(12) INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(19) World Intellectual Property Organization
International Bureau

(43) International Publication Date
15 October 2020 (15.10.2020)

WIPO | PCT

(10) International Publication Number
WO 2020/208555 A1

(51) International Patent Classification:
G16B 20/20 (2019.01) *G16B 20/30* (2019.01)

(21) International Application Number:
PCT/IB2020/053370

(22) International Filing Date:
08 April 2020 (08.04.2020)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:
62/831,663 09 April 2019 (09.04.2019) US

(71) Applicant: **ETH ZURICH** [CH/CH]; ETH transfer, HG E 43-49, Raemistrasse 101, 8092 Zurich (CH).

(72) Inventors: **MASON, Derek**; ETH transfer, HG E 43-49, Raemistrasse 101, 8092 Zurich (CH). **FRIEDENSOHN, Simon**; ETH transfer, HG E 43-49, Raemistrasse 101, 8092 Zurich (CH). **WEBER, Cedric**; ETH transfer, HG E 43-49, Raemistrasse 101, 8092 Zurich (CH). **REDDY, Sai**; ETH transfer, HG E 43-49, Raemistrasse 101, 8092 Zurich (CH).

(81) Designated States *(unless otherwise indicated, for every kind of national protection available)*: AE, AG, AL, AM, AO, AT, AU, AZ, BA, BB, BG, BH, BN, BR, BW, BY, BZ, CA, CH, CL, CN, CO, CR, CU, CZ, DE, DJ, DK, DM, DO, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IR, IS, JO, JP, KE, KG, KH, KN, KP, KR, KW, KZ, LA, LC, LK, LR, LS, LU, LY, MA, MD, ME, MG, MK, MN, MW, MX, MY, MZ, NA, NG, NI, NO, NZ, OM, PA, PE, PG, PH, PL, PT, QA, RO, RS, RU, RW, SA, SC, SD, SE, SG, SK, SL, ST, SV, SY, TH, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, WS, ZA, ZM, ZW.

(84) Designated States *(unless otherwise indicated, for every kind of regional protection available)*: ARIPO (BW, GH, GM, KE, LR, LS, MW, MZ, NA, RW, SD, SL, ST, SZ, TZ, UG, ZM, ZW), Eurasian (AM, AZ, BY, KG, KZ, RU, TJ,

(54) Title: SYSTEMS AND METHODS TO CLASSIFY ANTIBODIES

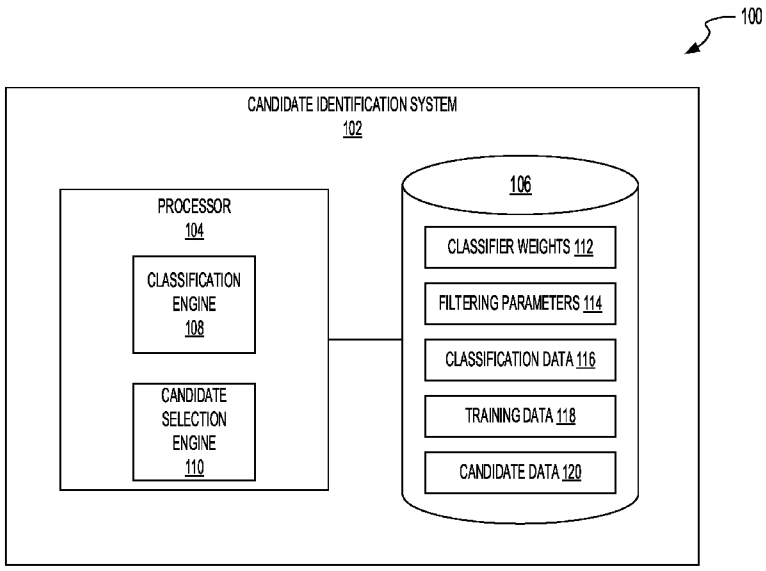


FIG. 1

(57) Abstract: The present disclosure describes systems and methods to make predictions classifying one or more properties of a binding protein such as an antibody, for example, antibody affinity or specificity for an antigen. The system can include one or more machine learning models that can extrapolate complex relationships between amino acid sequence and function. The system can be trained on high-quality training data generated through a two-step single-site and combinatorial deep mutational scanning approach. The trained models can then make predictions on novel variant sequences generated in silico. The present disclosure describes amino acid sequences generated by the systems and methods provided, and uses of the generated sequences to produce proteins for therapeutic and diagnostic use.

WO 2020/208555 A1

[Continued on next page]

TM), European (AL, AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HR, HU, IE, IS, IT, LT, LU, LV, MC, MK, MT, NL, NO, PL, PT, RO, RS, SE, SI, SK, SM, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, KM, ML, MR, NE, SN, TD, TG).

Published:

— *with international search report (Art. 21(3))*
— *before the expiration of the time limit for amending the claims and to be republished in the event of receipt of amendments (Rule 48.2(h))*
— *in black and white; the international application as filed contained color or greyscale and is available for download from PATENTSCOPE*

SYSTEMS AND METHODS TO CLASSIFY ANTIBODIES

CROSS-REFERENCE TO RELATED APPLICATIONS

[1] This application claims priority to U.S. Provisional Patent Application No. 62/831,663 filed April 9, 2019, which is incorporated herein by reference in its entirety.

