1 Molecular Similarity

Background

When applying machine learning methods to problems that require a molecule as input, we want to be able to quantify molecular similarity to accurately determine if our models can generalize well to unseen inputs. One of the most commonly used methods for quantifying molecular similarity is Tanimoto Similarity [1] on Morgan Fingerprints [2]. This essentially assigns some real-valued similarity to each pair of molecules based on the similarity of their fingerprints. However, this value is very difficult to interpret on its own and obfuscates the molecular graph origin of the Morgan Fingerprints.

Goal

Students should develop an alternative similarity/distance function that takes in two molecular graphs. One possible direction is to develop a metric based Graph Edit Distance [3], specialized for molecular graphs. Students should evaluate their method by (i) selecting a machine learning method on molecular inputs that published their training-test splits (for example [4]) and evaluating how different the training and test datasets are and (ii) evaluating scaffold-splitting (see [5] for a reference implementation) by comparing how different dataset splits are based on Tanimoto Similarity and the new similarity metric.

References

[1] https://www.jstor.org/stable/1706749
[2] https://doi.org/10.1021/c160017a018
[3] https://doi.org/10.1007/s10044-008-0141-y
[4] https://doi.org/10.1016/j.cell.2020.04.001
[5] https://github.com/chemprop/chemprop/blob/master/chemprop/data/scaffold.py

2 Molecular Graph Planarity

Background

Planar graphs are graphs that can be embedded into a plane. It has been shown that the presence of certain subgraphs are sufficient to show that a graph is planar (see review [1]). Planar graphs are particularly interesting because the subgraph isomorphism problem, while NP-complete for general graphs, can be solved in linear time for planar graphs [2]. While molecular graphs visually appear to be planar, there has not been any formal proof of this.

Goal

Students should first formally define what constitutes a semantically valid molecular graph. They should then attempt to prove (or disprove) planarity of their molecular graphs. As this is likely difficult to do for general molecules, it is perfectly acceptable to work on simplified definitions of molecules. For example, one can focus only on molecules that consist of Carbon, Oxygen, or Hydrogen atoms that have canonical degrees (4, 2, and 1 respectively). One can use the SMILES definition [3] as a reference for canonical degrees of organic atoms.

References

[1]: https://doi.org/10.1002/jgt.3190050304
[2]: https://doi.org/10.1142/9789812777638_0014
[3]: https://daylight.com/dayhtml/doc/theory/theory.smiles.html Section 3.2.1

3 Molecular String Representation

Background

One very commonly used method of representing molecules as strings is the SMILES format [1]. In short, it performs a depth-first traversal of a modified spanning tree of the molecular graph and outputs nodes and edges as they are visited. However, SMILES suffers from one shortcoming in that its validity cannot be determined by a context-free grammar. Ringbonds are denoted via numbered tags (1, 2, 3, etc.) that do not need to properly nest (1212 is an acceptable pattern for ringbond denotations). This is typically handled by current parsers by compartmentalizing syntactic validity (can be recognized by a SMILES parser) and semantic validity (represents a coherent molecule) and placing ringbond matching into the semantic validity camp.

Goal

Students should examine alternative ways of serializing molecular graphs into strings, or a similarly easily written type (not full file formats). Integers are a viable alternative. Students should include implementations of serialization and deserialization into this new molecular format. Students can safely assume input molecules will consist of only single, double, or triple bonds (no aromatic bonds) and students do not need to include any information about stereochemistry in their format.

References

[1]: https://daylight.com/dayhtml/doc/theory/theory.smiles.html

4 Interpretable Molecular Property Prediction

Background

Interpretable machine learning is a hot topic in both method development and application in recent years. A machine learning model can be debugged and audited only if it is interpretable [1]. Moreover, interpretable models in real world application could reduce safety concerns and enable social acceptance. In science, an interpretable model can go beyond prediction to help gain expert knowledge.

