Research Progress

Nicolás López

July 10th, 2020

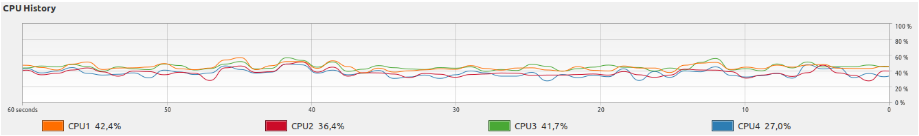
PIPPL3: *Protein-Protein Interaction Predictor using Length-3 paths information*

After meetings with J. Finke and M. Peñuela, the following issues were addressed:

1. Verify parallel computing of `Node2Vec` (N2V). Try `karateclub` implementation of edge embedding instead of `Node2Vec`.
2. Obtain an Average ROC Curve for all of the repetitions
3. Let the percentages fixed, as follows: r=0.90 (10% edge removal), valid_percent = 20 (20% of dataset for validation), test_percent = 20 (20% of dataset for test inside training, after extracting valid_percent)
4. The current results comparison is not sound. The subset of edges for testing different models should be the same.
5. Models of interest are:
    (a) L3 results
    (b) A3 Results
    (c) CN Results
    (d) L3+`Node2Vec` results
    (e) A3+`Node2Vec` results
    (f) `Node2Vec` results
6. Visualization colors of Confusion Matrix should be changed.
7. In the original paper (Barabasi et al.), the Human Interactome is used. It is recommended to test the model on both Human and Rice Interactomes.

Work done:

- (1.) The `Node2Vec` algorithm was tested for parallelism. The random walks procedure (`n2v = Node2Vec(G,...)`) uses CPUs correctly, while the N2V model training (`model = n2v.fit(...)`) does not. This is a problem as this is the time bottleneck fo the whole algorithm.

  After verifying the N2V implementation, it can be seen that the workers parameter is correctly passed to `gensim`, therefore this problem lies in the latter.

  When evaluating the `karateclub` implementation for Neighborhood-Based Node Level Embedding, specifically *DeepWalk* Algorithm (See this). This problem could not be avoided in a simple fashion algorithmically.

  Since it is hard to ensure connectivity in the graph after removing edges randomly, the only way to overcome this would be to rewrite the library. Furthermore, `karateclub` has not implementation of Edge Embedding.

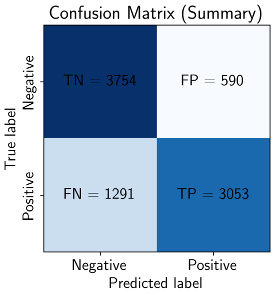
- (2.) The ROC Curves are now stored in memory (rocs_x and rocs_y) and then displayed in a final plot called `ROC_SUMMARY.png`. Also, the average AUC is calculated.

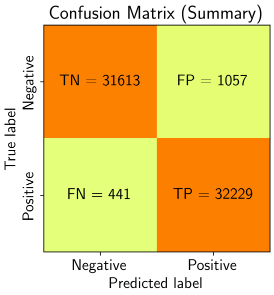
- (3.) Sampling percentages for training and validation are now fixed:

  - `r = 0.90` (10% edge removal from the graph $G = (V, E)$)
  - if R is the set of edges removed from E, then R' edges are chosen randomly such that $R' \in E$, $R \cap R' = \emptyset$ and $|R| = |R'|$ (balanced dataset for training and validation).
  - `valid_percent = 20` (20% of dataset for validation)
  - `test_percent = 20` (20% of dataset for test inside training, after extracting `valid_percent`)
  - the remaining dataset after extracting `validation` and `test` is used for `training`.

- (4.) All models are now compared correctly because of the unified dataset $R \cup R'$.

- (6.) Color palette was changed from *Blues* to *Wistia*.





- (5.) Deterministic models are already calculated, as well as A3+N2V. L3+N2V and N2V models are **pending**.

- (7.) **Pending** calculations