed by the feature selection algorithm. Features used at top of the trees have higher important weight.[1] The total weight is 1 of the 22 candidate features. The input features of each MLP neural network model were selected from the 22 candidate features based on their importance weights. The AHI threshold ≥ 5/hour had 7 input features. The AHI thresholds ≥ 10/hour, 15/hour, and 20/hour models had 8 input features each. The AHI threshold ≥ 25/hour model had 11 input features. The AHI threshold ≥ 30/hour model had 9 input features.

Construction of Neural Network Models

The MLP neural network had two layers: one hidden layer, and an output layer. The MLP neural network models were trained by the backpropagation learning method in conjunction with the well-known limited memory BFGS optimization algorithm (L-BFGS) with L2 regularization. The L-BFGS is a quasi-Newton method algorithm with limited computer memory.[3] During the optimization process, the parameters in neural network models were random initiated. The parameters of the model were optimized to reduce the output error. The output error was computed by the cross-entropy cost function. This process was repeated for all subjects in the training set over several iterations. After sufficient training, the model learned how to accurately compute the output result. In this study, we used two types of activation functions for the hidden layer of different neural network models: the logistic function (Logistic, in (4)) and the hyperbolic tangent function (Tanh, in (5)). The activation function of output layer is the logistic function. We applied the grid search method to optimize the hyper-parameters of the MLP neural network models and used the AUC as the metric. In the optimization procedure, we used a 10-fold cross-validation to evaluate the AUC.[4] The optimized hyper-parameters, activation function, and input features for each model are listed in Table S1.

$$f_{logistic}(x) = \frac{1}{1+e^{-x}} \quad (4)$$

$$f_{tanh}(x) = \frac{e^{x}-e^{-x}}{e^{x}+e^{-x}} \quad (5)$$

Table S1—Hyper-parameters and Selected Features of Each Model

AHI ≥	Activation function of hidden layer	Hidden neurons	L2 regularization term	Features
5	Logistic	4	10^{-4}	Age, BMI, MinO_2Sa, NC, O_2Sa90, O_2Sa95, Snore
10	Tanh	3	10^{-2}	Age, BMI, MinO_2Sa, NC, O_2Sa85, O_2Sa90, O_2Sa95, Snore
15	Logistic	6	10^{-3}	Age, BMI, MinO_2Sa, NC, O_2Sa85, O_2Sa90, O_2Sa95, Snore
20	Logistic	4	10^{-2}	Age, BMI, MinO_2Sa, NC, O_2Sa85, O_2Sa90, O_2Sa95, Snore
25	Tanh	10	10^{-3}	Age, BMI, MinO_2Sa, NC, O_2Sa75, O_2Sa80, O_2Sa85, O_2Sa90, O_2Sa95, Snore, SInPub
30	Tanh	7	10^{-3}	Age, BMI, MinO_2Sa, NC, O_2Sa80, O_2Sa85, O_2Sa90, O_2Sa95, Snore

BMI: Body Mass Index; MinO_2Sa: Minimal oxygen saturation in sleep; NC: Neck circumference; O_2Sa90: Percent of sleep time oxygen saturation below 90%; O_2Sa95: Percent of sleep time oxygen saturation below 95%; Snore: Frequency of Snoring; O_2Sa85: Percent of sleep time oxygen saturation below 85%; O_2Sa75: Percent of sleep time oxygen saturation below 75%; O_2Sa80: Percent of sleep time oxygen saturation below 80%; SInPub: Fall asleep while sitting inactive in a public place

References

1. Pedregosa F, Varoquaux G, Gramfort A, et al. Scikit-learn: machine learning in Python. J Mach Learn Res. 2011:2825-2830.
2. Geurts P, Ernst D, Wehenkel L. Extremely randomized trees. Mach Learn. 2006;63:3-42.
3. Liu DC, Nocedal J. On the limited memory BFGS method for large scale optimization. Math Program. 1989;45:503-528.
4. Mosteller F, Tukey JW. Data analysis, including statistics. In: Lindzey G, Aronson E, eds. Handbook of Social Psychology, Vol. 2. Addison–Wesley; 1968.