NAME

TopologicalAtomPairsFingerprints

SYNOPSIS

use Fingerprints::TopologicalAtomPairsFingerprints;

use Fingerprints::TopologicalAtomPairsFingerprints qw(:all);

DESCRIPTION

TopologicalAtomPairsFingerprints [Ref 57, Ref 59, Ref 72] class provides the following methods:

new, GenerateFingerprints, GetAtomPairIDs, GetDescription, SetAtomIdentifierType, SetAtomicInvariantsToUse, SetFunctionalClassesToUse, SetMaxDistance, SetMinDistance, StringifyTopologicalAtomPairsFingerprints

TopologicalAtomPairsFingerprints is derived from Fingerprints class which in turn is derived from ObjectProperty base class that provides methods not explicitly defined in TopologicalAtomPairsFingerprints, Fingerprints or ObjectProperty classes using Perl's AUTOLOAD functionality. These methods are generated on-the-fly for a specified object property:

```
Set<PropertyName>(<PropertyValue>);
$PropertyValue = Get<PropertyName>();
Delete<PropertyName>();
```

The current release of MayaChemTools supports generation of AtomTypesFingerpritns corresponding to following AtomtomIdentifierTypes:

```
AtomicInvariantsAtomTypes, DREIDINGAtomTypes, EStateAtomTypes,
FunctionalClassAtomTypes, MMFF94AtomTypes, SLogPAtomTypes,
SYBYLAtomTypes, TPSAAtomTypes, UFFAtomTypes
```

Based on the values specified for AtomIdentifierType along with other specified parameters such as AtomicInvariantsToUse and FunctionalClassesToUse, initial atom types are assigned to all non-hydrogen atoms in a molecule. Using the distance matrix for the molecule and initial atom types assigned to non-hydrogen atoms, all unique atom pairs within MinDistance and MaxDistance are identified and counted. An atom pair identifier is generated for each unique atom pair; the format of atom pair identifier is:

```
<AtomType1>-D<n>-<AtomType2>


AtomType1, AtomType2: Atom types assigned to atom1 and atom2
D: Distance between atom1 and atom2


where AtomType1 <= AtomType2
```

The atom pair identifiers for all unique atom pairs corresponding to non-hydrogen atoms constitute topological atom pairs fingerprints of the molecule.