BACKGROUND OF THE DISCLOSURE

[2] In antibody drug discovery, screening of phage or yeast display libraries is a standard practice for identifying therapeutic antibodies and can typically result in a number of potential lead variant candidates. However, the time and costs associated with lead candidate optimization often take up the majority of the drug preclinical discovery and development cycle. This is largely due to the fact that lead optimization of antibody molecules frequently includes addressing multiple parameters in parallel, including expression level, viscosity, pharmacokinetics, solubility, and immunogenicity. Once a lead candidate is discovered, additional engineering is often required. The fact that nearly all therapeutic antibodies require expression in mammalian cells as full-length IgG also means that the remaining development and optimization steps must occur in this context. Since mammalian cells lack the capability to replicate plasmids stably, this last stage of development is done at low-throughput, as elaborate cloning, transfection and purification strategies must be implemented to screen libraries in the max range of about 10^3 antibody molecules. This can result in only minor changes (e.g., single point mutations) being screened. Interrogating such a small fraction of protein sequence space also implies that addressing one development issue can frequently cause the rise of another or even diminish antigen binding altogether, making multi-parameter optimization challenging.

SUMMARY OF THE DISCLOSURE

[3] Provided herein are systems and methods for the classification of amino acid sequences of binding proteins, including, for example, an antibody that binds to an antigen or a receptor that binds to a ligand. In some embodiments, the methods provided herein combine directed evolution with machine learning to develop new proteins based on an input amino acid

sequence. In some embodiments, the methods provided can identify an amino acid sequence that improves one or more properties the binding protein, for example, an increase in the affinity or specificity of an antibody binding to an antigen, or two or more antigens (e.g., multispecific).

[4] According to at least one aspect of the disclosure, a method can include providing an input amino acid sequence that represents a portion of a binding protein. In some embodiments, the portion is an antigen binding portion of an antibody. In some embodiments, the portion affects one or more properties of the binding protein (e.g., antigen binding affinity). The method can include generating a first training data set comprising a first plurality of variant sequences. Each of the first plurality of sequences can include a single site mutation in the input amino acid sequence of the binding protein (e.g., an antibody). The method can include generating a second training data set comprising a second plurality of sequences. Each of the second plurality of sequences can include a plurality of variants at positions based on enrichment scores of the first training data set comprising the first plurality of sequences. The method can include providing the second training data set to a classification engine comprising a first machine learning model to generate a plurality of parameters for the first machine learning model. The method can include determining, by the classification engine based on the plurality of parameters for the first machine learning model, a first affinity binding score for a proposed amino acid sequence to an antigen. In some embodiments, the parameters comprise weights and biases for the first learning model. The method can include selecting the proposed amino acid sequence for further analysis and validation and/or expression based on the first affinity binding score satisfying a threshold. In some embodiments, further analysis and validation of the proposed amino acid sequence is based on one more parameters related to the developability and/or therapeutic potential of the proposed amino acid sequence.

[5] The method can include determining, by the classification engine, a second affinity binding score for the proposed amino acid sequence using a second machine learning model of the classification engine. The method can include selecting the proposed amino acid sequence for expression based on the first affinity binding score and the second affinity binding score

satisfying the threshold. The method can include determining, by the classification engine, an affinity binding score for each of a plurality of proposed amino acid sequences. The method can include determining, by a candidate selection engine, one or more parameters for each of the plurality of proposed amino acid sequences. The method can include selecting, by the candidate selection engine, candidate variants from the plurality of proposed amino acid sequences based on the affinity binding score and the one or more parameters for each of the plurality of proposed amino acid sequences. The one or more parameters can include protein sequence based metrics such as the Levenshtein distance value, charge value, hydrophobicity index value, CamSol score, minimum affinity rank, or average affinity ranking. The protein sequence based metrics can also include sequence motifs associated with manufacturing liabilities, such as n-glycosylation sites, deamidation sites, isomerization sites, methionine oxidation, tryptophan oxidation and paired or unpaired cysteine residues. The one or more parameters can also include protein structured based metrics such as the solvent accessible surface area (SASA), patches positive charges (PPC), patches negative charges (PNC), patches surface hydrophobicity (PSH) and surface Fv charge symmetry parameter (SFvCSP).

[6] The first machine learning model can include a recurrent neural network (RNN), a convolutional neural network (CNN), a standard artificial neural network (ANN), a support vector machine (SVM), a random forest ensemble (RF) or logistic regression (LR) model . The input amino acid sequence can be a portion of a complementarity determining region (CDR) of the antibody. The input amino acid sequence can be a CDRH1, CDRH2, CDRH3, CDRL1, CDRL2, CDRL3, a region within the framework domains of the antibody (e.g., FR1, FR2, FR3, FR4) or a region within the constant domains of the antibody (e.g., CH1, CH2, CH3), or any combination thereof, for which improvement of one or more properties of the antibody is desired. The input amino acid sequence can be a full length heavy chain or a full length light chain. The input amino acid sequence can be a recombinant sequence comprising one or more portion of an antibody. The antibody can be a therapeutic antibody. The first training data set can be generated by deep mutational scanning. The deep mutational scanning can include generating a first library of variant sequences wherein each variant sequence is modified at a single amino acid position relative to the input amino acid

sequence. The first library can include variant sequences representing each amino acid position of the input amino acid sequence.

[7] The first library can include variant sequences representing all 20 amino acids at each position of the input amino acid sequence. The first library of variant sequences can be generated by mutagenesis of the nucleic acid sequences encoding the input amino acid sequence. The first library of variant sequences can be generated by mutagenesis and introduction of the mutant sequences into a suitable expression system. The mutagenesis method can include any suitable method, such as error-prone PCR, recombination mutagenesis, alanine scanning mutagenesis, structure-guided mutagenesis, or homology-directed repair (HDR). The expression system can be, for example, a mammalian, yeast, bacteria, or phage expression system. The first library of variant sequences can be generated by high throughput mutagenesis in a mammalian cell. The first library of variant sequences can be generated by CRISPR/Cas9-mediated homology-directed repair (HDR). The deep mutational scanning can include generating a plurality of antibodies that can include the first library of variant sequences. The deep mutational scanning can include screening the plurality of antibodies and the first library of variant sequences for binding to an antigen and determining the sequence and frequency of variants selected for binding to the antigen, thereby obtaining the first training data set.

