Quantum Recommendation System

A recommendation system uses the past purchases or ratings of n products by a group of m users, in order to provide personalized recommendations to individual users. The information is modeled as an m × n preference matrix which is assumed to have a good rank-k approximation, for a small constant k. In the matrix each entry denotes how much an user (say m2) liked a product (say n8), higher the value higher the satisfaction.

This algorithm has a running time of O(poly(k)polylog(mn)) in contrast to the best known classical algorithms that perform the task with complexity polynomial in the dimension of the matrix.

The key difference of this algorithm from that of the classical algorithm is that in classical case we try to reconstruct the matrix i.e. find the missing values of the preference matrix that takes time polynomial to the dimension of the matrix, where as in the quantum case we try to sample for the high values and that gives us an exponential speed-up.

Assumption: The underlying assumption in recommendation systems is that one can infer information about a specific user from the information about all other users because, in some sense, the majority of users belong to some well-defined "types". In other words, most people's likes are not unique but fall into one of a small number of categories. To put it more formally, the preference matrix P of recommendation systems can be well approximated by a low-rank matrix and this assuption is widely used.

Data structure used: The input to the quantum procedure is a vector $x \in R^n$ and a matrix $A \in R^{mxn}$. We assume that the input is stored in a classical data structure such that an algorithm that has quantum access to the data structure can create the quantum state $|x>$ corresponding to the vector x and the quantum states $|A_i>$ corresponding to each row A_i of the matrix A, in time polylog(mn). A little more detail has been provided below:

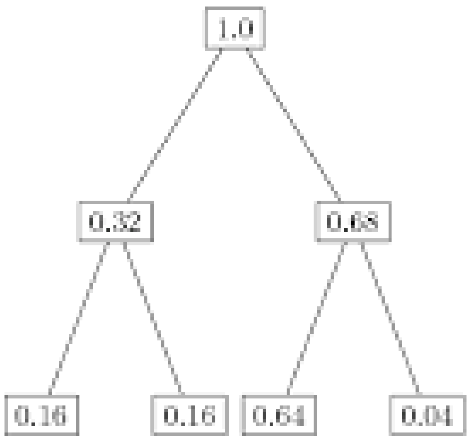
Let $A \in R$ m×n be a matrix. Entries (i, j, A_{ij}) arrive in the system in an arbitrary order and w denotes the number of entries that have already arrived in the system. There exists a data structure to store the entries of A with the following properties:

i. The size of the data structure is $O(w \cdot \log^2 (mn))$.

ii. The time to store a new entry (i, j, A_{ij}) is $O(\log^2 (mn))$.

iii. A quantum algorithm that has quantum access to the data structure can perform the mapping $U_0 : |i> |0> \rightarrow |i> |A_i>$ for $i \in [m]$, corresponding to the rows of the matrix currently stored in memory and the mapping $V_0 : |0> |j> \rightarrow |A_0>|j_i>$, for $j \in [n]$, where $A_0 \in R^m$ has entries $A_{0,i} = ||A_i||$ in time polylog(mn).

The data structure consists of an array of m binary trees B_i , $i \in [m]$. The trees B_i are initially empty. When a new entry (i, j, A_{ij}) arrives the leaf node j in tree B_i is created if not present and updated otherwise. The leaf stores the value A^2_{ij} as well as the sign of A_{ij} . The depth of each tree B_i is at most $\lceil \log n \rceil$ as there can be at most n leaves. An internal node v of B_i stores the sum of the values of all leaves in the subtree rooted at v, i.e. the sum of the square amplitudes of the entries of A_i in the subtree. Hence, the value stored at the root is $||A_i||^2$. When a new entry arrives, all the nodes on the path from that leaf to the tree root are also updated.



Let $|\phi\rangle = 0.4\,|00\rangle + 0.4\,|01\rangle + 0.8\,|10\rangle + 0.2\,|11\rangle$.

