Predicting Transcription Factor Complexes Using Similarities in Genomic Motifs

Lichy Han, Maulik Kamdar, Daniel Kim

Biomedical Informatics Training Program, Stanford University

December 9, 2015

1 Introduction and Related Work

With the completion of the Human Genome Project in 2001, researchers believed this revolutionary effort would rapidly lead to a deluge of biomedical advances. However, our improvements in understanding complex disease mechanisms and inventing new therapeutics have been limited, falling short of lofty expectations. It has become increasingly clear that knowing only the sequence of one's DNA is insufficient to capture the underlying biology. Every cell in the human body has the same genomic DNA, but each cell utilizes a different subset of the DNA to perform different functions across the body. As such, understanding gene regulation, or how cells select different parts of the genome to use, is key to understanding human developmental biology and human diseases[1].

Gene regulation is known to be an interplay of accessible DNA sequences with regulatory proteins, known as transcription factors (TFs)[2]. In particular, the identity and function of a cell is very reliant on enhancers and promoters, which are regions of regulatory DNA that do not code for proteins but influence the expression of other genes. Promoters initiate transcription of downstream genes whereas enhancers stimulate the rate of transcription, and the two work together in concert to regulate gene transcription. Specifically, these DNA elements influence nearby genes by binding TFs which then act as repressor or activator switches on nearby genes. This binding is specific because of sequence patterns, known as DNA motifs, which are selective for specific regulatory proteins. As such, there is a combinatorial logic encoded in DNA sequences which give hints as to which TFs and in what combinations lead to downstream effects on a cell. In other words, gene regulation is based on proteins forming complexes based on the DNA sequence they're binding to, which then leads to tissue-specific function.

The interactions between promoters and enhancers have a natural and intuitive network representation, where regulatory elements are nodes, and an undirected edge exists if the elements share

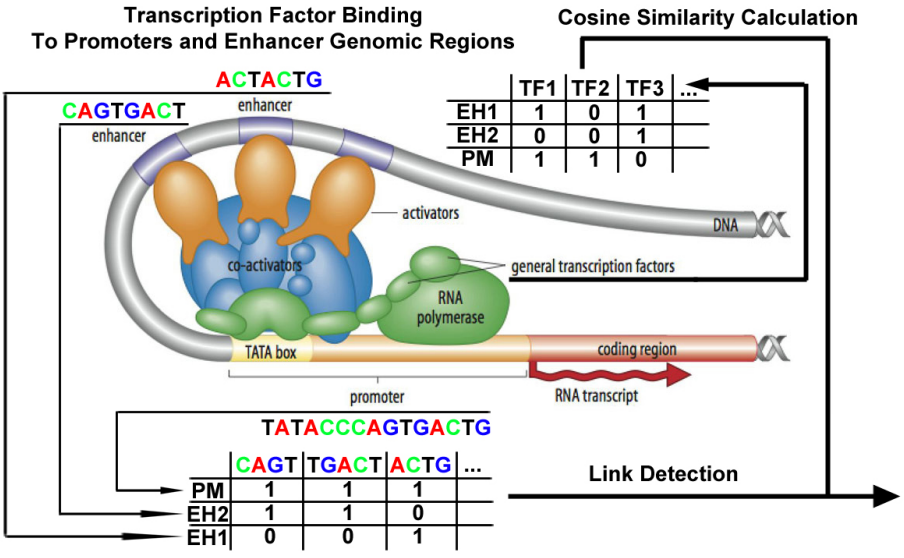

Figure 1: TF binding profile similarity and k-mer featurization of promoters and enhancers.

similar binding motifs. TF complexes, which are of primary interest in understanding transcription regulation, would manifest as clusters of regulatory elements that share similar motifs. In order to characterize all of these regulatory interactions, experimentalists would need to individually test billions of possible edges. Thus, our understanding of these networks is far from complete, and suffers from limitations in experimental results and knowledge curation. In order to combat this issue, multiple approaches have been developed in link prediction and community detection of regulatory elements.

Because of the abundance of data in yeast, network methods have been extensively developed in this organism. One of the first papers applying network ideas to gene regulation built a network by looking at genomic localization of tagged regulators across the genome [3]. This experimental procedure allowed them to generate a network of transcriptional regulators and how they are associated with each other across genomic positions. They then looked for network motifs within this network to better understand regulatory architecture within the genome. After doing community detection, they then took the groups of genes and the promoters associated with those genes to perform motif analysis and find motifs that define regulatory binding.

In a similar work in yeast, an algorithm was developed to find a network of gene modules utilizing optimization methods [4]. By optimizing for the most genes explained by the minimal set of transcription factors and then iteratively adding to seed modules, the algorithm performs link prediction and community detection to find biologically meaningful modules.

In a larger piece of work, a regulatory network was built across all DNA binding data in the ENCODE data compendium [5]. This network simply relied on overlap of DNA motifs across the genome to determine edges in the network, but this simple methodology led to many key insights in gene regulation across many cell types.

Given the richness of applying network concepts to gene regulation, we thus propose to take gene regulation concepts and additionally apply machine learning concepts to enable robust link prediction that refines the regulatory network.