Graph neural network (GNN) has been widely applied to predict small molecule properties in black-box ways [2-3]. However, understanding which part of a small molecule structure contributes to the property will enable understanding of pharmacology, discovery of novel pharmacophores and design of optimized molecules.

Goal

Recently, several interpretable machine learning techniques for GNN have been proposed. In this project, we will apply an interpretable GNN method [4-6] to molecular property prediction and identify the important substructures related to the property.

References

[1]: Interpretable machine learning, Christoph Molnar.
https://christophm.github.io/interpretable-ml-book/
[2]: A review paper of GNN on property prediction.
https://www.sciencedirect.com/science/article/pii/S1740674920300305
[3]: Deep learning model to discovery antibiotics.
https://www.sciencedirect.com/science/article/abs/pii/S0092867420301021
[4] https://arxiv.org/abs/2007.00119
[5] https://arxiv.org/abs/2001.06216
[6] https://arxiv.org/abs/1903.03894
[7] Useful data source: http://moleculenet.ai/

5 Fast Protein Domain Detection

Background

Identifying functional domains from protein sequence is an important step of protein annotation. Traditionally, protein domain annotation is done by searching a protein family motif database [1-2] against the query protein sequence using HMMer [3]. However, as the protein family motif database becomes increasingly larger and the requirement of annotating a large number of protein sequences, traditional motif searching methods are not scalable.

Protein domain recognition problem is similar to object recognition in computer vision. Recent developments of object detection methods can be successfully used to efficiently recognize objects in modern videos (20+ frames/s) [4-5], which avoids wastefully going through an exhaustive list of potential object locations.

Goal

We will apply an object detection method in computer vision to detect functional domains in proteins sequences. First, we will generate training data and build a baseline model by using HMMer to annotate nonribosomal peptide synthase proteins. Second, we will adapt an object detection method to detecting protein domains so as to improve the domain detection efficiency while maintaining the accuracy.

[1] Pfam database: http://pfam.xfam.org/
[2] InterPro database: https://www.ebi.ac.uk/interpro/
[3] HMMer: http://hmmer.org/
[4] https://arxiv.org/pdf/1904.07850.pdf
[5] https://arxiv.org/pdf/1904.08189.pdf

6 Peptide Fingerprint Prediction

Background

Identifying precursor peptides from tandem mass spectra (MS2) is an open problem in proteomics. There are two strategies to address this problem: in silico database search and de novo prediction. In silico database search method first predicts the theoretical spectra of known peptide sequences in a database and then matches the experimental and the theoretical spectra. De novo prediction methods do not use any side information and directly predict the peptide sequence based from MS2.

However, both methods suffer from the mixture spectra problem which is due to fragmentation of multiple peptide precursors in a single MS2 scan. Mixture spectra decreases the identification performance of database search engines. De novo sequencing approaches are expected to be even more sensitive to the quality of the MS2. Moreover, mixtured spectra are unavoidable in MS2 acquisition techniques such as data independent acquisition [1]. One possible solution to the mixture spectra problem is to predict the peptide fingerprint (k-mers) instead.

Goal

We will develop a machine learning method to predict 3-mers based on MS2. First, we will train a model on NIST peptide spectral library [2]. Then, we will apply this method to search for either real mixtured spectra or simulated mixtured spectra from data independent acquisition[1]

[1] https://pubmed.ncbi.nlm.nih.gov/31919359/
[2] Peptide spectral library NIST datasets
https://chemdata.nist.gov/dokuwiki/doku.php?id=peptidew:download

7 Clustering Mass Spectra Datasets

Background

Most mass spectral datasets contain spectra that are nearly identical. It is usually beneficial to detect and remove these duplicates for downstream analysis. This can be done by clustering to identify the unique ones.