- Rotation on qubit 1:
  $|0\rangle\,|0\rangle \to (\sqrt{0.32}\,|0\rangle + \sqrt{0.68}\,|1\rangle)\,|0\rangle$
- Rotation on qubit 2 conditioned on qubit 1:

$$(\sqrt{0.32}\,|0\rangle + \sqrt{0.68}\,|1\rangle)\,|0\rangle \to$$
$$\sqrt{0.32}\,|0\rangle\,\frac{1}{\sqrt{0.32}}(0.4\,|0\rangle + 0.4\,|1\rangle)+$$
$$\sqrt{0.68}\,|1\rangle\,\frac{1}{\sqrt{0.68}}(0.8\,|0\rangle + 0.2\,|1\rangle)$$

Figure 1: Vector state preparation illustrated for 4-dimensional state $|\varphi>$

Singular value estimation: The main technical tool for recommendation system is the singular value decomposition. Given a state $|x> = \Sigma_i\, \alpha_i|v_i>$ for an arbitrary vector $x \in R^n$ the task is to estimate the singular values corresponding to each singular vector in coherent superposition. The singular value decomposition i done is following manner.

Let $A \in R^{m\times n}$ be a matrix with singular value decomposition $A = \Sigma_i\, \sigma_i u_i v^t_i$ stored in the data structure mentioned above. Let $\varepsilon > 0$ be the precision parameter. There is an algorithm with running time O(polylog(mn)/ε) that performs the mapping $\Sigma_i\, \alpha_i|v_i>\to \Sigma_i\, \alpha_i|v_i>|\sigma_{0,i}>$, where $\sigma_{0,i} \in \sigma_i \pm \varepsilon||A||_F$ for all i with probability at least 1 − 1/poly(n).

The idea for our singular value estimation algorithm is to find isometries $P \in R^{mn\times m}$ and $Q \in R^{mn\times n}$ that can be efficiently applied, and such that $A / ||A_k|| = P^tQ$. Using P and Q, we define a unitary matrix W acting on R^{mn}, which is also efficiently implementable and such that the row singular vector v_i of A with singular value σ_i is mapped to an eigenvector Qv_i of W with eigenvalue $e^{i\theta}{}_i$ such that $\cos(\theta_i/2) = \sigma_i / ||A||_F$ (note that $\cos(\theta_i/2) > 0$ as $\theta_i \in [-\pi, \pi]$). The algorithm consists of the following steps: first, map the input vector $\Sigma_i\, \alpha_i|v_i>$to $\Sigma_i\, \alpha_i\, |Qv_i>$ by applying Q; then, use phase estimation with unitary W to compute an estimate of the eigenvalues θ_i and hence of the singular values $\sigma_i = ||A||_F \cos(\theta_i / 2)$; and finally undo Q to recover the state $\Sigma_i\, \alpha_i|v_i>|\sigma_i>$. This procedure is described below:

Require: $A \in R^{m\times n}$, $x \in R^n$ in the data structure mentioned above, precision parameter $\varepsilon > 0$.

