## CLAIMS

What is claimed is:

1. A data processing method, comprising:

adding an architecture parameter to each feature interaction item in a first model, to obtain a second model, wherein the first model is a factorization machine FM-based model, and the architecture parameter represents importance of a corresponding feature interaction item;

performing optimization on architecture parameters in the second model, to obtain the optimized architecture parameters; and

obtaining, based on the optimized architecture parameters and the first model or the second model, a third model through feature interaction item deletion comprising obtaining, based on the first model or the second model, the third model by deleting a feature interaction item corresponding to an architecture parameter in the optimized architecture parameters whose value is less than a threshold.

2. The method according to claim 1, wherein optimization allows the optimized architecture parameters to be sparse.

3. The method according to any one of claims 1 or 2, wherein optimization allows a value of an architecture parameter of at least one feature interaction item to be equal to zero after optimization is completed.

4. The method according to claim 3, wherein optimization is performed on the architecture parameters in the second model using a generalized regularized dual averaging gRDA optimizer, wherein the gRDA optimizer allows the value of the architecture parameter of the at least one feature interaction item to tend to zero during an optimization process.

5. The method according to any one of claims 1 to 4, wherein the method further comprises:

performing optimization on model parameters in the second model, wherein optimization comprises scalarization processing on the model parameters.

6. The method according to claim 5, wherein optimization comprises batch normalization BN processing on the model parameters in the second model.

7. The method according to any one of claim 5, wherein the performing optimization on architecture parameters in the second model and the performing optimization on model parameters in the second model comprise:

performing simultaneous optimization on both the architecture parameters and the model parameters in the second model by using same training data.

8. The method according to any one of claims 1 to 7, wherein the method further comprises: training the third model to obtain a click-through rate CTR prediction model or a conversion rate CVR prediction model.

9. A data processing method, comprising:

inputting data of a target object into a click-through rate CTR prediction model or a conversion rate CVR prediction model, to obtain a prediction result of the target object;

determining a recommendation result of the target object based on the prediction result of the target object, wherein

the CTR prediction model or the CVR prediction model is obtained through the following process:adding an architecture parameter to each feature interaction item in a first model, to obtain a second model, wherein the first model is a factorization machine FM-based model, and the architecture parameter represents importance of a corresponding feature interaction item;

performing optimization on architecture parameters in the second model, to obtain the optimized architecture parameters;

obtaining, based on the optimized architecture parameters and the first model or the second model, a third model through feature interaction item deletion comprising obtaining, based on the first model or the second model, the third model by deleting a feature interaction item corresponding to an architecture parameter in the optimized architecture parameters whose value is less than a threshold; and

training the third model by using a training sample of the target object, to obtain the CTR prediction model or the CVR prediction model.

10. The method according to claim 11, wherein optimization allows the optimized architecture parameters to be sparse.

11. The method according to any one of claims 9 to 10, wherein optimization allows a value of an architecture parameter of at least one feature interaction item to be equal to zero after optimization is completed.

12. The method according to claim 11, wherein optimization is performed on the architecture parameters in the second model using a generalized regularized dual averaging gRDA optimizer, wherein the gRDA optimizer allows the value of the architecture parameter of the at least one

feature interaction item to tend to zero during an optimization process.

13. The method according to any one of claims 9 to 12, wherein the process of obtaining the CTR prediction model or the CVR prediction model further comprises: performing optimization on model parameters in the second model, wherein optimization comprises scalarization processing on the model parameters.

14. The method according to claim 13, wherein optimization comprises batch normalization BN processing on the model parameters in the second model.

15. The method according to claim 13, wherein the performing optimization on architecture parameters in the second model and the performing optimization on model parameters in the second model comprise:

performing simultaneous optimization on both the architecture parameters and the model parameters in the second model by using the same training data as that in the training sample of the target object.

16. An image processing apparatus, comprising:

a memory, configured to store a program; and

a processor, configured to execute the program stored in the memory, wherein when the program stored in the memory is executed, the processor is configured to perform the method according to any one of claims 1 to 15.

17. A computer-readable storage medium, wherein the computer-readable medium stores program code to be executed by a device; and when the program code is executed, the device performs the method according to any one of claims 1 to 15.

18. A chip, comprising at least one processor and a data interface, wherein

the at least one processor is configured to invoke and run, by using the data interface, a computer program stored in a memory, to enable the chip to perform the method according to any one of claims 1 to 15.