Spectral repositories contain billions of spectra making brute force quadratic techniques computationally intensive. Two approaches have been proposed to address this challenge. In the first approach, hierarchical clustering based on dot-product measure of similarity is used to speed up clustering [1]. The second approach is based on the fact that mass spectra is sparse. In this approach, the peaks in the mass spectrum are indexed in a table, which is used for fast scan of query spectra [cite MS-Fragger]. Each of these ideas, on their own, results in ~100X speed up in clustering.

Goal

To design and implement a method that combines hierarchical clustering and sparse indexing ideas. This could be a general purpose hierarchical clustering that is specifically designed for dot-product similarity measure and sparse dara.

[1] https://pubmed.ncbi.nlm.nih.gov/18067247/

8 Speeding up search for natural product discovery by integer linear programming

Natural products usually have potent antimicrobial and antitumor molecules, but currently high-throughput techniques for natural product discovery by mining large datasets is nonexistent. One of the fundamental problems in natural product discovery is the efficient search of potential natural product structures against mass spectra. Consider a string of n nodes, where each node i has mass of either a_i or b_i . This gives us a total of 2^n possible combinations.

Here is the problem of identifying the molecular structure of antibiotics from mass spectrometry signals. For each combination of the 2^n combination of masses $X = \{x_1, x_2, \ldots, x_n\}$, we compute theoretical spectra of X, $TheoSpec(X)$, as the set of all $n(n-1)$ sums of (potentially cyclic) substrings of X. For example for $X = \{3, 5, 6, 2\}$

$$TheoSpec(X) = \{3,\ 3+5,\ 3+5+6,\ 3+5+6+1,\ 5,\ 5+6,\ 5+6+2,\ 6,\ 6+2,\ 6+2+3,\ 2,\ 2+3,\ 2+3+5\}$$

$$= \{2,\ 3,\ 5, 6,\ 8,\ 10,\ 11,\ 13,\ 14,\ 15\}$$

Given a mass spectrum Y the goal is to find combination X (among 2^n possible combinations) that maximize

$$Score(X, Y) = |TheoSpec(X) \cap Y|$$

For example, if $Y = \{1, 3, 5, 14, 17\}$, then

$$Score(X, Y) = |\{2,\ 3,\ 5,\ 6,\ 8,\ 10,\ 11,\ 13,\ 14,\ 15\} \cap \{1, 3, 5, 14, 17\}| = |\{3, 5, 14\}| = 3$$

Goal

Currently, this computational is not feasible for large values of n, as the runtime grows with $O(2^n n^2)$. The goal of this project is to develop integer linear programming approaches to make this problem feasible for larger values of n.

9 Molecular Classification

Background

ClassyFire [1,2] is a commonly used method to annotate molecules with a tree-based taxonomy. This allows broad categorization of a molecular dataset. It works by associating molecular substructures with particular compound classes. Currently it is available as a web application that users can query. Large scale classifications are difficult to do with this web application.

Goal

Design and implement a machine learning model to classify molecules into their ClassyFire superclasses (optionally other levels of the chemical ontology used behind the scenes in ClassyFire). Students should generate their own training data by inputting compounds into ClassyFire. Any machine learning model that operates on molecular structures is acceptable, but exploring a graph neural network based model for this is especially interesting.

[1] https://doi.org/10.1186/s13321-016-0174-y
[2] classyfire.wishartlab.com/

10 Molecular Generation Traces

Background

GraphRNNs [1] differ from message passing neural networks in that they operate on traces of the nodes of a graph instead of aggregating neighborhoods into latent vectors. They can be very useful for generative models, since generating graphs according to some trace of nodes is very natural. However, there hasn't been any exploration of exactly what kind of trace is a good choice for molecular graphs.

Goal

Design and compare multiple methods of ordering nodes in a molecular graph (should take into account more chemical context than a simple random breadth-first trace). This will include the implementation of a GraphRNN and selection of a benchmarking task. Some possible tasks (including training data) are any of the ones in MoleculeNet [3].

[1] http://proceedings.mlr.press/v80/you18a.html
[2] classyfire.wishartlab.com/
[3] Useful data source: http://moleculenet.ai/