CLAIMS

What is claimed is:

1. A method of building a privacy-preserving neural network model using at least one processor, the method comprising:

performing first neural network operations using a non-private neural network of the privacy-preserving neural network model based on first input data in plaintext to produce first output data in plaintext, wherein the non-private neural network is pre-trained and learnable parameters of the non-private neural network are fixed while performing the first neural network operations;

encrypting the first output data from the non-private neural network using a homomorphic encryption method to produce first encrypted data; and

performing second neural network operations homomorphically using a private neural network of the privacy-preserving neural network model based on the first encrypted data to produce second encrypted data, wherein learnable parameters of the private neural network are trained while performing the second neural network operations.

2. The method according to claim 1, wherein the non-private neural network comprises one or more first convolution layers and one or more first fully connected layers obtained from a pre-trained neural network, wherein the private neural network only comprises one or more second fully connected layers.

3. The method according to claim 1, wherein the non-private neural network only comprises one or more third convolution layers obtained from a pre-trained neural network, wherein the private neural network comprises one or more fourth convolution layers and one or more first fully connected layers obtained from the pre-trained neural network, and one or more second fully connected layers.

4. The method according to claim 3, wherein the one or more fourth convolution layers of the private neural network obtained from the pre-trained neural network are one or more lower

convolution layers of the pre-trained neural network with respect to the one or more third convolution layers obtained from the pre-trained neural network.

5. The method according to claim 4, wherein the performing second neural network operations homomorphically using a private neural network of the privacy-preserving neural network model based on the first encrypted data to produce second encrypted data comprises fine tuning the one or more fourth convolution layers of the private neural network obtained from the pre-trained neural network.

6. The method according to any one of claims 2 to 5, wherein the one or more second fully connected layers form a shallow fully connected network.

7. The method according to any one of claims 2 to 6, wherein the pre-trained neural network is an open-source pre-trained neural network.

8. The method according to any one of claims 1 to 7, wherein the first input data is labeled data from a training dataset.

9. The method according to claim 8, wherein the training dataset is a private dataset associated with a prediction task of interest and is different from a dataset associated with an original prediction task of interest on which the pre-trained neural network was trained.

10 The method according to any of claims 2 to 7, further comprising:

not transferring a last fully connected layer of the pre-trained neural network to the privacy-preserving neural network model.

11. The method according to any one of claims 1 to 10, wherein

said performing the second neural network operations comprises computing a plurality of non-linear functions, and

for each of the plurality of non-linear functions, the non-linear function is computed homomorphically using a look-up table (LUT) algorithm, wherein the LUT algorithm is configured to produce a polynomial having encoded therein a plurality of values of the non-

linear function and to produce an output ciphertext corresponding to one of the plurality of values based on the polynomial and an input ciphertext to the LUT algorithm.

12. The method according to claim 11, wherein the plurality of values of the non-linear function are encoded into coefficients of the polynomial.

13. The method according to claim 12, wherein the plurality of non-linear functions are a plurality of activation functions.

14. The method according to any one of claims 11 to 13, wherein

each fully connected layer of the private neural network comprises a plurality of nodes, and

for each fully connected layer of the private neural network, each of the plurality of nodes of the fully connected layer is configured to compute a respective non-linear function of the plurality of non-linear functions homomorphically using the LUT algorithm.

15. The method according to claim 14, wherein

said performing the second neural network operations further comprises computing a plurality of inner product functions homomorphically, and

for each fully connected layer of the private neural network, each of the plurality of nodes of the fully connected layer is configured to compute a respective inner product function of the plurality of inner product functions homomorphically based on an input ciphertext to the node and a weight matrix associated with the node to produce an output ciphertext, wherein the input ciphertext to the LUT algorithm for the node to compute the respective non-linear function homomorphically using the LUT algorithm corresponds to the output ciphertext of the respective inner product function.

16. The method according to claim 15, further comprising, for each of the plurality of nodes of the fully connected layer, controlling a size of the output ciphertext produced by the respective inner product function to have a predetermined number of bits.

17. The method according to any one of claims 11 to 16, wherein the LUT algorithm is configured to, for each of a plurality of polynomial multiplication operations therein, perform polynomial multiplication of polynomials in residue numeral system (RNS) form.

18. A system for building a privacy-preserving neural network model, the system comprising:

at least one memory; and

at least one processor communicatively coupled to the at least one memory and configured to perform the method of building a privacy-preserving neural network model according to any one of claims 1 to 17.