[8] The second training data set can be generated by deep mutational scanning-guided combinatorial mutagenesis. The deep mutational scanning-guided combinatorial mutagenesis can include generating a second library of variant sequences wherein each variant sequence is modified at two or more amino acid positions based on the first training data set. The second library of variant sequences can be generated by high throughput mutagenesis in a mammalian cell. The second library of variant sequences is generated by CRISPR/Cas9-mediated homology-directed repair (HDR). The deep mutational scanning-guided combinatorial mutagenesis can include generating a plurality of antibodies comprising the second library of variant sequences. The combinatorial deep mutational scanning can include screening the plurality of antibodies that can include the second library of variant sequences

for binding to the antigen and determining the sequence of variants selected for binding to the antigen, thereby obtaining the second training data set.

[9] Also provided herein are proteins or peptides comprising an amino acid sequence generated by the methods provided herein. In some embodiments, the generated amino acid sequence is a CDRH3. In some embodiments, the protein or peptide comprising an amino acid sequence generated herein is an antibody or fragment thereof. In some embodiments, the protein or peptide comprising an amino acid sequence generated herein is a full length antibody. In some embodiments, the protein or peptide comprising an amino acid sequence generated herein is a fusion protein comprising one or more portions of an antibody. In some embodiments, the protein or peptide comprising an amino acid sequence generated herein is an scFv or an Fc fusion protein. In some embodiments, the protein or peptide comprising an amino acid sequence generated herein is a chimeric antigen receptor. In some embodiments, the protein or peptide comprising an amino acid sequence generated herein is a recombinant protein. In some embodiments, the protein or peptide comprising an amino acid sequence generated herein binds to an antigen. In some embodiments, the antigen is associated with a disease or condition. In some embodiments, the antigen is a tumor antigen, an inflammatory antigen, pathogenic antigen (e.g., viral, bacterial, yeast, parasitic). In some embodiments, the protein or peptide comprising an amino acid sequence generated herein has one or more improved properties compared to a protein or peptide comprising the input amino acid sequence. In some embodiments, the protein or peptide comprising an amino acid sequence generated herein has improved affinity for an antigen compared to a protein or peptide comprising the input amino acid sequence. In some embodiments, the protein or peptide comprising an amino acid sequence generated herein has improved biophysical properties for manufacturing compared to a protein or peptide comprising the input amino acid sequence. In some embodiments, the protein or peptide comprising an amino acid sequence generated herein has reduced immunogenic risk compared to a protein or peptide comprising the input amino acid sequence. In some embodiments, the protein or peptide comprising an amino acid sequence generated herein can be administered to treat an inflammatory disease, infectious disease, cancer, genetic disorder, organ transplant rejection, autoimmune disease or an immunological disorder. In some embodiments, the protein or peptide comprising an amino

acid sequence generated herein can be used for the manufacture of a medicament to treat an inflammatory disease, infectious disease, cancer, genetic disorder, organ transplant rejection, autoimmune disease and immunological disorder. Also provided herein are cells comprising one more proteins or peptides comprising an amino acid sequence generated herein. The cell can be a mammalian cell, a bacterial cell, a yeast cell or any cell that can express a protein or peptide comprising an amino acid sequence generated herein. The cell can be an immune cell, such as a T cell (e.g., a cell used in Chimeric Antigen Receptor (CAR) T-cell therapy). In some embodiments, the protein or peptide comprising an amino acid sequence generated herein can be used to detect an antigen in a biological sample.

[10] Also provided herein are proteins or peptides comprising an amino acid sequence shown any of FIGS. 15A-D, 23 A-O. In some embodiments, the amino acid sequence shown any of FIGS. 15A-D, 23 A-O is a CDRH3. In some embodiments, the protein or peptide comprising an amino acid sequence shown any of FIGS. 15A-D, 23 A-O is an antibody or fragment thereof. In some embodiments, the protein or peptide comprising an amino acid sequence shown any of FIGS. 15A-D, 23 A-O is a full length antibody. In some embodiments, the protein or peptide comprising an amino acid sequence shown any of FIGS. 15A-D, 23 A-O is a fusion protein comprising one or more portions of an antibody. In some embodiments, the protein or peptide comprising an amino acid sequence shown any of FIGS. 15A-D, 23 A-O is an scFv or an Fc fusion protein. In some embodiments, the protein or peptide comprising an amino acid sequence shown any of FIGS. 15A-D, 23 A-O is a chimeric antigen receptor. In some embodiments, the protein or peptide comprising an amino acid sequence shown any of FIGS. 15A-D, 23 A-O is a recombinant protein. In some embodiments, the protein or peptide comprising an amino acid sequence shown any of FIGS. 15A-D, 23 A-O binds to the HER2 (human epidermal growth factor receptor 2) antigen. In some embodiments, the protein or peptide comprising an amino acid sequence shown any of FIGS. 15A-D, 23 A-O has one or more improved properties compared to the trastuzumab (Herceptin) antibody. In some embodiments, the protein or peptide comprising an amino acid sequence shown any of FIGS. 15A-D, 23 A-O has improved affinity for the HER2 antigen compared to the trastuzumab (Herceptin) antibody. In some embodiments, the protein or peptide comprising an amino acid sequence shown any of FIGS. 15A-D, 23 A-O