2 Methods

2.1 Data

We extracted data from the Encyclopedia of DNA Elements (ENCODE) consortium[6], which aims to identify and catalog all DNA functional elements, including promoters and enhancers. As DNA regulatory elements are cell type specific, we used data from GM12878, a precursor white blood cell line with a significant number of genomic datasets. To find enhancers and promoters, we utilized chromatin accessibility data (DNase-seq) [7] in GM12878 in addition to available chromatin state maps [8] to annotate enhancer and promoter regions within this cell type. These enhancers and promoters thus become the nodes of the graph.

In order to perform edge prediction, we featurized our nodes by noting the presence of k-mers (k length sequences) in each DNA sequence to make a k-mer vector (Figure 1). For k, we used 4 and 5, as DNA motifs are inherently this length, and a value less than 4 would be redundant and provide little biological insight. Thus, each feature vector contains every possible 4-mer and 5-mer, and the presence of any particular k-mer would be marked with a 1 in the corresponding vector index. After constructing the node features, each edge is featurized by considering the shared k-mers between the two nodes (i.e. an 'AND' between the two vectors).

We used data from chromatin immunoprecipitation sequencing (ChIP-seq) experiments in GM12878 [9][10]. ChIP-seq is an experimental technique that conveys information about where a protein binds to DNA genomically. Using ChIP-seq data, we identified which TFs bind to each promoter and enhancer. In ENCODE, 86 transcription factors have ChIP-seq data in this cell type. We extracted this data and annotated each enhancer and promoter with the presence or absence of a ChIP-seq peak for each TF (Figure 1). Using the TF binding patterns, we calculated the cosine similarity (Equation 1) and the Jaccard similarity (Equation 2) between each pair (A, B) of enhancers and promoters.

$$C(A,B) = \frac{A \cdot B}{||A||||B||} \tag{1}$$

$$J(A,B) = \frac{|A \cap B|}{|A \cup B|} \tag{2}$$

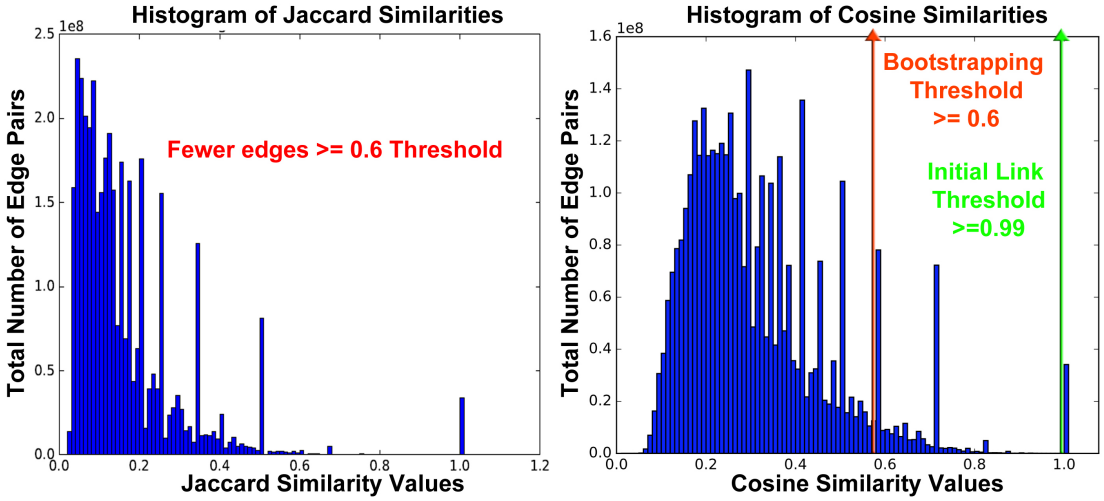

Figure 2: Histograms of Jaccard and cosine similarities between different enhancers and promoters

Based on the distribution of distances (Figure 2), we chose to use the cosine similarity due to its smoother and more widespread distribution. As the majority of the edges had a Jaccard similarity value of less than 0.2, the cosine similarity metric offers more granularity. We considered our "positive edge set" to be any edge with a similarity greater than or equal to 0.99, and we considered the "unknown edge set" as pairs of nodes with cosine similarity greater than or equal to 0.6. Our "negative edge set" were those edges with a similarity value below 0.6. These cutoffs were selected empirically. The 0.6 cutoff was selected as these represent approximately the top 5% of all edges. As edges become increasingly dissimilar, their likelihood of sharing underlying k-mer motifs decreases, thus making less similar edges unlikely to be selected and less meaningful biologically. The 0.99 cutoff is a significant peak in both the tanimoto and cosine similarity histograms that deviates from the decreasing pattern, indicating that their high similarity is likely to be of biological relevance. We kept our edge placement threshold stringent as these serve as positive controls for our machine learning edge prediction.