The current release of MayaChemTools generates the following types of topological atom pairs fingerprints vector strings:

```
FingerprintsVector;TopologicalAtomPairs:AtomicInvariantsAtomTypes:MinD
istance1:MaxDistance10;223;NumericalValues;IDsAndValuesString;C.X1.BO1
.H3-D1-C.X3.BO3.H1 C.X2.BO2.H2-D1-C.X2.BO2.H2 C.X2.BO2.H2-D1-C.X3.BO3.
H1 C.X2.BO2.H2-D1-C.X3.BO4 C.X2.BO2.H2-D1-N.X3.BO3 C.X2.BO3.H1-D1-...;
2 1 4 1 1 10 8 1 2 6 1 2 2 1 2 1 2 2 1 2 1 5 1 10 12 2 2 1 2 1 9 1 3 1
1 1 2 2 1 3 6 1 6 14 2 2 2 3 1 3 1 8 2 2 1 3 2 6 1 2 2 5 1 3 1 23 1...


FingerprintsVector;TopologicalAtomPairs:AtomicInvariantsAtomTypes:MinD
istance1:MaxDistance10;223;NumericalValues;IDsAndValuesPairsString;C.X
1.BO1.H3-D1-C.X3.BO3.H1 2 C.X2.BO2.H2-D1-C.X2.BO2.H2 1 C.X2.BO2.H2-D1-
C.X3.BO3.H1 4 C.X2.BO2.H2-D1-C.X3.BO4 1 C.X2.BO2.H2-D1-N.X3.BO3 1 C.X2
.BO3.H1-D1-C.X2.BO3.H1 10 C.X2.BO3.H1-D1-C.X3.BO4 8 C.X3.BO3.H1-D1-C.X
3.BO4 1 C.X3.BO3.H1-D1-O.X1.BO1.H1 2 C.X3.BO4-D1-C.X3.BO4 6 C.X3.BO...


FingerprintsVector;TopologicalAtomPairs:DREIDINGAtomTypes:MinDistance1
:MaxDistance10;157;NumericalValues;IDsAndValuesString;C_2-D1-C_3 C_2-D
1-C_R C_2-D1-N_3 C_2-D1-O_2 C_2-D1-O_3 C_3-D1-C_3 C_3-D1-C_R C_3-D1-N_
R C_3-D1-O_3 C_R-D1-C_R C_R-D1-F_ C_R-D1-N_3 C_R-D1-N_R C_2-D2-C_3 C_2
1 1 1 2 1 7 1 1 2 23 1 1 2 1 3 5 5 2 1 5 28 2 3 3 1 1 1 2 4 1 1 4 9 3
1 4 24 2 4 3 3 4 5 5 14 1 1 2 3 22 1 3 4 4 1 1 1 1 2 2 5 1 4 21 3 1...


FingerprintsVector;TopologicalAtomPairs:EStateAtomTypes:MinDistance1:M
axDistance10;251;NumericalValues;IDsAndValuesString;aaCH-D1-aaCH aaCH-
D1-aasC aasC-D1-aasC aasC-D1-aasN aasC-D1-dssC aasC-D1-sF aasC-D1-ssNH
aasC-D1-sssCH aasN-D1-ssCH2 dO-D1-dssC dssC-D1-sOH dssC-D1-ssCH2 d...;
10 8 5 2 1 1 1 1 1 2 1 1 1 2 2 1 4 10 12 2 2 6 3 1 3 2 2 1 1 1 1 1 1 1
1 1 5 2 1 1 6 12 2 2 2 2 6 1 3 2 2 5 2 2 1 2 1 1 1 1 1 1 3 1 3 19 2...
```

```
FingerprintsVector;TopologicalAtomPairs:FunctionalClassAtomTypes:MinDi
stance1:MaxDistance10;144;NumericalValues;IDsAndValuesString;Ar-D1-Ar
Ar-D1-Ar.HBA Ar-D1-HBD Ar-D1-Hal Ar-D1-None Ar.HBA-D1-None HBA-D1-NI H
BA-D1-None HBA.HBD-D1-NI HBA.HBD-D1-None HBD-D1-None NI-D1-None No...;
23 2 1 1 2 1 1 1 1 2 1 1 7 28 3 1 3 2 8 2 1 1 1 5 1 5 24 3 3 4 2 13 4
