Appendix

First, we define the problem for matrix factorization models, where the loss function $\mathcal{L}(\mathcal{D};\Theta_R)$ is defined as:

$$\mathcal{L}(\mathcal{D};\Theta_R) = \sum_{(u_i,v_i,r_{i,j})\in\mathcal{D}} ||\mathbf{u}_i\mathbf{v}_j^T - r_{i,j}||^2$$

Here, $\Theta_R = (\mathbf{U},\mathbf{V})$, where $\mathbf{U}$ and $\mathbf{V}$ represent the user embedding parameters and item embedding parameters, respectively, constituting the matrix factorization model parameters Θ_R, with $\mathbf{U} = (m * d), \mathbf{V} = (n * d)$, where d is the embedding size, and m and n represent the number of users and items, respectively.

For the computation of the influence function values for a specific training sample $\ddagger = (u_i, v_j, r_{i,j}) \in \mathcal{D}$, i.e., a u-i interaction, the influence function can be computed as follows if this interaction has already been involved in the training process of the matrix factorization model:

$$\mathcal{I}(u_i, v_j, r_{ij};\Theta_R) = -\nabla_{\Theta_R}\mathcal{L}(\mathcal{D};\Theta_R)^T H_{\Theta_R}^{-1}\nabla_{\Theta_R}\mathcal{L}(u_i, v_j, r_{ij};\Theta_R)$$

where $H_{\Theta_R} = \nabla^2_{\Theta_R}\mathcal{L}(\mathcal{D};\Theta_R)$, i.e., the Hessian matrix of the loss function. The direct computation of the influence function involving the Hessian matrix has a time complexity of $O(|\mathcal{D}|(m+n)^2d^2 + (m+n)^3d^3)$, hence we use a stochastic estimation method to estimate $H_{\Theta_R}^{-1}$, which is hereafter denoted as H.

The inverse matrix can be expanded as $H_j^{-1} = \sum_{i=0}^{j}(I-H)^i, \lim_{j\to\infty} H_j^{-1} = H^{-1}$.

It can be considered as a geometric matrix series, where the sum of the first j terms can be viewed as the inverse matrix, hence the recursive formula: $H_j^{-1} = I + (I-H)H_{j-1}^{-1}$.

In the computation process of the influence function, if we consider the influence function as the product of three partial derivatives, it can be observed that the first term $\nabla_{\Theta_R}\mathcal{L}(\mathcal{D};\Theta_R)$ remains constant, hence it can be computed together with H^{-1}. During the ith iteration of recursion, we randomly sample $(u_x, v_y, r_{x,y})$ from $\mathcal{D}$ to update the recursive formula:

$$H_j^{-1}v = v + (I - \nabla^2_{\Theta_R}\mathcal{L}((u_x, v_y, r_{x,y});\Theta_R))H_{j-1}^{-1}v$$

where $v = \nabla_{\Theta_R}\mathcal{L}(\mathcal{D};\Theta_R)$. We set a hyperparameter t to limit the number of samples, defining $s_{test} = H_t^{-1}v$. Since H^{-1} is a symmetric matrix, the influence function is then $\mathcal{I}(u_i, v_j, r_{ij};\Theta_R) = -s_{test}^T\mathcal{L}(u_i, v_j, r_{ij};\Theta_R)$. With the problem defined, we now aim to prove that under the stochastic estimation method, when we make minor modifications to certain parameter values, such as altering the embeddings of user x and item y, $\mathbf{u}_x, \mathbf{v}_y$, whether there is a significant change in the influence value $\mathcal{I}(u_i, v_j, r_{ij};\Theta_R)$ for another training sample $\ddagger = (u_i, v_j, r_{i,j}) \in \mathcal{D}$.

First, we perform a detailed analysis of the second-order partial derivatives during sampling. When sampling $(u_x, v_y, r_{x,y}) \in \mathcal{D}$, the recursive formula is updated, and the structure of $\nabla^2_{\Theta_R}\mathcal{L}((u_x, v_y, r_{x,y});\Theta_R)$ is decomposed as follows:

$$\nabla^2_{\Theta_R}\mathcal{L}((u_x,v_y,r_{x,y});\Theta_R)=\begin{bmatrix}\frac{\partial^2\mathcal{L}((u_x,v_y,r_{x,y})}{\partial\mathbf{U}^2} & \frac{\partial^2\mathcal{L}((u_x,v_y,r_{x,y})}{\partial\mathbf{U}\partial\mathbf{V}}\\ \frac{\partial^2\mathcal{L}((u_x,v_y,r_{x,y})}{\partial\mathbf{V}\partial\mathbf{U}} & \frac{\partial^2\mathcal{L}((u_x,v_y,r_{x,y})}{\partial\mathbf{V}^2}\end{bmatrix}$$