can be administered to treat a HER2 positive cancer. In some embodiments, the protein or peptide comprising an amino acid sequence shown any of FIGS. 15A-D, 23 A-O can be administered to treat a HER2 positive breast cancer. In some embodiments, the protein or peptide comprising an amino acid sequence shown any of FIGS. 15A-D, 23 A-O can be used for the manufacture of a medicament to treat a HER2 positive breast cancer. In some embodiments, the HER2 positive cancer is a metastatic cancer. Also provided herein are cells comprising one more proteins or peptides comprising an amino acid sequence shown any of FIGS. 15A-D, 23 A-O. The cell can be a mammalian cell, a bacterial cell, a yeast cell or any cell that can express a protein or peptide comprising an amino acid sequence shown any of FIGS. 15A-D, 23 A-O. The cell can be an immune cell, such as a T cell (e.g., a CAR-T cell). In some embodiments, the protein or peptide comprising an amino acid sequence shown any of FIGS. 15A-D, 23 A-O can be used to detect a HER2 antigen in a biological sample.

[11] The foregoing general description and the following description of the drawings and detailed description are exemplary and explanatory and are intended to provide further explanation of the invention as claimed. Other objects, advantages, and novel features will be readily apparent to those skilled in the art from the following brief description of the drawings and detailed description.

BRIEF DESCRIPTION OF THE DRAWINGS

[12] The accompanying drawings are not intended to be drawn to scale. Like reference numbers and designations in the various drawings indicate like elements. For purposes of clarity, not every component may be labeled in every drawing. In the drawings:

[13] FIG. 1 illustrates a block diagram of an example system to select antibody candidates.

[14] FIG. 2A illustrates an example neural network that can be used with the example system illustrated in FIG. 1.

[15] FIG. 2B illustrates an example receiver operating characteristic.

[16] FIG. 3A illustrates another example neural network that can be used with the example system illustrated in FIG. 1.

[17] FIG. 3B illustrates an example receiver operating characteristic.

[18] FIG. 4A illustrates an example flow process for generating training data that can be used with the example system illustrated in FIG. 1.

[19] FIG. 4B illustrates an example flow process for selecting candidate variants using the example system illustrated in FIG. 1.

[20] FIG. 5A illustrates (A) the Trastuzumab (Herceptin) CDRH3 variant sequence and (B) Flow cytometry profile following the integration of tiled mutations by homology-directed mutagenesis.

[21] FIG. 5B illustrates antigen-specific variants that underwent 3 rounds of enrichment (C) Corresponding heatmap following sequencing analysis of the pre-sorted (Ab+) and post-sorted (Ag+) populations. Black circles mark wild type amino acids. (D) The resulting sequence logo plot generated by positively enriched mutations per position.

[22] FIG. 5C illustrates (E) 3D protein structure of trastuzumab in complex with its target antigen, HER2 (Cho et al. (2003) *Nature* 421 (6924): 756–60). Locations of surface-exposed amino acid positions: 102D, 103G, 104F, and 105Y are provided.

[23] FIG. 6A illustrates (A) Sequence logo plot and (B) Flow cytometry plots resulting from transfection of a rationally designed library. Two rounds of enrichment were performed to produce a library of antigen-specific variants.

[24] FIG. 6B illustrates how next-generation sequencing was performed on the library (Ab+), non-binding variants (Ag-), and binding variants after 1 and 2 rounds of enrichment (Ag+1, Ag+2) (C, D) Amino acid frequency plots of (C) antigen binding variants and (D) non-binding variants reveals nearly indistinguishable amino acid usages across all positions.

[25] FIGS. 7A-E illustrate an example filtering policy that can be used with the example system illustrated in FIG. 1. Histograms show the parameters distributions of all predicted variants at the different stages of filtering. FIG. 7A illustrates (A) Levenshtein distance from wild-type trastuzumab;and (B) Net charge of the VH domain. FIG. 7B illustrates (C) CDRH3 hydrophobicity index; and (D) CamSol intrinsic solubility score. FIG. 7C illustrates (E) Minimum NetMHCIIpan % Rank across all possible 15-mers; and (F) Average NetMHCIIpan % Rank across all possible 15-mers. FIG. 7D illustrates (G) count numbers for sequences with various average netMHC scores; and (H) overall developability scores for experimental and predicted binders. FIG. 7E illustrates (I) filtering parameters and the number of sequences at the corresponding stage of filtering

[26] FIG. 8 illustrates a block diagram of an example method to identify antibodies with an antigen affinity using the example system illustrated in FIG. 1.

[27] FIGS. 9A-9B illustrate the Trastuzumab (Herceptin) CDRH3 variant and CDRH3 sequence and flow cytometry data following transfection of the hybridoma cells with either gRNA only (bottom left panel), gRNA+DMS ssODN library (bottom middle panel), or gRNA+DMS-combinatorial mutagenesis library (bottom right panel). The top middle panel is a representative flow cytometry plot of the Trastuzumab CDRH3 variant prior to transfection.

[28] FIG. 10 illustrates exemplary flow cytometry data for Trastuzumab (Herceptin) CDRH3 deep mutation scanning. (A) Flow cytometry plot, heatmap, and sequence logo plot following FACS for antibody expressing (Ab+) cells and antigen-specific (Ag+) cells. (B) Flow cytometry plot, heatmap, and sequence logo plot following a second round of enrichment for antigen-specific (Ag+2) cells; Decreased antigen concentration was used for flow cytometry labeling. (C) Flow cytometry plot, heatmap, and sequence logo plot following a third round of enrichment for antigen-specific (Ag+3) cells; Labeling for flow cytometry performed with antigen containing an alternatively conjugated fluorophore (Alexa Fluor 488). All enrichment ratios (ER) are calculated by dividing the frequency of a mutant

found in the respective Ag+ population by the frequency of the mutant found in the Ab+ population.