2.2 Link Prediction via Boosting

We start with a network of 50,000 randomly selected edges from the positive edge set, in which the nodes represent regulatory DNA regions of the genome, the existing edges represent shared TF binding

Algorithm 1 Confidence-rated AdaBoost

Given: $(x_1, y_1), ..., (x_m, y_m)$ where $x_i \in X$, $y_i \in \{-1, +1\}$.
Initialize: $D_1(i) = 1/m$ for $i = 1, ..., m$
for $t = 1$ to T **do**
  Train weak learner using distribution D_t
  Get weak hypothesis $h_t : X \to \mathbb{R}$
  Choose $\alpha_t \in \mathbb{R}$
  Select h_t and α_t to minimize the normalization factor

$$Z_t = \sum_{i=1}^{m} D_t(i) \exp(-\alpha_t y_i h_t(x_i))$$

  for $i = 1$ to m **do**
    Update $D_{t+1}(i) = \dfrac{D_t(i) \exp(-\alpha_t y_i h_t(x_i))}{Z_t}$
  end for
end for
Output final hypothesis $H(x) = \text{sign}\left(\sum_{t=1}^{T} \alpha_t h_t(x)\right)$

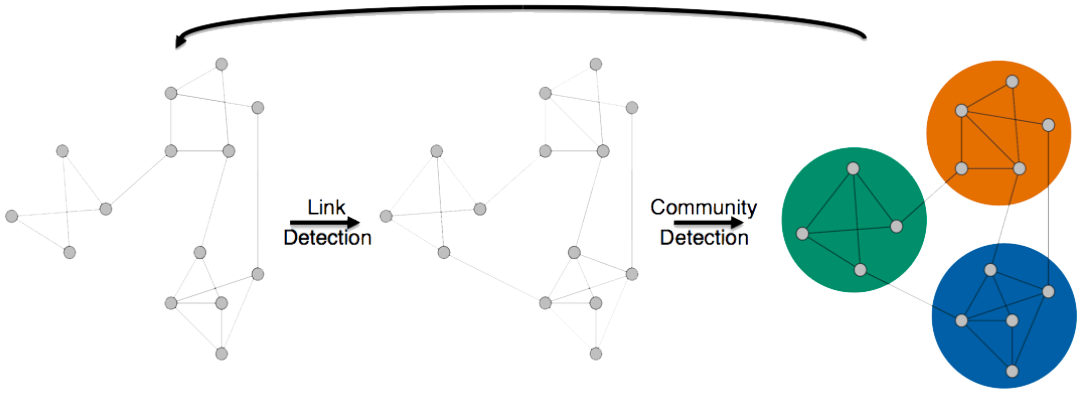

Figure 3: Iterative link detection and community detection

patterns, and every edge has a feature vector built around shared DNA sequence k-mers. Our method starts with an iterative refinement. To start, we first perform edge prediction utilizing confidence-rated boosting [11]. The importance of boosting here is that the TF complexes are combinatorial. As such, linear methods will not be as successful in predicting a shared combinatorial feature set. We utilize confidence rated AdaBoost, detailed in algorithm 1. This supervised method is trained on the current network of positive edges and an equal number of edges subsampled from the negative set so that we are able to run the algorithm quickly and efficiently. We run the boosting algorithm for 200 iterations each time.

Once trained, the boosting algorithm is then used to make prediction on unknown links from the unknown edge set. Since we have confidence-rated predictions, we then take the highest confidence edges and add them to the set of known edges. We perform community detection using available methods (described below) and note the communities formed for evaluation. We then re-train again with this new set of edges, and we do this iteratively until we begin to see overfitting (at which point the active learning based on boosting margins has been exhausted). At the end, we perform community detection one more time, and then we analyze the final communities for enrichment in specific TF complexes (Figure 3).

2.3 Community Detection

Based on the comparative analysis of community detection algorithms conducted by Lancichinetti et al. [12], we selected three methods to detect communities of promoters and enhancers based on their transcription factor binding profiles.

1. **CFinder** [13]: This method allows the detection of overlapping communities in a network. The method defines a community as a union of k-cliques (i.e. complete subgraphs of size k) and use an existing algorithm with an exponential run-time for locating all cliques in a network and then performing component analysis on a clique-clique matrix to detect communities.

2. **Infomap** [14]: This method compresses the information of a random walk across the graph, by optimizing a quality function (minimum description length [15]) using greedy search.

3. **Louvain** [16]: This method relies upon a local-optimization heuristic for maximizing the Newman-Girvan modularity [17] in the neighborhood of each node. Newman-Girvan Modularity is a metric that quantifies the quality of an assignment of nodes to communities by evaluating how much more densely connected the nodes within a community are compared to how connected they would be, on average, in a suitably defined random network. After a partition is obtained, nodes are replaced by node groups and this process is iterated until the modularity does not increase with respect to the original network. The use of a heuristic increases scalability for large networks with desirable accuracy and computational complexity.

Out of the 3 methods, only CFinder allows the existence of overlapping communities, i.e. a node can be present in two or more communities. However, to the best of our knowledge, CFinder does not

Iteration	Edges Added	Communities Detected
0 (start)	100,000	2,743
1	8,228	2,337
2	2,576	2,324
3	2,079	2,321
4	2,062	2,334
5	2,132	2,332
6	0	2,337
7	3	2,337
8	2	2,337
9	3	2,332
10	2,114	2,336

Table 1: Edges added and communities detected on every iteration

have a C++ API. On a computer with 16GB RAM, we were unable to completely load our network and execute the method to detect communities, due to memory requirements. A possible reason can be that the computational complexity of this method is very high, as the computational time needed to find all k cliques of a graph is an exponentially growing function of the graph size.