1. Create $|x> = \Sigma_i\, \alpha_i|v_i>$.

2. Append a first register $| 0^{\lceil \log m\rceil} >$ and create the state $|Qx_i >= \Sigma_i\, \alpha_i|Qv_i>$.

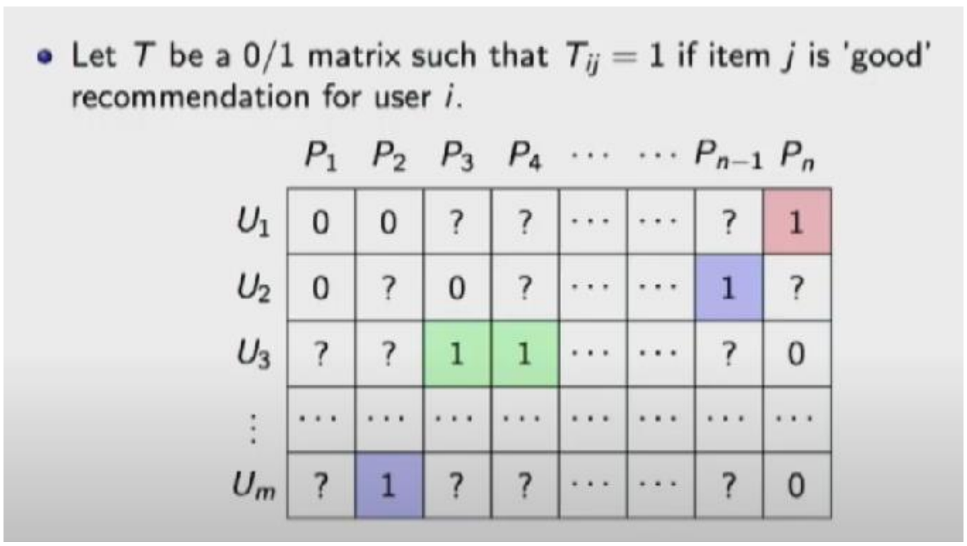
3. Perform phase estimation with precision parameter $2\varepsilon > 0$ on the input $|Qx>$ for the unitary $W = U \cdot V$ *[The matrices P, Q are a factorization of A, i.e. $A / ||A||_F = P^tQ$. Moreover, $P^tP = I_m$, $Q^tQ = I_n$, and multiplication by P, Q, i.e. the mappings $|y> \rightarrow |Py>$ and $|x> \rightarrow |Qx>$ can be performed in time O(polylog(mn)). The unitary $W = U \cdot V$, where U, V are the reflections $U = 2PP^t - I_{mn}$ and $V = 2QQ^t - I_{mn}$ can be implemented in time O(polylog(mn)).]* and obtain $\Sigma_i\, \alpha_i|Qv_i,\theta_{0,i}>$.

4. Compute $\sigma_{0,i} = \cos(\theta_{0,i} / 2)||A||_F$ where $\theta_{0,i}$ is the estimate from phase estimation, and uncompute the output of the phase estimation.

5. Apply the inverse of the transformation in step 2 to obtain $\Sigma_i\, \alpha_i|v_i>|\sigma_{0,i}>$.

It should be mentioned that the rank-k approximation $A_k = \Sigma_{i \in [k]}\, \sigma_i u_i v_i^t$ minimizes $||A - A_k||_F$.

Matrix sampling:


- Let T be a 0/1 matrix such that $T_{ij} = 1$ if item j is 'good' recommendation for user i.

	P_1	P_2	P_3	P_4	$\cdots$	$\cdots$	P_{n-1}	P_n
U_1	0	0	?	?	$\cdots$	$\cdots$	?	1
U_2	0	?	0	?	$\cdots$	$\cdots$	1	?
U_3	?	?	1	1	$\cdots$	$\cdots$	?	0
$\vdots$	$\cdots$	$\cdots$	$\cdots$	$\cdots$	$\cdots$	$\cdots$	$\cdots$	$\cdots$
U_m	?	1	?	?	$\cdots$	$\cdots$	?	0

Figure: 2

From the preference matrix we construct the binary matrix T shown above with a low rank approximation as before for reason similar to classical recommendation case. But as with the preference matrix we don’t really know all the entries of T. So, we can set all the ? Marks to 0 and will obtain a matrix T_0 that is a subsample of the ideal matrix T.