[29] FIG. 11 illustrates exemplary workflow and flow cytometry data for generating antigen specific libraries in mammalian cells. Libraries are generated by transfecting gRNA and ssODN donor templates containing rationally designed libraries. Antibody expressing cells (Ab+) are enriched by magnetic activated cell sorting (MACS). Ab+ cells can then undergo multiple rounds of enrichment for antigen-specific variants. Antigen-specific libraries are designed from enrichment ratios calculated following sequential rounds of antigen enrichment during DMS studies. (A) Libraries designed from DMS data following one round of antigen enrichment (Ag+, FIG. 10A). (B) Libraries designed from DMS data following two rounds of antigen enrichment (Ag+2, FIG. 10B). (C) Libraries designed from DMS data following three rounds of antigen enrichment (Ag+3, FIG. 10C).

[30] FIG. 12 illustrates the exemplary next-generation sequencing results for sequence reads, alignment, and number of unique sequences detected for NGS performed on the library (Ab+), non-binding variants (Ag-), and binding variants after 1 and 2 rounds of enrichment (Ag+1, Ag+2).

[31] FIGS. 13A and 13B illustrate the exemplary next-generation sequencing results for sequence reads, alignment, and number of unique sequences detected for NGS performed on the combinatorial mutagenesis libraries.

[32] FIGS. 14A and 14B illustrate exemplary flow cytometry data for Trastuzumab (Herceptin) CDRH3 DMS-based combinatorial mutagenesis libraries. Following transfection and integration of the DMS-based combinatorial mutagenesis library, the frequency of antigen-specific variants can be used to assist in model performance and evaluation. In the example provided, approximately 10% of antibody variants are antigen-specific.

[33] FIGS. 15A to FIG. 15D illustrate experimental validation data for 104 variants obtained by in silico selection.

[34] FIGS. 16A-D illustrate experimental validation data for antibody sequences predicted according to the methods disclosed therein. FIG. 16A depicts protein expression levels for various predicted antibody sequences as compared to expression levels of trastuzumab (farthest right). FIG. 16B depicts binding kinetics of the predicted antibody sequences. The binding kinetics of trastuzumab is indicated in the nanomolar range. FIG. 16C depicts thermal stability of the predicted antibody sequences as compared to thermal stability of trastuzumab (farthest right). FIG. 16D depicts immunogenicity risk of two predicted sequences (C and F) as compared to trastuzumab.

[35] FIGS. 17A–21B illustrate model performance curves for classification of binders and non-binders on unseen test data. 30% of the initial data set was split into two test data sets (15% each). One test data set contains the same ratio of binding and non-binding sequences present in the training data set (TEST SET A) and the other test data set contains an approximate ratio of 10/90 binding and non-binding sequences (TEST SET B) to resemble physiological frequencies observed in the data illustrated in FIGS. 14A-B. (Top panels) ROC (receiver operating character) curve and PR (precision-recall) curve observed on the classification of sequences in TEST SET A; (Bottom panels) ROC curve and PR curve observed on the classification of sequence in TEST SET B; (A) LSTM-RNN (Long-short term memory recurrent neural network) ROC curves (left panels), LSTM-RNN PR curves (right panels); (B) CNN (convolutional neural network) ROC curves (left panels), CNN PR curves (right panels)

[36] FIG. 22 provides a summary of the AUC (area under the curve), average PR and the number of predicted binders for each of the model performance curves shown in FIGS. 17-21.

[37] FIGS. 23A–23O illustrate exemplary data for the flow cytometry analysis (left) and biolayer interferometry affinity analysis (right) for the tested variants.

[38] FIG. 24A illustrates a table of flow cytometry labeling conditions for deep mutational scanning studies.

[39] FIG. 24B illustrates flow cytometry labeling conditions for DMS-guided combinatorial mutagenesis libraries.

[40] FIG. 25 illustrates exemplary flow cytometry data for Trastuzumab (Herceptin) CDRL3 deep mutation scanning. (A) Flow cytometry plot, heatmap, and sequence logo plot following FACS for antibody expressing (Ab+) cells and antigen-specific (Ag+) cells. (B) Flow cytometry plot, heatmap, and sequence logo plot following a second round of enrichment for antigen-specific (Ag+2) cells; Decreased antigen concentration was used for flow cytometry labeling. (C) Flow cytometry plot, heatmap, and sequence logo plot following a third round of enrichment for antigen-specific (Ag+3) cells; Labeling for flow cytometry performed with antigen containing an alternatively conjugated fluorophore (Alexa Fluor 488). All enrichment ratios (ER) are calculated by dividing the frequency of a mutant found in the respective Ag+ population by the frequency of the mutant found in the Ab+ population.

[41] FIG. 26 illustrates exemplary next-generation sequencing results for sequence reads, alignment, and number of unique sequences detected from NGS performed on the CDRL3 library (Ab+) and binding variants after 1 and 2 rounds of enrichment (Ag+1, Ag+2).