3 Results

3.1 Network Creation

From ENCODE, we curated 57,531 enhancers and 66,819 promoters in GM12878. These 124,530 nodes yield over 7 billion possible edges. A histogram of all cosine similarities for all edges with nonzero similarity is shown in the right panel of Figure 2. Using the cosine similarity, We used a threshold of 0.99 as our initial link threshold to be used as our positive edge set in training our classifier. Our negative edge set (no edge between two nodes) are those pairs of nodes that have a similarity below 0.6. Any edge with a similarity between 0.6 and 0.99 was considered an unknown edge to be used in link prediction. Out of 3,703,163,268 total nonzero edges, 34,134,515 edges had similarity values greater than 0.99 and 162,516,059 edges had similarity values between 0.6 and 0.99.

After constructing the initial edge set of our network, we featurized each edge by taking the intersection of the k-mers represented in the motifs of the two connected nodes. Using only $k = 4$ and $k = 5$ with 5 base choices (A, G, C, T, No Call) results in 3,750 binary features for each edge. Each edge feature set is thus represented as bit vector to be used in our boosting framework.

3.2 Link Prediction

We created our positive and negative training set by generating a bootstrap sample set from the entire set of edges. We randomly selected 50,000 edges from the positive edge set with cosine similarity greater than 0.99, and 50,000 edges from the negative edge set with cosine similarity less than 0.6.

We then ran the boosting algorithm on the edges and gained 19,199 new edges, with a certain number per each iteration (Table 1). We found that later iterations did not add as many edges — this may be due to subsampling considerations, since the subsampling was only redone from the original 100,000 edges chosen at the very beginning. This phenomena could likely be remedied if we did a full resampling at each iteration, allowing us to draw from the diversity of all known edges.

3.3 Community Detection

In the initial stages, both Louvain and Infomap methods result in almost the same community structure for our data — a possible explanation can be that the graphs in the initial iterations are mainly composed of disconnected components, and nodes of any component are interlinked with all other nodes in the component. However, Infomap takes much more time for execution, and almost exceeds memory requirements on a standard 16GB Memory machine. Running this method on a compute