1 1 4 1 5 22 4 4 3 1 19 1 1 1 1 1 2 2 3 1 1 8 25 4 5 2 3 1 26 1 4 1 ...

FingerprintsVector;TopologicalAtomPairs:MMFF94AtomTypes:MinDistance1:M
axDistance10;227;NumericalValues;IDsAndValuesPairsString;C5A-D1-C5B 2
C5A-D1-CB 1 C5A-D1-CR 1 C5A-D1-N5 2 C5B-D1-C5B 1 C5B-D1-C=ON 1 C5B-D1-
CB 1 C=ON-D1-NC=O 1 C=ON-D1-O=CN 1 CB-D1-CB 18 CB-D1-F 1 CB-D1-NC=O 1
COO-D1-CR 1 COO-D1-O=CO 1 COO-D1-OC=O 1 CR-D1-CR 7 CR-D1-N5 1 CR-D1-OR
2 C5A-D2-C5A 1 C5A-D2-C5B 2 C5A-D2-C=ON 1 C5A-D2-CB 3 C5A-D2-CR 4 ...

FingerprintsVector;TopologicalAtomPairs:SLogPAtomTypes:MinDistance1:Ma
xDistance10;329;NumericalValues;IDsAndValuesPairsString;C1-D1-C10 1 C1
-D1-C11 2 C1-D1-C5 1 C1-D1-CS 4 C10-D1-N11 1 C11-D1-C21 1 C14-D1-C18 2
C14-D1-F 1 C18-D1-C18 10 C18-D1-C20 4 C18-D1-C22 2 C20-D1-C20 3 C20-D
1-C21 1 C20-D1-N11 1 C21-D1-C21 1 C21-D1-C5 1 C21-D1-N11 1 C22-D1-N4 1
C5-D1-N4 1 C5-D1-O10 1 C5-D1-O2 1 C5-D1-O9 1 CS-D1-O2 2 C1-D2-C1 3...

FingerprintsVector;TopologicalAtomPairs:SYBYLAtomTypes:MinDistance1:Ma
xDistance10;159;NumericalValues;IDsAndValuesPairsString;C.2-D1-C.3 1 C
.2-D1-C.ar 1 C.2-D1-N.am 1 C.2-D1-O.2 1 C.2-D1-O.co2 2 C.3-D1-C.3 7 C.
3-D1-C.ar 1 C.3-D1-N.ar 1 C.3-D1-O.3 2 C.ar-D1-C.ar 23 C.ar-D1-F 1 C.a
r-D1-N.am 1 C.ar-D1-N.ar 2 C.2-D2-C.3 1 C.2-D2-C.ar 3 C.3-D2-C.3 5 C.3
-D2-C.ar 5 C.3-D2-N.ar 2 C.3-D2-O.3 4 C.3-D2-O.co2 2 C.ar-D2-C.ar 2...

FingerprintsVector;TopologicalAtomPairs:TPSAAtomTypes:MinDistance1:Max
Distance10;64;NumericalValues;IDsAndValuesPairsString;N21-D1-None 3 N7
-D1-None 2 None-D1-None 34 None-D1-O3 2 None-D1-O4 3 N21-D2-None 5 N7-
D2-None 3 N7-D2-O3 1 None-D2-None 44 None-D2-O3 2 None-D2-O4 5 O3-D2-O
4 1 N21-D3-None 7 N7-D3-None 4 None-D3-None 45 None-D3-O3 4 None-D3-O4
5 N21-D4-N7 1 N21-D4-None 5 N21-D4-O3 1 N21-D4-O4 1 N7-D4-None 4 N...

FingerprintsVector;TopologicalAtomPairs:UFFAtomTypes:MinDistance1:MaxD
istance10;157;NumericalValues;IDsAndValuesPairsString;C_2-D1-C_3 1 C_2
-D1-C_R 1 C_2-D1-N_3 1 C_2-D1-O_2 2 C_2-D1-O_3 1 C_3-D1-C_3 7 C_3-D1-C
_R 1 C_3-D1-N_R 1 C_3-D1-O_3 2 C_R-D1-C_R 23 C_R-D1-F_ 1 C_R-D1-N_3 1
C_R-D1-N_R 2 C_2-D2-C_3 1 C_2-D2-C_R 3 C_3-D2-C_3 5 C_3-D2-C_R 5 C_3-D
2-N_R 2 C_3-D2-O_2 1 C_3-D2-O_3 5 C_R-D2-C_R 28 C_R-D2-F_ 2 C_R-D2-...
```