[42] FIG. 27 illustrates exemplary workflow and flow cytometry data for generating antigen specific libraries in mammalian cells at multiple locations along the antibody (e.g. CDRL3 and CDRH3). Initial libraries are generated by transfecting gRNA and ssODN donor templates containing rationally designed libraries for the first region. Antibody expressing cells (Ab+) are enriched by fluorescence activated cell sorting (FACS). Libraries in the second region are then generated by transfecting gRNA and ssODN donor templates containing rationally designed libraries for the second region. Antibody expressing cells (Ab+) are enriched by fluorescence activated cell sorting (FACS). Ab+ cells can then undergo multiple rounds of enrichment for antigen-specific variants. Antigen-specific libraries are designed from enrichment ratios calculated following sequential rounds of antigen enrichment during DMS studies. (A) CDRL3 libraries designed from DMS data following two rounds of antigen enrichment (Ag+2, FIG. 25C). (B) CDRH3 libraries

designed from DMS data following two rounds of antigen enrichment (Ag+3, FIG. 10C). (C-D) Experimental results from sanger sequencing experiments derived from the final CDRL3+CDRH3 mutagenesis library validating genetic diversity introduced into both regions. (E) illustrates exemplary workflow and flow cytometry data for generating antigen specific libraries first at CDRL3 and then at CDRH3.

[43] FIG. 28 illustrates exemplary data for Adalimumab (Humira) CDRH3 deep mutation scanning. Heatmap and sequence logo plot generated from deep sequencing of libraries following FACS for antibody expressing (Ab+) cells and antigen-specific (Ag+) cells; Labeling for flow cytometry performed with antigen containing an alternatively conjugated fluorophore (Alexa Fluor 488).

[44] FIG. 29 illustrates exemplary next-generation sequencing results for sequence reads, alignment, and number of unique sequences detected from NGS performed on the adalimumab CDRH3 library (Ab+) and binding variants after 1 and 2 rounds of enrichment (Ag+1, Ag+2).

DETAILED DESCRIPTION

[45] The various concepts introduced above and discussed in greater detail below may be implemented in any of numerous ways, as the described concepts are not limited to any particular manner of implementation. Examples of specific implementations and applications are provided primarily for illustrative purposes.

[46] Phage and yeast display screening are useful for high-throughput screening of large mutagenesis libraries ($>10^9$), however they are primarily used for only increasing affinity or specificity to the target antigen. Nearly all therapeutic antibodies can require expression in mammalian cells as full-length IgG, which means that the development and optimization steps following initial selection must occur in this context. Since mammalian cells lack the capability to stably replicate plasmids, this last stage of development is done at very low-throughput, as elaborate cloning, transfection and purification strategies must be implemented to screen libraries in the max range of 10^3 antibodies. Thus, only minor changes

(e.g., point mutations) are screened at this stage, typically resulting in only a few optimized leads. Interrogating such a small fraction of protein sequence space also implies that addressing one development issue will frequently cause rise of another or even diminish antigen binding altogether, making multi-parameter optimization very challenging.

[47] The methods described herein include an improved therapeutic antibody development process that employs an effective combination of directed evolution from rationally designed mutagenesis libraries with machine learning. Deep learning models to interrogate and predict antigen-specificity from a massive diversity of antibody sequence space enables the generation of thousands of optimized lead candidates.

[48] In some aspects, a mammalian display platform is used, where rationally designed site-directed mutagenesis libraries are introduced using high throughput mutagenesis systems for mammalian expression, such as by CRISPR/Cas9-mediated homology-directed repair (HDR). The inventors have found that screening and deep sequencing of relatively small libraries (e.g., about 10^4) generated based on the described methods, produced high quality data capable of training deep neural networks that predict antigen-binding based on antibody sequence with over 80% precision.

[49] Once trained according to the methods described herein, machine learning models can then be used to predict millions of antigen binders from a much larger *in silico* generated library variants (e.g., ~10^8 variants were generated by the methods described herein when trastuzumab was used as an input amino acid sequence). These variants can be subjected to multiple developability filters, resulting in tens of thousands of optimized lead candidates. As described herein in the Examples, when the present methods were applied to the heavy chain complementarity determining region 3 (CDRH3) of an exemplary antibody, the therapeutic antibody Trastuzumab, it was observed that of the small subset of only 30 optimized lead candidates that were expressed and assayed for antigen binding, 29 were shown to be antigen-specific. Thus, nearly all optimized lead candidates that were selected for testing possessed the predicted property. With its scalable throughput and capacity to interrogate across a vast protein sequence space, the methods described herein can be applied to a wide

variety of applications that involve the engineering and optimization of antibody and other protein-based therapeutics.

[50] The present disclosure describes systems and methods to make predictions of protein sequence-phenotype relationships and can be employed for the identification of therapeutic antibodies with one or more desired parameters, such as antigen specificity or affinity. The system can include one or more machine learning models that can extrapolate complex relationships between protein sequence and function. In some aspects, the models can be trained on high-quality training data generated through a two-step directed evolution approach, that combines single-site mutagenesis scanning followed by a combinatorial deep mutational scanning approach. The trained models described herein can then make predictions regarding new antibody sequences generated *in silico*. The systems and methods described herein enable the interrogation of a much larger sequence space than what is physically possible with standard expression systems, such as phage or bacterial display. For example, for a short stretch of 10 amino acids, the combinatorial sequence diversity explodes to 10^{13}, a size which is nearly impossible to interrogate experimentally. In some aspects, the systems described herein can also perform multi-parameter optimization to identify, from the variants classified by the models as antigen-binders, the antigen-binder classified variants that are most likely to exhibit antigen-specificity.

[51] FIG. 1 illustrates a block diagram of an example system 100 to select antibody lead candidates. The candidate identification system 102 can include one or more processors 104 and one or more memories 106. The processors 104 can execute processor-executable instructions to perform the functions described herein. The processor 104 can execute a classification engine 108 and a candidate selection engine 110. The memory 106 can store processor-executable instructions, generate data, and collected data. The memory 106 can store one or more classifier weights 112 and filtering parameters 114. The memory 106 can also store classification data 116, training data 118, and candidate data 120.