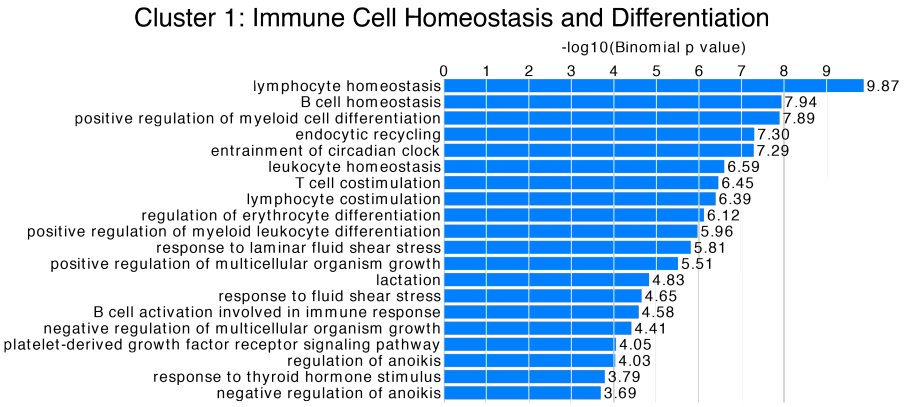

Figure 4: Gene ontology enrichment for cluster 1

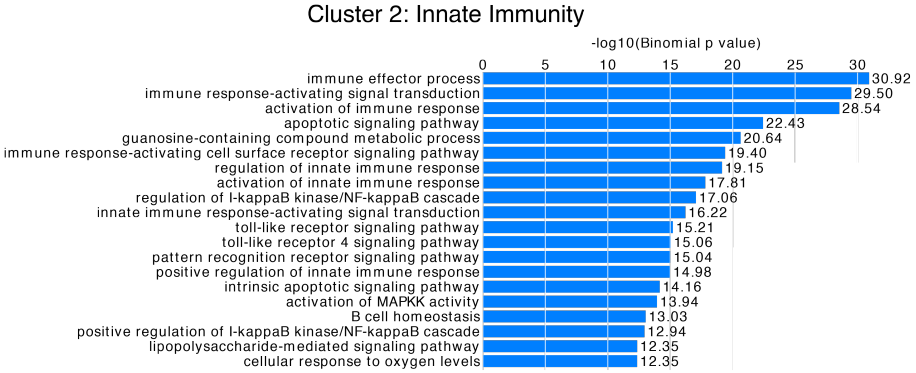

Figure 5: Gene ontology enrichment for cluster 2

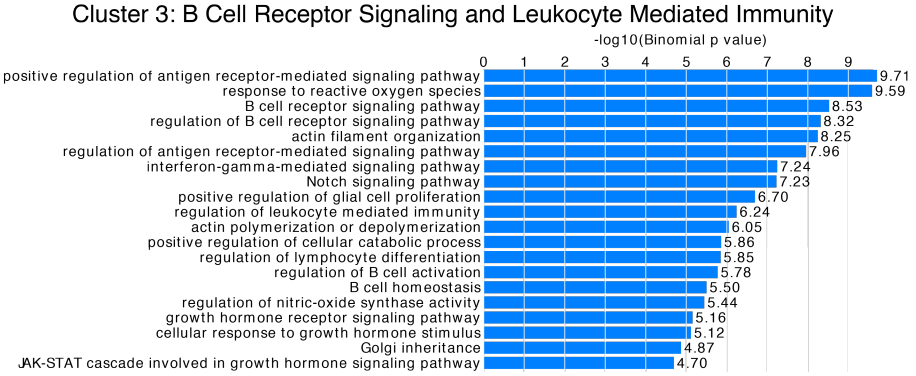

Figure 6: Gene ontology enrichment for cluster 3

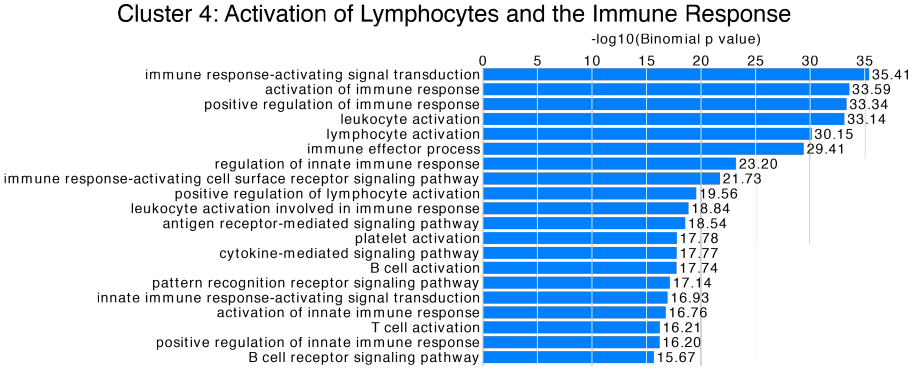

Figure 7: Gene ontology enrichment for cluster 4

cluster takes approximately one hour per iteration, whereas the Louvain method takes only approximately two minutes.

The following research only uses the Louvain method for community detection, due to its scalability and efficiency. As shown in the Table 1, we observed a sharp drop in the number of communities as we increased the number of iterations. It can be argued that the new edges generated are joining the previously disconnected components together, however, we checked and the Louvain method is not simply presenting connected components as communities, as the number of inter-community edges also increases (not shown here), i.e. nodes with an edge between them are clustered in different communities. Hence, a reasonable explanation is that our missing link prediction results in tighter, more relevant, communities. We further evaluate the community structure in the next section.

3.4 Gene Ontology Enrichments in Clusters

After community detection, we then used the Genomic Regions Enrichment of Annotations Tool (GREAT) [18] to determine whether the clusters of genomic regions (enhancers and promoters) had nearby genes that were enriched in specific functions. We did this check using the top 4 clusters generated by community detection. We then examined the functions of these nearby genes by analyzing the Gene Ontology (GO) [19] term enrichment in each community. These GO terms reflect biological functions or processes, and significant enrichment in particular sets of GO terms allows us to gain further insight into each community's role in our particular cell line.

At baseline, prior to link prediction, we found that each cluster had a slightly different functional enrichment. As our chosen cell line consisted of premature white blood cells, we expect to see functions related to the immune system, such as B cells and leukocytes which play key roles in the adaptive immune system. Cluster 1, which had 3248 genomic regions, showed an enrichment for GO terms related to immune cell homeostasis and differentiation, as shown in Figure 4. Cluster 2, which had 4558 regions, showed an enrichment for Gene Ontology terms related to innate immunity as well as signaling through innate immunity receptors (Figure 5). Cluster 3, which had 1990 regions, showed an enrichment for regions responsive to signaling pathways through B-cell receptors and activation of leukocyte mediated immunity (Figure 6). Cluster 4, which had 4093 regions, showed an enrichment for regions related to lymphocytic activation as well as general activation of the immune response (Figure 7). As such, we see that the community detection, even prior to link prediction and re-determination of community detection, already shows a significant segregation of functions. In other words, we find that distinct groups of genomic regions, as segregated by the transcription factors that bind them, perform different functions in the overall function of the B cell.

We then considered the clusters after link prediction from boosting. We found that only one cluster changed, Cluster 3, which grew from 1990 nodes to 5048 nodes. We also found that new GO terms were present and enriched in this cluster, specifically around the Fc receptor signaling pathway [20] (Figure 8). This is a specific signaling pathway in B cells that helps them activate and respond to foreign antigens, and reflects the changing nature of the clusters to increasingly specific cell functions. We thus found that link prediction was able to improve the community definitions and find even more specific functions within an individual community.