METHODS

new

```
$NewTopologicalAtomPairsFingerprints = new TopologicalAtomPairsFingerprints(
                                                %NamesAndValues);
```

Using specified *TopologicalAtomPairsFingerprints* property names and values hash, new method creates a new object and returns a reference to newly created TopologicalAtomPairsFingerprints object. By default, the following properties are initialized:

```
Molecule = ''
Type = 'TopologicalAtomPairs'
MinDistance = 1
MaxDistance = 10
AtomIdentifierType = ''
AtomicInvariantsToUse = ['AS', 'X', 'BO', 'H', 'FC']
FunctionalClassesToUse = ['HBD', 'HBA', 'PI', 'NI', 'Ar', 'Hal']
```

Examples:

```
$TopologicalAtomPairsFingerprints = new TopologicalAtomPairsFingerprints(
                            'Molecule' => $Molecule,
                            'AtomIdentifierType' =>
                                        'AtomicInvariantsAtomTypes');

$TopologicalAtomPairsFingerprints = new TopologicalAtomPairsFingerprints(
                            'Molecule' => $Molecule,
                            'MinDistance' => 1,
                            'MaxDistance' => 10,
                            'AtomIdentifierType' =>
                                        'AtomicInvariantsAtomTypes',
                            'AtomicInvariantsToUse' =>
                                        ['AS', 'X', 'BO', 'H', 'FC'] );

$TopologicalAtomPairsFingerprints = new TopologicalAtomPairsFingerprints(
```

```
                                'Molecule' => $Molecule,
                                'AtomIdentifierType' =>
                                                'EStateAtomTypes');

    $TopologicalAtomPairsFingerprints = new TopologicalAtomPairsFingerprints(
                                'Molecule' => $Molecule,
                                'AtomIdentifierType' =>
                                                'SLogPAtomTypes');

    $TopologicalAtomPairsFingerprints = new TopologicalAtomPairsFingerprints(
                                'Molecule' => $Molecule,
                                'MinDistance' => 1,
                                'MaxDistance' => 10,
                                'AtomIdentifierType' =>
                                                'FunctionalClassAtomTypes',
                                'FunctionalClassesToUse' =>
                                                ['HBD', 'HBA', 'PI', 'NI', 'Ar', 'Hal']);

    $TopologicalAtomPairsFingerprints->GenerateFingerprints();
    print "$TopologicalAtomPairsFingerprints\n";
```

GetDescription

```
    $Description = $TopologicalAtomPairsFingerprints->GetDescription();
```

Returns a string containing description of topological atom pairs fingerprints fingerprints.

GenerateFingerprints

```
    $TopologicalAtomPairsFingerprints->GenerateFingerprints();
```

Generates topological atom pairs fingerprints and returns *TopologicalAtomPairsFingerprints*.

GetAtomPairIDs

```
    $AtomPairIDsRef = $TopologicalAtomPairsFingerprints->GetAtomPairIDs();
    @AtomPairIDs = $TopologicalAtomPairsFingerprints->GetAtomPairIDs();
```

Returns atom pair IDs corresponding to atom pairs count values in topological atom pairs fingerprints vector as an array or reference to an array.

SetAtomIdentifierType

```
    $TopologicalAtomPairsFingerprints->SetAtomIdentifierType($IdentifierType);
```

Sets atom *IdentifierType* to use during atom pairs fingerprints generation and returns *TopologicalAtomPairsFingerprints*.

Possible values: *AtomicInvariantsAtomTypes, DREIDINGAtomTypes, EStateAtomTypes, FunctionalClassAtomTypes, MMFF94AtomTypes, SLogPAtomTypes, SYBYLAtomTypes, TPSAAtomTypes, UFFAtomTypes*.

SetAtomicInvariantsToUse

```
    $TopologicalAtomPairsFingerprints->SetAtomicInvariantsToUse($ValuesRef);
    $TopologicalAtomPairsFingerprints->SetAtomicInvariantsToUse(@Values);
```

Sets atomic invariants to use during *AtomicInvariantsAtomTypes* value of *AtomIdentifierType* for topological atom pairs fingerprints generation and returns *TopologicalAtomPairsFingerprints*.

Possible values for atomic invariants are: *AS, X, BO, LBO, SB, DB, TB, H, Ar, RA, FC, MN, SM*. Default value: *AS,X,BO,H,FC*.

The atomic invariants abbreviations correspond to:

```
    AS = Atom symbol corresponding to element symbol

    X<n>   = Number of non-hydrogen atom neighbors or heavy atoms
    BO<n> = Sum of bond orders to non-hydrogen atom neighbors or heavy atoms
    LBO<n> = Largest bond order of non-hydrogen atom neighbors or heavy atoms
    SB<n> = Number of single bonds to non-hydrogen atom neighbors or heavy atoms
    DB<n> = Number of double bonds to non-hydrogen atom neighbors or heavy atoms
    TB<n> = Number of triple bonds to non-hydrogen atom neighbors or heavy atoms
    H<n>   = Number of implicit and explicit hydrogens for atom
    Ar     = Aromatic annotation indicating whether atom is aromatic
    RA     = Ring atom annotation indicating whether atom is a ring
    FC<+n/-n> = Formal charge assigned to atom
    MN<n> = Mass number indicating isotope other than most abundant isotope
    SM<n> = Spin multiplicity of atom. Possible values: 1 (singlet), 2 (doublet) or
            3 (triplet)
```

Atom type generated by AtomTypes::AtomicInvariantsAtomTypes class corresponds to:

```
    AS.X<n>.BO<n>.LBO<n>.<SB><n>.<DB><n>.<TB><n>.H<n>.Ar.RA.FC<+n/-n>.MN<n>.SM<n>
```

Except for AS which is a required atomic invariant in atom types, all other atomic invariants are optional. Atom type specification doesn't include atomic invariants with zero or undefined values.