[52] The system 100 can include one or more candidate identification systems 102. The candidate identification system 102 can include at least one logic device, such as the

processors 104. The candidate identification system 102 can include at least one memory element 106, which can store data and processor-executable instructions. The candidate identification system 102 can include a plurality of computing resources or servers located in at least one data center. The candidate identification system 102 can include multiple, logically-grouped servers and facilitate distributed computing techniques. The logical group of servers may be referred to as a data center, server farm, or a machine farm. The servers can also be geographically dispersed. The candidate identification system 102 can be any computing device. For example, the candidate identification system 102 can be or can include one or more laptops, desktops, tablets, smartphones, portable computers, or any combination thereof.

[53] The candidate identification system 102 can include one or more processors 104. The processor 104 can provide information processing capabilities to the candidate identification system 102. The processor 104 can include one or more of digital processors, analog processors, digital circuits to process information, an analog circuit designed to process information, a state machine, and/or other mechanisms for electronically processing information. Each processor 104 can include a plurality of processing units or processing cores. The processor 104 can be electrically coupled with the memory 106 and can execute the classification engine 108 and the candidate selection engine 110.

[54] The processor 104 can include one or more microprocessors, application-specific integrated circuits (ASIC), field-programmable gate arrays (FPGA), or combinations thereof. The processor 104 can be an analog processor and can include one or more resistive networks. The resistive network can include a plurality of inputs and a plurality of outputs. Each of the plurality of inputs and each of the plurality of outputs can be coupled with nanowires. The nanowires of the inputs can be coupled with the nanowires of the outputs via memory elements. The memory elements can include ReRAM, memristors, or PCM. The processor 104, as an analog processor, can use analog signals to perform matrix-vector multiplication.

[55] The candidate identification system 102 can include one or more classification engines 108. The classification engine 108 can include one or more machine learning algorithms configured to extract features from data and classify the data based on the extracted features. For example, the classification engine 108 can include one or more of a recurrent neural network (e.g., a type of artificial neural network derived from feedforward neural networks in which connections between nodes form a directed graph along a temporal sequence to allow for temporal dynamic behavior), a convolutional neural network (e.g., a neural network with layers of nodes that are connected to one-another and use convolution in at least one of the layers), a standard artificial neural network (e.g., a computing system based on a collection of connected units or nodes configured to learn to perform tasks based on examples or training data), a support vector machine (e.g., a supervised learning model with associated learning functions that analyze data used for classification and regression analysis), a random forest ensemble (e.g., a computing system learning method for classification, regression and other tasks that operate by constructing a multitude of decision trees at training time and outputting the class is the mode of the classes or mean prediction of the individual trees) or a logistic regression model (e.g., a statistical technique that can use a logistic function to model the probability of a certain class or event existing such as a binary dependent variable).

[56] For example, the classification engine 108 can include an artificial neural network. The neural network can include an input layer, a plurality of hidden layers, and an output layer. The neural network can be a multi-layered neural network, a convolution neural network, or a recurrent neural network, including a long-short-term-memory (LSTM) neural network. The classification engine 108 can include a plurality of neural networks or classification models. For example, the classification engine 108 can process the classification data 116 with a first classification model (e.g., a convolution neural network) and also with a second classification model (e.g., an LSTM neural network). As described below in relation to the candidate selection engine 110, the candidate selection engine 110 can select candidate antibodies as the antibodies that were identified by the first and second classification model.

[57] During a training phase, the classification engine 108 can process training data 118 to generate the weights and biases for one or more of the classification engine's machine

learning models. Once trained, the classification engine 108 can store the weights and biases as the classifier weights 112 in the memory 106. The generation of the training data and training of the classification engine 108 is described further in relation to the memory 106, training data 118, and examples, below.

[58] The classification engine 108 can generate the weights and biases by inputting the training data 118 into the neural network and comparing the resulting classification to the expected classification (as defined by the input data's label). For example, in an example system that includes 10 output neurons that each correspond to a different classification, the classification engine 108 can use back-propagation and gradient descent to minimize the cost or error between the expected result and result determined by the classification engine 108. Once the classification engine 108 has trained its neural network, the classification engine 108 can save the weights and biases to the memory 106 as classifier weights 112. The models (e.g., the convolution neural network and the LSTM neural network) of the classification engine 108 are described further in relation to FIGS. 2 and 3, among others.

[59] The candidate identification system 102 can include a candidate selection engine 110. For a given protein sequence space (e.g., all possible protein sequence variants), the classification engine 108 can classify a large number of the variants as antigen-binders. The candidate selection engine 110 can select candidate variants from the variants classified as antigen-binders for further testing or study. The candidate selection engine 110 can select the candidate variants by applying one or more filtering policies to the antigen-binder classified variants. The filtering policies can include one or more filtering parameters 114, each with an associated threshold or other constraint. The candidate selection engine 110 can select the antigen-binder classified variants as candidate variants if the antigen-binder classified variant satisfies the, for example, threshold of the respective filtering parameters 114.

[60] The candidate selection engine 110 can select an antigen-binder classified variant as a candidate variant if more than one model of the classification engine 108 classifies the variant as an antigen-binder. For example, the classification engine 108 can include a convolution neural network and an LSTM neural network. The classification engine 108 can

classify each of the variants in the variant space with the convolution neural network and the LSTM neural network to generate two classifications for each variant (e.g., one classification by the convolution neural network and a second classification by the LSTM neural network). When the classification engine 108 performs classification with multiple models to generate multiple classifications for each variant. A consensus between the models can be one of the filtering parameters 114. For example, variants not classified as antigen-binder classified variants by both the convolution neural network and the LSTM neural network can be discarded from further processing. The candidate data 120 can include variants that are classified as antigen-binder classified variants by both the convolution neural network and the LSTM neural network.