19. A computer program product, embodied in one or more non-transitory computer-readable storage mediums, comprising instructions executable by at least one processor to perform the method of building a privacy-preserving neural network model according to any one of claims 1 to 17.

20. A method of performing privacy-preserving prediction using a privacy-preserving neural network model, the method comprising:

performing, at a first system, first neural network operations using a non-private neural network of the privacy-preserving neural network model based on first input data in plaintext to produce first output data in plaintext;

encrypting, at the first system, the first output data from the non-private neural network using a homomorphic encryption method to produce first encrypted data; and

performing, at a second system, second neural network operations homomorphically using a private neural network of the privacy-preserving neural network model based on the first encrypted data transmitted from the first system and received by the second system to produce second encrypted data corresponding to an encrypted prediction result with respect to the first input data.

21. The method according to claim 20, wherein the non-private neural network comprises one or more first convolution layers obtained from a pre-trained neural network.

22. The method according to claim 21, wherein the non-private neural network further comprises one or more first fully connected layers obtained from the pre-trained neural network.

23. The method according to claim 22, wherein the private neural network comprises one or more second fully connected layers.

24. The method according to claim 21, wherein the private neural network comprises one or more second convolution layers and one or more first fully connected layers obtained from the pre-trained neural network, and one or more second fully connected layers.

25. The method according to claim 23 or 24, wherein the one or more second fully connected layers form a shallow fully connected network.

26. The method according to any one of claims 21 to 25, wherein the pre-trained neural network is an open-source pre-trained neural network.

27. The method according to any one of claims 20 to 26, wherein

said performing the second neural network operations comprises computing a plurality of non-linear functions, and

for each of the plurality of non-linear functions, the non-linear function is computed homomorphically using a look-up table (LUT) algorithm, wherein the LUT algorithm is configured to produce a polynomial having encoded therein a plurality of values of the non-linear function and to produce an output ciphertext corresponding to one of the plurality of values based on the polynomial and an input ciphertext to the LUT algorithm.

28. The method according to claim 27, wherein the plurality of values of the non-linear function are encoded into coefficients of the polynomial.

29. The method according to claim 28, wherein the plurality of non-linear functions are a plurality of activation functions.

30. The method according to any one of claims 27 to 29, wherein

each fully connected layer of the private neural network comprises a plurality of nodes, and

for each of one or more fully connected layers of the private neural network, each of the plurality of nodes of the fully connected layer is configured to compute a respective non-linear function of the plurality of non-linear functions homomorphically using the LUT algorithm.

31. The method according to claim 30, wherein

said performing the second neural network operations further comprises computing a plurality of inner product functions homomorphically, and

for said each of one or more fully connected layers of the private neural network, each of the plurality of nodes of the fully connected layer is configured to compute a respective inner product function of the plurality of inner product functions homomorphically based on an input ciphertext to the node and a weight matrix associated with the node to produce an output ciphertext, wherein the input ciphertext to the LUT algorithm for the node to compute the respective non-linear function homomorphically using the LUT algorithm corresponds to the output ciphertext of the respective inner product function.

32. The method according to claim 31, further comprising, for each of the plurality of nodes of the fully connected layer, controlling a size of the output ciphertext produced by the respective inner product function to have a predetermined number of bits.

33. The method according to any one of claims 27 to 32, wherein the LUT algorithm is configured to, for each of a plurality of polynomial multiplication operations therein, perform polynomial multiplication of polynomials in residue numeral system (RNS) form.

34. The method according to any one of claims 20 to 33, wherein the first system has stored therein the non-private neural network of the privacy-preserving neural network model and the second system has stored therein the private neural network of the privacy-preserving neural network model.

35. The method according to any one of claims 20-34, wherein the privacy-preserving neural network model is built according to the method of any one of claims 1 to 17.

36. A system for performing privacy-preserving prediction using a privacy-preserving neural network model, the system comprising:

a first system comprising:

at least one memory; and

at least one processor communicatively coupled to the at least one memory and configured to perform said first neural network operations and said encrypting the first output data of the method of performing privacy-preserving prediction according to any one of claims 20 to 35; and

a second system comprising:

at least one memory; and

at least one processor communicatively coupled to the at least one memory and configured to perform said second neural network operations homomorphically of the method of performing privacy-preserving prediction according to any one of claims 20 to 35.

37. A computer program product, embodied in one or more non-transitory computer-readable storage mediums, comprising instructions executable by at least one processor to perform the method of privacy-preserving prediction according to any one of claims 20 to 35.