In addition to usage of abbreviations for specifying atomic invariants, the following descriptive words are also allowed:

```
X : NumOfNonHydrogenAtomNeighbors or NumOfHeavyAtomNeighbors
BO : SumOfBondOrdersToNonHydrogenAtoms or SumOfBondOrdersToHeavyAtoms
LBO : LargestBondOrderToNonHydrogenAtoms or LargestBondOrderToHeavyAtoms
SB :  NumOfSingleBondsToNonHydrogenAtoms or NumOfSingleBondsToHeavyAtoms
DB : NumOfDoubleBondsToNonHydrogenAtoms or NumOfDoubleBondsToHeavyAtoms
TB : NumOfTripleBondsToNonHydrogenAtoms or NumOfTripleBondsToHeavyAtoms
H :  NumOfImplicitAndExplicitHydrogens
Ar : Aromatic
RA : RingAtom
FC : FormalCharge
MN : MassNumber
SM : SpinMultiplicity
```

AtomTypes::AtomicInvariantsAtomTypes module is used to assign atomic invariant atom types.

SetFunctionalClassesToUse

```
$TopologicalAtomPairsFingerprints->SetFunctionalClassesToUse($ValuesRef);
$TopologicalAtomPairsFingerprints->SetFunctionalClassesToUse(@Values);
```

Sets functional classes invariants to use during *FunctionalClassAtomTypes* value of *AtomIdentifierType* for topological atom pairs fingerprints generation and returns *TopologicalAtomPairsFingerprints*.

Possible values for atom functional classes are: *Ar, CA, H, HBA, HBD, Hal, NI, PI, RA*. Default value [Ref 24]: *HBD,HBA,PI,NI,Ar,Hal*.

The functional class abbreviations correspond to:

```
HBD: HydrogenBondDonor
HBA: HydrogenBondAcceptor
PI :  PositivelyIonizable
NI : NegativelyIonizable
Ar : Aromatic
Hal : Halogen
H : Hydrophobic
RA : RingAtom
CA : ChainAtom
```

```
 Functional class atom type specification for an atom corresponds to:

    Ar.CA.H.HBA.HBD.Hal.NI.PI.RA or None
```

AtomTypes::FunctionalClassAtomTypes module is used to assign functional class atom types. It uses following definitions [Ref 60-61, Ref 65-66]:

```
HydrogenBondDonor: NH, NH2, OH
HydrogenBondAcceptor: N[!H], O
PositivelyIonizable: +, NH2
NegativelyIonizable: -, C(=O)OH, S(=O)OH, P(=O)OH
```

SetMaxDistance

```
$TopologicalAtomPairsFingerprints->SetMaxDistance($Distance);
```

Sets maximum distance to use during topological atom pairs fingerprints generation and returns *TopologicalAtomPairsFingerprints*.

SetMinDistance

```
$TopologicalAtomPairsFingerprints->SetMinDistance($Distance);
```

Sets minimum distance to use during topological atom pairs fingerprints generation and returns *TopologicalAtomPairsFingerprints*.

StringifyTopologicalAtomPairsFingerprints

```
$String = $TopologicalAtomPairsFingerprints->
              StringifyTopologicalAtomPairsFingerprints();
```

Returns a string containing information about *TopologicalAtomPairsFingerprints* object.

AUTHOR

Manish Sud <msud@san.rr.com>

SEE ALSO

Fingerprints.pm, FingerprintsStringUtil.pm, AtomNeighborhoodsFingerprints.pm, AtomTypesFingerprints.pm, EStateIndiciesFingerprints.pm, ExtendedConnectivityFingerprints.pm, MACCSKeys.pm, PathLengthFingerprints.pm, TopologicalAtomTripletsFingerprints.pm, TopologicalAtomTorsionsFingerprints.pm, TopologicalPharmacophoreAtomPairsFingerprints.pm, TopologicalPharmacophoreAtomTripletsFingerprints.pm

COPYRIGHT