Literature Review 1

My primary paper is “**Visualizing Dataflow Graphs of Deep Learning Models in TensorFlow**”, which is published in IEEE Transactions on Visualization and Computer Graphics on August 2017. And my secondary paper is “**Level-of-Detail Visualization of Clustered Graph Layouts**”, which is referred on first one’s 10th references, published in Visualization, 2007. APVIS '07. 2007 6th International Asia-Pacific Symposium on February 2007. In primary paper, they showed a view of the TensorFlow Graph Visualizer. It is part of the TensorFlow machine intelligence platform. To visualizing their underlying dataflow graphs, the good thing helps users realize complex machine learning architectures. The tool executes by giving a series of graph transformations that enable standard layout techniques to produce a clear interactive diagram. To handle up the graph, they decouple non-critical nodes from the layout. To provide an overview, they established a clustered graph taking the hierarchical structure annotated in the source code. They perform edge bundling to enable stable and responsive cluster expansion for supporting exploration of nested structure on demand. Finally, they detected and highlight repeated structures to emphasize a model's modular composition. They describe example usage scenarios and report user feedback to show the utility of the visualizer. Users can find the visualizer useful for understanding, debugging, and sharing the structures of their models. And the secondary paper is talking about the level-of-detail techniques enable a comprehensible interactive visualization of large and complex clustered graph layouts both in 2D or 3D. Implicit surfaces are used for the visually simplified representation of vertex clusters, and so-called edge bundles are formed for the simplification of edges. Additionally, dedicated transition techniques are provided for continuously adaptive and adjustable views of graphs that range from very abstract to very detailed representations. In first one paper, because they want to simplify large graph, they build up a hierarchical clustered graph that can provide an overview and support cluster expansion to discover details. They used hierarchical namespaces that developers provide to create a clustered flow layout. To help users maintain mental maps during exploration, a clustered graph must also be responsive and stable. They utilized secondary method here. Also, drawing edges directly between all visible nodes still clutters expanded graphs. To both declutter the view and support interactive expansion, they bundled and routed edges between groups such that edges only connect nodes that are siblings in the hierarchy. The technic they used is from secondary paper. Finally, when group nodes are expanded, they routed edges along the hierarchy instead of directly drawing edges between all visible nodes. This way is also same from secondary paper. The following paragraph I will discuss about the main technic using in secondary paper, for primary paper used them three times.

The level-of-detail approach that only modifies the visual representation of the graph and does not alter the structure or the computed layout of the graph, so that it will not happens that the structure of the graph is modified, which often results in an unavoidable recalculation of the layout, eliminates relevant information about the collapsed vertices and edges. In secondary paper, the author used this method to reduce the polygon count for interactively visualizing large and complex graphs, and the visual complexity of the information in general. The technics substitutes a complex object by a simplified object or the substitution of a group of objects by single one. And groups of vertices and groups of edges are substituted if the grouping information are at different abstraction levels based on the spatial distribution of the vertices, so this method is very suitable for clustered layouts. This layout is used and mentioned in primary paper. What is Clustered Graph Layouts? It consists of a graph G and a rooted tree T such that the vertices of G are exactly the leaves of T. How about Level-of-detail? It is a model or object with different resolutions depending on its distance to the viewer. The purpose is to decrease the polygon count, whereas the reduced visual quality of the model or object ideally remains unnoticed due to its smaller size as a result of its distance and the usage of a perspective projection. The polygon reduce is to reduce the amount of presented information. Clustered graph layouts focus on the spatial grouping of vertices, whereas the edge routing or the minimization of edge crossing is of less importance. Hence, the representation of the edges is strictly constrained by the vertex positions in the layout. The edge is simply visualized by a direct connection between two vertices and all edges between two clusters of vertices are represented as one aggregated edge that is also visualized by a direct connection between the clusters. Because an edge is not a direct connection between two vertices, and it is routed according to the clusters of the graph, the routing information is used to form edge bundles, which group edges according to the cluster hierarchy. How to generate the routed edge? To combine the straight edges and edge routing tree, so that they can perform routing edges. The result is an ordered set of ports for each edge that enables the drawing of edge segments that are routed correspondent to the implicit surfaces of the clusters. Now they have many edge segments use the same ports, so edge segments between the same two ports are occluding each other. As results, the information about the number, the weight, and the structure of the edges is lost for these edge segments. The edge bundles eliminate this occlusion to form with the individual. The computation of the individual translation vectors at the ports for 2D is a non-overlapping side by side placement of the edge segments at a linear cross section of an edge bundle. The rest parts are not used in primary paper, so I will briefly discuss contents of primary on next paragraph.

They designed a visual tool that tackles one aspect of this challenge: interactive exploration of the dataflow architecture behind machine learning models. Their design focused on users and data, so that they can present a series of graph transformation to address many challenges, and demonstrate usage scenarios. Also, they discussed user reactions, which indicate that the visualizer addresses users' need. The visualize graphs with similar structures are extracting non-critical nodes seems successful: viewers apparently understand the overall structure of the graph despite the lack of direct edges. The developers were willing to change their own code in the interest of improving the visualization, manually adding metadata to their graph for clarifying the layout. It means users derived significant value from the visualizations. By using the visual technics and a tight feedback loop between artifact visualization and creation, users may be willing to add critical pieces of metadata.