$$=\begin{bmatrix}0 & \dots & 0 & \dots & 0 & 0 & \dots & 0 & \dots & 0\\ \vdots & \ddots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots & \ddots & \vdots\\ 0 & \dots & \frac{\partial^2\mathcal{L}((u_x,v_y,r_{x,y})}{\partial\mathbf{u}_x^2} & \dots & 0 & 0 & \dots & \frac{\partial^2\mathcal{L}((u_x,v_y,r_{x,y})}{\partial\mathbf{u}_x\partial\mathbf{v}_y} & \dots & 0\\ \vdots & \ddots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots & \ddots & \vdots\\ 0 & \dots & 0 & \dots & 0 & 0 & \dots & 0 & \dots & 0\\ 0 & \dots & 0 & \dots & 0 & 0 & \dots & 0 & \dots & 0\\ \vdots & \ddots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots & \ddots & \vdots\\ 0 & \dots & \frac{\partial^2\mathcal{L}((u_x,v_y,r_{x,y})}{\partial\mathbf{v}_y\partial\mathbf{u}_x} & \dots & 0 & 0 & \dots & \frac{\partial^2\mathcal{L}((u_x,v_y,r_{x,y})}{\partial\mathbf{v}_y^2} & \dots & 0\\ \vdots & \ddots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots & \ddots & \vdots\\ 0 & \dots & 0 & \dots & 0 & 0 & \dots & 0 & \dots & 0\end{bmatrix}$$

Considering this large Hessian matrix, sized $(m+n)d \times (m+n)d$, we view it as a block matrix for further analysis. The component $\frac{\partial^2\mathcal{L}((u_x,v_y,r_{x,y})}{\partial\mathbf{u}_x^2}$ is a $d\times d$ matrix:

$$\frac{\partial^2\mathcal{L}((u_x,v_y,r_{x,y})}{\partial\mathbf{u}_x^2}=\begin{bmatrix}\frac{\partial^2\mathcal{L}((u_x,v_y,r_{x,y})}{\partial\mathbf{u}_{x1}^2} & \frac{\partial^2\mathcal{L}((u_x,v_y,r_{x,y})}{\partial\mathbf{u}_{x1}\partial\mathbf{u}_{x2}} & \dots & \frac{\partial^2\mathcal{L}((u_x,v_y,r_{x,y})}{\partial\mathbf{u}_{x1}\partial\mathbf{u}_{xd}}\\ \frac{\partial^2\mathcal{L}((u_x,v_y,r_{x,y})}{\partial\mathbf{u}_{x2}\partial\mathbf{u}_{x1}} & \frac{\partial^2\mathcal{L}((u_x,v_y,r_{x,y})}{\partial\mathbf{u}_{x2}^2} & \dots & \frac{\partial^2\mathcal{L}((u_x,v_y,r_{x,y})}{\partial\mathbf{u}_{x2}\partial\mathbf{u}_{xd}}\\ \vdots & \vdots & \ddots & \vdots\\ \frac{\partial^2\mathcal{L}((u_x,v_y,r_{x,y})}{\partial\mathbf{u}_{xd}\partial\mathbf{u}_{x1}} & \frac{\partial^2\mathcal{L}((u_x,v_y,r_{x,y})}{\partial\mathbf{u}_{xd}\partial\mathbf{u}_{x2}} & \dots & \frac{\partial^2\mathcal{L}((u_x,v_y,r_{x,y})}{\partial\mathbf{u}_{xd}^2}\end{bmatrix}$$

The gradient of the loss function over the entire dataset $\mathcal{D}$ with respect to Θ_R is given by:

$$\nabla_{\Theta_R}\mathcal{L}(\mathcal{D};\Theta_R)=\begin{bmatrix}\frac{\partial\mathcal{L}(\mathcal{D};\Theta_R)}{\partial\mathbf{U}}\\ \frac{\partial\mathcal{L}(\mathcal{D};\Theta_R)}{\partial\mathbf{V}}\end{bmatrix}=\begin{bmatrix}\frac{\partial\mathcal{L}(\mathcal{D};\Theta_R)}{\partial\mathbf{u}_1}\\ \frac{\partial\mathcal{L}(\mathcal{D};\Theta_R)}{\partial\mathbf{u}_2}\\ \vdots\\ \frac{\partial\mathcal{L}(\mathcal{D};\Theta_R)}{\partial\mathbf{u}_m}\\ \frac{\partial\mathcal{L}(\mathcal{D};\Theta_R)}{\partial\mathbf{v}_1}\\ \vdots\\ \frac{\partial\mathcal{L}(\mathcal{D};\Theta_R)}{\partial\mathbf{v}_n}\end{bmatrix}$$

When we estimate $H_j^{-1}v$ using a recursive approach and sample $(u_x,v_y,r_{x,y})\in\mathcal{D}$, the term $\nabla^2_{\Theta_R}\mathcal{L}((u_x,v_y,r_{x,y});\Theta_R)H_{j-1}^{-1}v$ changes. From the above decomposition of the two terms, it is clear that most positions in this matrix product are zero, with non-zero values only in derivatives related to $\mathbf{u}_x$ and $\mathbf{v}_y$. Therefore, when using stochastic estimation to calculate the influence function, if the $t+1$-th sample modifies the embedding from $(\mathbf{u}_x,\mathbf{v}_y)$ to $(\mathbf{u}_x+\Delta\mathbf{u}_x,\mathbf{v}_y+\Delta\mathbf{v}_y)$, only derivatives involving $\mathbf{u}_x$ and $\mathbf{v}_y$ change.