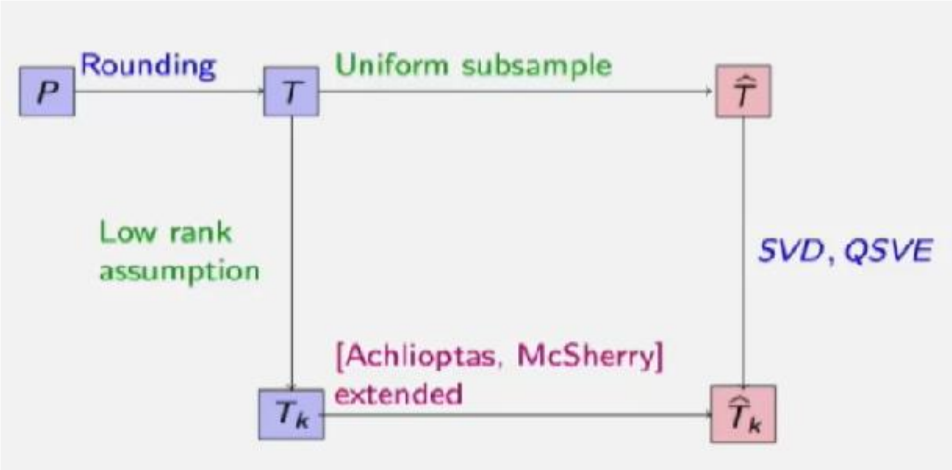


FIGURE: Matrix sampling based recommendation system.

Figure : 3

Figure 3 describes the matrix sampling based recommendation system. As discussed earlier, we create a matrix T for the preference matrix P by rounding the entries according to a certain threshold value, but many entries of T are unknown so what we have is a uniform subsample of matrix T using the technique discussed previously. In classical recommendation we use a low rank assumption on T to get T_k and results show that sampling for T_k gives us good results. In our case what we can do is to do singular value decomposition (SVD) in the classical case and use quantum singular value estimation (QSVE) in quantum case as shown earlier. Results show that it sufficies to sample from T_k - hat and gives us a good result.

<u>Main Algorithm:</u> The steps of the algorithm are as follows:

- Prepare state $|T_{0,i}>$ corresponding to user i.
- Apply quantum projection algorithm to $|T_{0,i}>$ to obtain $|(T_{\geq\sigma,k})_{0,i}>$ that refers to states with singular value greater than σ with some deviations.
- Measure quantum state in computational basis to obtain recommendation.

The threshold $\sigma = \epsilon\sqrt{p}\, ||T_0||_F / \sqrt{(2k)}$ and $\kappa = 1/3$. Running time depends on the threshold.

<u>The Projection Algorithm:</u>

Require: $A \in R^{m\times n}$, $x \in R^n$ in the data structure specified above; parameters $\sigma, \kappa > 0$.

1. Create $|x> = \Sigma_i\, \alpha_i\, |v_i>$.

2. Apply the singular value estimation on $|x>$ with precision $\epsilon = \kappa\sigma / 2||A||_F$ to obtain the state $\Sigma_i\, \alpha_i\, |v_i>\, |\sigma_{0,i}>$.

3. Apply on a second new register the unitary V that maps $|t\rangle |0\rangle \rightarrow |t\rangle |1\rangle$ if $t < \sigma - \kappa\sigma / 2$ and $|t\rangle |0\rangle \rightarrow |t\rangle |0\rangle$ otherwise, to get the state $\boldsymbol{\Sigma_{i \in S0}\ \alpha_i |v_i\rangle |\sigma_{0,i}\rangle |0\rangle + \Sigma_{i \in S1}\ \alpha_i |v_i\rangle |\sigma_i\rangle |1\rangle}$, where S0 is the union of all i's such that $\sigma_i \geq \sigma$ and some i's with $\sigma_i \in [(1 - \kappa)\sigma, \sigma)$.

4. Apply the singular value estimation on the above state to erase the second register $\boldsymbol{\Sigma_{i \in S0}\ \alpha_i |v_i\rangle |0\rangle + \Sigma_{i \in S1}\ \alpha_i |v_i\rangle |1\rangle = \beta |A^{+}_{\geq\sigma,\kappa} A_{\geq\sigma,\kappa}\, x\rangle |0\rangle + \sqrt{(1 - |\beta|^2)} |A^{+}_{\geq\sigma,\kappa} A_{\geq\sigma,\kappa}\, x\rangle^{\perp} |1\rangle}$, with $\beta = ||A^{+}_{\geq\sigma,\kappa} A_{\geq\sigma,\kappa}\, x|| \,/\, ||x||$.

5. Measure the second register in the standard basis. If the outcome is |0i, output the first register and exit. Otherwise repeat step 1.

This report is based on https://arxiv.org/abs/1603.08675