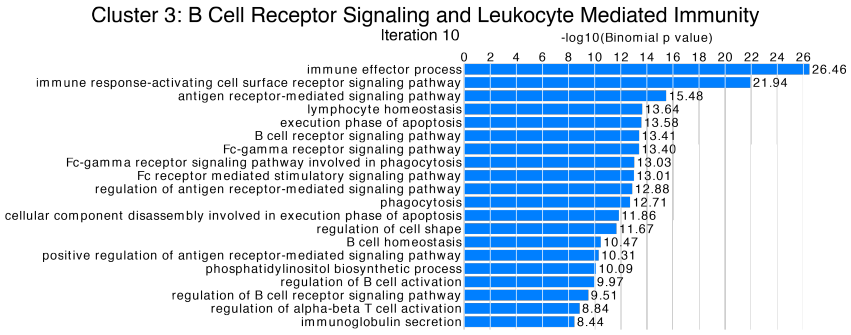

Figure 8: Gene ontology enrichment for cluster 3 after 10 iterations of link prediction

3.5 Transcription Factor Clusters

With our clusters, we looked for enrichment of ChIP-seq peaks of transcription factors. We found that each cluster initially was enriched for one transcription factor. Cluster 1 contained SPI1 (also known as PU.1), cluster 2 contained ZNF384, cluster 3 contained EBF1, and cluster 4 contained RUNX3. SPI1, EBF1, and RUNX3. All of these TFs are known to be key regulators of B cell differentiation. Of particular interest in these individual clusters is ZNF384, which is known to have a role in B-cell precursor acute lymphoblastic leukemia as a part of a recurrent fusion to EP300. This finding is of particular interest since EP300 is an important chromatin remodeller involved in activating enhancers - this suggests a strong connection between the subset of enhancers marked with ZNF384 and genes related to cancer. Although we did not find major complexes of transcription factors in the initial clusters, we found that individual transcription factors are playing important defined functional roles in the genome.

After link prediction, we found that cluster 3, which grew and gained GO terms, also became more diverse in ChIP-seq peaks that were represented. The top 10 transcription factors that were represented were RUNX3, ZNF384, SPI1, EBF1, CUX1, BATF, PAX5, STAT5A, MEF2C, and NFIC, many of which are considered master regulators within the white blood cell lineage. This enables us to investigate possible transcription factor complexes that play a role in determining B cell lineage fates and specific B cell function. This further highlights the ability of our iterative link prediction to identify specific cell function based on transcription factor binding.

4 Visualization

We extended the GenomeSnip platform [21] to visualize the enhancers and promoters, as well as the edges (positive and predicted) between them, and the detected communities. This extended version is deployed using an Amazon EC2 Instance and is accessible online at `tinyurl.com/GenomicWheel`. The user can select the iteration (original subsample, 1-10) for which he wishes to visualize the transcription factor binding network, and the edges added up to that particular iteration, and the communities detected, are processed and visualized by the platform.

As shown in Figure 9, the human genome is visualized in a circular layout called the *'Genomic Wheel'*. The 24 chromosomes of the human genome form the arcs of the Genomic Wheel. Any user can interactively select a particular chromosome of interest and zoom into it. Each chromosome is further divided into 'ideograms' — 'ideograms' are different banding patterns observed under a microscope after staining a tightly-coiled chromosome with special chemicals. Hence this provides a schematic representation of the chromosome. Starting from the innermost chromosomal layer, subsequent layers of the Genomic Wheel indicate the ideograms in a given chromosome, and the protein-coding genes in a given ideogram, respectively. All these layers can be accessed by interactively clicking the desired region of interest. Different kinds of ideograms based on their staining patterns, and different kinds of genes are represented using arcs of different colors. For example, the genes implicated in cancer are visualized using different shades of red, based on the type of role of the gene. The enhancers (pink circles) and the promoter regions (light blue circles) are superimposed on the outermost 'Gene' layer using their genomic coordinates. The different communities are visualized as circles, arranged in a Bubble Layout, in the center of the Genomic Wheel, with the size of the circle representing the number of nodes in the community. Originally the communities are shaded in a light blue color, but when the user selects an ideogram in the Genomic Wheel, the communities in which the enhancers and the promoters of the clicked region are present are highlighted with a dark blue color.

Hence in our visualization, the enhancers and promoters are represented as nodes positioned at their genomic locations. To reduce the overall clutter usually generated in node-edge visualizations, instead of visualizing all edges (including the concept of visualizing only those edges incident on nodes in a clicked ideogram), we have adopted a different method where the nodes in the clicked ideogram are connected to their relevant communities using a black colored-edge. If a particular node is connected to other nodes in different communities, then the node is also connected to those communities. The width of these node-community edges are proportional to the number of nodes in the given community, that the incident node has edges to. Edges emanating from all the nodes in a clicked ideogram are bundled to allow minimum clutter. On selecting any enhancer or promoter node, the color of the respective node-community edges turn red, and new edges from the particular community are drawn to other nodes that are connected to the incident node, and thus are similar