The change in the influence function for $(u_i,v_j,r_{i,j})\in\mathcal{D}$ when a perturbation is added to $(u_x,v_y,r_{x,y})\in\mathcal{D}$ is given by:

$$
\begin{aligned}
\mathcal{I}(u_i, v_j, r_{ij}; H_{n+1}^{-1}, \Theta_R) - \mathcal{I}(u_i, v_j, r_{ij}; H_n^{-1}, \Theta_R) = \\
-\Big[\Big(H_{t+1}^{-1} \nabla_{\Theta_R} \mathcal{L}(\mathcal{D}; \Theta_R)\Big)^T - \Big(H_t^{-1} \nabla_{\Theta_R} \mathcal{L}(\mathcal{D}; \Theta_R)\Big)^T\Big] \nabla_{\Theta_R} \mathcal{L}(u_i, v_j, r_{ij}; \Theta_R) = \\
\begin{bmatrix} 0 \\ \vdots \\ \Delta \frac{\partial^2 \mathcal{L}((u_x, v_y, r_{x,y})}{\partial \mathbf{u}_x^2} \cdot \frac{\partial \mathcal{L}(\mathcal{D}; \Theta_R)}{\partial \mathbf{u}_x} \\ + \\ \Delta \frac{\partial^2 \mathcal{L}((u_x, v_y, r_{x,y})}{\partial \mathbf{u}_x \mathbf{v}_y} \cdot \frac{\partial \mathcal{L}(\mathcal{D}; \Theta_R)}{\partial \mathbf{u}_x} \\ \vdots \\ \Delta \frac{\partial^2 \mathcal{L}((u_x, v_y, r_{x,y})}{\partial \mathbf{v}_y \mathbf{u}_x} \cdot \frac{\partial \mathcal{L}(\mathcal{D}; \Theta_R)}{\partial \mathbf{v}_y} \\ + \\ \Delta \frac{\partial^2 \mathcal{L}((u_x, v_y, r_{x,y})}{\partial \mathbf{v}_y^2} \cdot \frac{\partial \mathcal{L}(\mathcal{D}; \Theta_R)}{\partial \mathbf{v}_y} \\ \vdots \\ 0 \end{bmatrix}^T \cdot \begin{bmatrix} 0 \\ \vdots \\ \frac{\partial \mathcal{L}(u_i, v_j, r_{ij}; \Theta_R)}{\partial u_i} \\ 0 \\ \vdots \\ \frac{\partial \mathcal{L}(u_i, v_j, r_{ij}; \Theta_R)}{\partial v_j} \\ \vdots \\ 0 \end{bmatrix} = \\
\epsilon \cdot \frac{\partial^2 F}{\partial v_y \partial u_i} \cdot \Big(\frac{\partial \mathcal{L}(u_i, v_j, r_{ij}; \Theta_R)}{\partial u_i}\Big)^T + \epsilon \cdot \frac{\partial^2 F}{\partial u_x \partial v_j} \cdot \Big(\frac{\partial \mathcal{L}(u_i, v_j, r_{ij}; \Theta_R)}{\partial v_j}\Big)^T = \\
\epsilon \cdot \Big(2(u_i^T \Delta v_y + \Delta v_y^T u_i) \cdot 2(r_{iy} - u_i v_y^T) v_y^T \\
+2(\Delta u_x^T v_j + v_j^T \Delta u_x) \cdot 2(r_{xj} - u_x v_j^T) u_x^T\Big) = \\
\epsilon \cdot \sum_{(i,y) \in \mathcal{D}, t=1}^{d} \Big(\frac{\partial^2 F}{\partial v_y \partial u_i} \cdot 2(r_{iy} - u_i v_y^T) v_y \\
+ \frac{\partial^2 F}{\partial u_{xt} \partial v_{jt}} \cdot 2(r_{xj} - u_x v_j^T) v_j\Big) = \\
4\epsilon \cdot \Big((\mathbf{u}_i^T \Delta \mathbf{v}_y + \Delta \mathbf{v}_y^T \mathbf{u}_i)(r_{iy} - \mathbf{u}_i \mathbf{v}_y^T) \mathbf{v}_y^T \\
+(\Delta \mathbf{u}_x^T \mathbf{v}_j + \mathbf{v}_j^T \Delta \mathbf{u}_x)(r_{xj} - \mathbf{u}_x \mathbf{v}_j^T) \mathbf{u}_x^T\Big) \leq \\
4d^2 \Big(|\epsilon(r_{iy} - \mathbf{u}_i \mathbf{v}_y^T) v_y^T| + |\epsilon(r_{xj} - \mathbf{u}_x \mathbf{v}_j^T) u_x^T|\Big) \leq \\
20d^2 \Big(|\epsilon \mathbf{v}_y^T| + |\epsilon \mathbf{u}_x^T|\Big)
\end{aligned}
$$