Algorithm Description: Auto Feature Selection and Model Training Pipeline

Algorithm 1 Auto Feature Selection and Model Training Pipeline

Class `EnsembleForRFE(BaseEstimator)`
  Input: svm_C, tree_max_depth, tree_min_samples_split, gbm_learning_rate, gbm_n_estimators
  Operations:
    Initialize SVM, Decision Tree, and Gradient Boosting with specified parameters
    Define methods: fit(X, y), predict(X), set_params(**params)
  End Class

Function `setup_feature_selection()`
  Operations:
    Create an instance of EnsembleForRFE
    Initialize RFECV with EnsembleForRFE and StratifiedKFold
    Initialize SelectKBest with mutual_info_classif
    Combine RFECV and SelectKBest using FeatureUnion
  Return: feature_selection

Function `train_model(X, Y, feature_selection, parameters, n_iter, n_cv, n_jobs)`
  Operations:
    Configure a pipeline with StandardScaler, feature_selection, and StackingClassifier
    Define RandomizedSearchCV with the pipeline and specified parameters
  Return: clf (trained model)

Function `auto_feature_selection(data_file, label_file, label_col, threshold, show_plot, show_progress, n_iter, n_cv, n_jobs, save_path, sleep_interval, use_tkagg)`
  Operations:
    Load and preprocess data from data_file and label_file
    Call setup_feature_selection() to configure feature selection
    Define parameter distribution for RandomizedSearchCV
    Train the model using train_model()
    if show_progress **then**
      Initialize and manage a progress bar
      Run RandomizedSearchCV in a separate thread and monitor progress
    else
      Directly execute RandomizedSearchCV
    Extract and save results, display plots if required
  End Function

This document illustrates a complete pipeline from data loading, preprocessing, feature selection setup, model training with hyperparameter tuning, to result extraction and evaluation. The progression from data setup to final results encapsulates the complexity and interdependencies of the involved components.