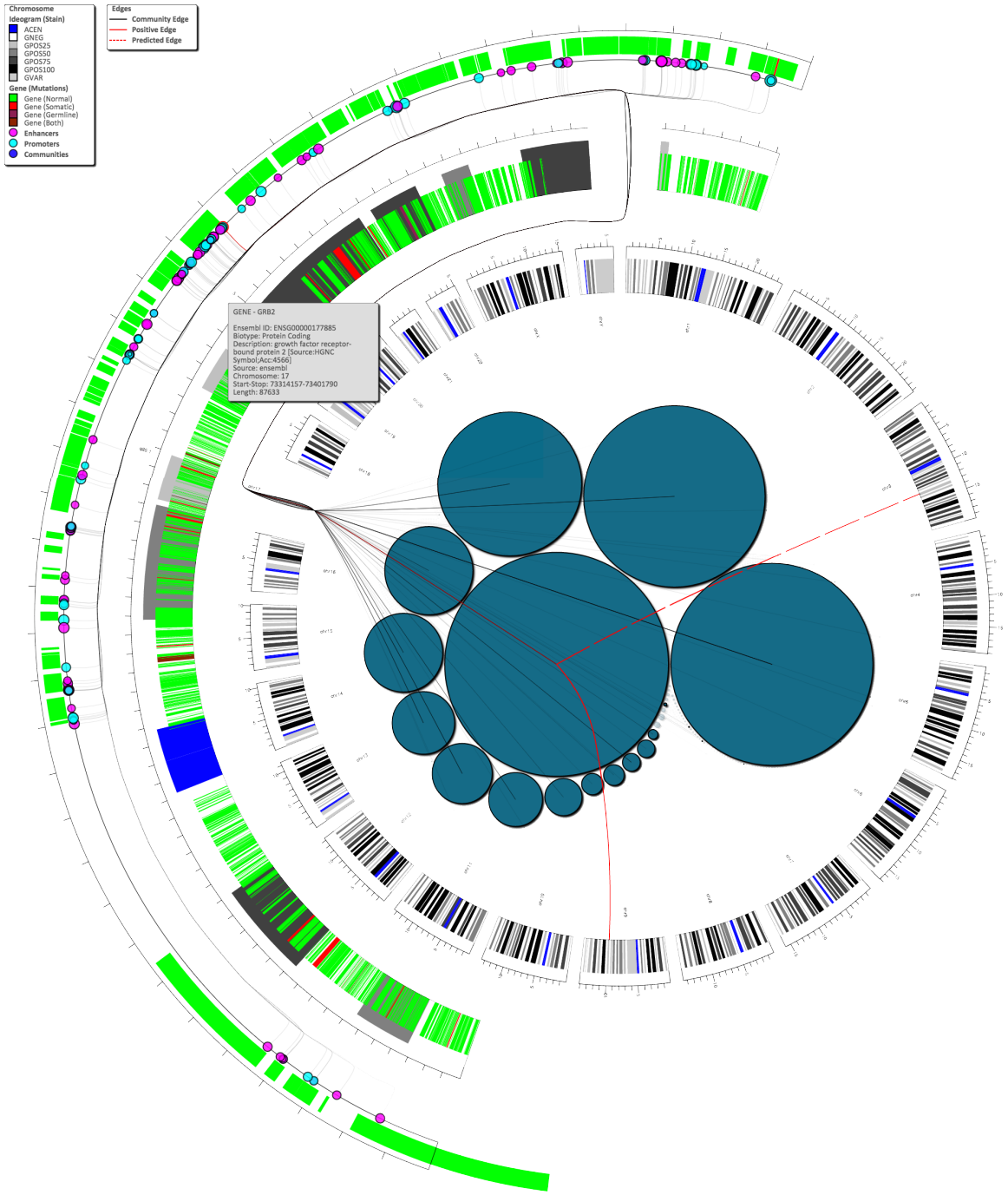

Figure 9: The enhancers and the promoters are superimposed on a circular visualization of the entire human genome called 'Genomic Wheel', with the communities present in the center of this visualization. In this example, we highlight the promoter of the GRB2 Gene, a member of the largest community, and its positive and predicted edges.

in their transcription binding profile. If the edge between the two nodes was present in the original subsampled set, i.e. a positive edge, it is represented as a straight red line. If the edge was predicted on an iteration then it is represented by a dashed red line, with the length of the dashes inversely proportional to the iteration number. For example, in Figure 9, the right-most promoter of the GRB2 gene is clicked — it can be seen that this promoter is a member of the largest community and has one positive edge to a node in chromosome 9 and a predicted edge to a node in chromosome 3.

Instead of using a generic force-directed layout with the enhancers and the promoters as the nodes, we decided to use this visualization method for two main reasons, *i)* this maps the enhancers and promoters to their respective genomic coordinates and enables a biomedical researcher to intuitively navigate the human genome and explore the desired region of interest, i.e. he can determine if the promoters or enhancers of a given gene have a similar transcription factor binding profile as some other promoter/enhancer, and *ii)* due to the large number of enhancers and promoters ($\approx 120,000$ nodes), as well as the edges between them ($\approx$34 million edges for a cosine similarity threshold of $>= 0.99$), this visualization enables to explore only one region of the human genome at a given time, and edges for only one particular node, without creating a hairball. Hence, we created a novel visualization for the transcription factor binding networks as well as the communities. We believe that this platform will help biomedical researchers to gain further insights into the different transcription factor binding complexes, and enable serendipitous knowledge exploration.

5 Conclusion

In this work, we successfully created networks of transcription factors, with edges reflecting TF similarity based on biological data. Using the Louvain method for community detection, we found that transcription factors with high motif similarity form clusters that correspond to distinct immunological functions in premature white blood cells. We further found that iteratively adding high confidence edges using confidence rated AdaBoost in a supervised machine learning model enables us to discover communities with more specific functions. Whereas the original communities detected are dominated by a single transcription factor, after iterative link prediction, these more specific communities show evidence of transcription factors complexes. We present these results using an interactive web-based visualization, which enables researchers to thoroughly explore the edges and clusters in relation to the entire human genome. Our efforts will lead to a better understanding of the TF complexes and groups of sequences that define functional roles. This will enable the development of therapies that can specifically and accurately change the course of disease, shepherding in the era of targeted therapeutics and personalized medicine.

References

[1] R. J. Britten and E. H. Davidson, "Gene regulation for higher cells: a theory," *Science*, vol. 165, no. 891, pp. 349–357, 1969.

[2] O. Hobert, "Gene regulation by transcription factors and microrna," *Science*, vol. 319, no. 5871, pp. 1785–1786, 2008.

[3] T. I. Lee, N. J. Rinaldi, F. Robert, D. T. Odom, Z. Bar-Joseph, G. K. Gerber, N. M. Hannett, C. T. Harbison, C. M. Thompson, I. Simon, *et al.*, "Transcriptional regulatory networks in saccharomyces cerevisiae," *science*, vol. 298, no. 5594, pp. 799–804, 2002.

[4] Z. Bar-Joseph, G. K. Gerber, T. I. Lee, N. J. Rinaldi, J. Y. Yoo, F. Robert, D. B. Gordon, E. Fraenkel, T. S. Jaakkola, R. A. Young, *et al.*, "Computational discovery of gene modules and regulatory networks," *Nature biotechnology*, vol. 21, no. 11, pp. 1337–1342, 2003.

[5] M. B. Gerstein, A. Kundaje, M. Hariharan, S. G. Landt, K.-K. Yan, C. Cheng, X. J. Mu, E. Khurana, J. Rozowsky, R. Alexander, *et al.*, "Architecture of the human regulatory network derived from encode data," *Nature*, vol. 489, no. 7414, pp. 91–100, 2012.

[6] ENCODE Project Consortium, "The encode (encyclopedia of dna elements) project," *Science*, vol. 306, no. 5696, pp. 636–640, 2004.

[7] L. Song and G. E. Crawford, "Dnase-seq: a high-resolution technique for mapping active gene regulatory elements across the genome from mammalian cells," *Cold Spring Harbor Protocols*, vol. 2010, no. 2, 2010.

[8] E. M. Mendenhall and B. E. Bernstein, "Chromatin state maps: new technologies, new insights," *Current opinion in genetics & development*, vol. 18, no. 2, pp. 109–115, 2008.

[9] A. Valouev, D. S. Johnson, A. Sundquist, C. Medina, E. Anton, S. Batzoglou, R. M. Myers, and A. Sidow, "Genome-wide analysis of transcription factor binding sites based on chip-seq data," *Nature Methods*, vol. 5, no. 9, pp. 829–834, 2008.

[10] P. J. Park, "Chip–seq: advantages and challenges of a maturing technology," *Nature Reviews Genetics*, vol. 10, no. 10, pp. 669–680, 2009.

[11] R. E. Schapire and Y. Singer, "Improved boosting algorithms using confidence-rated predictions," *Machine learning*, vol. 37, no. 3, pp. 297–336, 1999.

[12] A. Lancichinetti and S. Fortunato, "Community detection algorithms: a comparative analysis," *Physical review E*, vol. 80, no. 5, p. 056117, 2009.

[13] G. Palla, I. Derényi, I. Farkas, and T. Vicsek, "Uncovering the overlapping community structure of complex networks in nature and society," *Nature*, vol. 435, no. 7043, pp. 814–818, 2005.

[14] M. Rosvall and C. T. Bergstrom, "Maps of random walks on complex networks reveal community structure," *Proceedings of the National Academy of Sciences*, vol. 105, no. 4, pp. 1118–1123, 2008.

[15] P. D. Grünwald, *The minimum description length principle*. MIT press, 2007.

[16] V. D. Blondel, J.-L. Guillaume, R. Lambiotte, and E. Lefebvre, "Fast unfolding of communities in large networks," *Journal of Statistical Mechanics: Theory and Experiment*, vol. 2008, no. 10, p. P10008, 2008.

[17] M. E. Newman and M. Girvan, "Finding and evaluating community structure in networks," *Physical review E*, vol. 69, no. 2, p. 026113, 2004.

[18] C. Y. McLean, D. Bristor, M. Hiller, S. L. Clarke, B. T. Schaar, C. B. Lowe, A. M. Wenger, and G. Bejerano, "Great improves functional interpretation of cis-regulatory regions," *Nature biotechnology*, vol. 28, no. 5, pp. 495–501, 2010.

[19] M. Ashburner, C. A. Ball, J. A. Blake, D. Botstein, H. Butler, J. M. Cherry, A. P. Davis, K. Dolinski, S. S. Dwight, J. T. Eppig, *et al.*, "Gene ontology: tool for the unification of biology," *Nature genetics*, vol. 25, no. 1, pp. 25–29, 2000.

[20] M. Daëron, "Fc receptor biology," *Annual review of immunology*, vol. 15, no. 1, pp. 203–234, 1997.

[21] M. Kamdar, A. Iqbal, M. Saleem, H. Deus, and S. Decker, "Genomesnip: Fragmenting the genomic wheel to augment discovery in cancer research," in *Conference on Semantics in Healthcare and Life Sciences (CSHALS)*, 2014.