[61] The filtering parameters 114 can include a similarity metric requirement to a known wild-type antibody sequence. For example, the candidate selection engine 110 can calculate a Levenshtein distance between each variant in the variant space and the known wild-type sequence to determine a similarity between the respective variant and the wild-type sequence. The filtering policy can indicate that each candidate variant must satisfy a similarity threshold with the wild-type sequence. For example, the candidate selection engine 110 can select antigen-binder classified variants as candidate variants for storage in the candidate data 120 if the antigen-binder classified variants have a Levenshtein distance less than 5, for example. The candidate selection engine 110 can select antigen-binder classified variants that have a Levenshtein distance greater than 5 in some examples.

[62] The filtering parameters 114 can include a similarity metric to human antibody repertoire sequences. For example, the candidate selection engine 110 can calculate a Levenshtein distance between each variant in the variant space to a collection of human antibody sequences (e.g., from patient B cells) to determine a similarity between the respective variant and the human repertoire. Based on filtering policy, the candidate selection engine 110 can select candidate variants that satisfy a similarity threshold to human repertoire sequences.

[63] The filtering parameters 114 can include any developability attribute of a protein, including, for example, a net charge, hydrophobicity index, viscosity, clearance threshold,

solubility, affinity, chemical stability, thermal stability, expressability, specificity, cross-reactivity, or any combination thereof. The candidate selection engine 110 can calculate, for each antigen-binder classified variant, the net change and the hydrophobicity of the antigen-binder classified variant. Based on the net charge and the hydrophobicity, the candidate selection engine 110 can calculate a viscosity value and clearance value for the antigen-binder classified variant. For example, viscosity can decrease with increasing variable fragment (Fv) net charge and increasing Fv charge symmetry parameter (FvCSP). The filtering parameters 114 can include a clearance value based on the variable fragment (Fv) charge between about 0 and about 6.2 with a CDRL1+CDRL3+CDRH3 hydrophobicity index sum less than 4.0. The candidate selection engine 110 can identify protein sequence motifs associated with manufacturing liabilities, such as n-glycosylation sites, deamidation sites, isomerization sites, methionine oxidation, tryptophan oxidation and paired or unpaired cysteine residues. For example, the candidate selection engine 110 can select antigen-binder classified variants with zero sequence motifs associated with manufacturing liabilities. The candidate selection engine 110 can include a protein solubility predictor to predict a protein solubility for each of the antigen-binder classified variants. For example, the candidate selection engine 110 can select antigen-binder classified variants with a solubility greater than 1 as candidate variants. In some implementations, the candidate selection engine 110 can select the antigen-binder classified variants with a solubility or other developability attribute above a threshold. The threshold can be a value threshold. The threshold can be a variable or relative threshold. For example, the threshold can be the top 5%, 10%, or other percentage of the antigen-binder classified variants. In another example, the candidate selection engine 110 can select antigen-binder classified variants above a number of standard deviations above the average.

[64] The candidate selection engine 110 can calculate an affinity binding score for each of the antigen-binder classified variants for MHC Class II molecules in order to filter out candidate peptides that may be immunogenic. For example, the candidate selection engine 110 can predict the peptide binding affinity of the variant sequences to MHC Class II molecules by utilizing a tool, such as NetMHCIIpan, which predicts binding of peptides to the three human MHC class II isotypes HLA-DR, HLA-DP and HLA-DQ. The CDRH3 sequences can be

padded with 10 amino acids on the 5' and 3' ends and then all possible 15-mers can be run through NetMHCIIpan. The candidate selection engine 110 can determine an antigen-binder classified variant's percentage rank predicted affinity for MHC Class II compared to a set of 200,000 random natural peptides. The candidate selection engine 110 can filter out antigen-binder classified variants with a percentage rank less than about 20%, 15%, 10%, 5%, or 2%. The lower the percentage rank, the higher the predicted affinity of the antigen-binder classified variant for MHC Class II. In some aspects, sequences can be filtered out if any of the 15-mers contain a % Rank <15. The average % Rank across all 15-mers for the remaining sequences can further be calculated and those with an average % Rank <70 can be filtered out. The mean and median values for the predicted binding affinity can further be calculated across all MHC class II alleles for each of the 15-mers and those sequences with a mean and/or median greater than a defined threshold can be filtered out. The filtering policy can indicate that an antigen-binder classified variant must satisfy one or more of the filtering parameters 114 to be selected as a candidate variant and be stored as candidate data 120.

[65] The candidate identification system 102 can include one or more memories 106. The memory 106 can be or can include a memory element. The memory 106 can store machine instructions that, when executed by the processor 104 can cause the processor 104 to perform one or more of the operations described herein. The memory 106 can include but is not limited to, electronic, optical, magnetic, or any other storage devices capable of providing the processor 104 with instructions. The memory 106 can include a floppy disk, CD-ROM, DVD, magnetic disk, memory chip, ROM, RAM, EEPROM, EPROM, flash memory, optical media, or any other suitable memory from which the processor 104 can read instructions. The instructions can include code from any suitable computer programming language such as, but not limited to, C, C++, C#, Java, JavaScript, Perl, HTML, XML, Python, and Visual Basic.

[66] The candidate identification system 102 can store classifier weights 112 in the memory 106. The classifier weights 112 can be a data structure that includes the weights and biases that define the neural networks of the classification engine 108. Once trained, the classification engine 108 can store the classifier weights 112 to the memory 106 for later retrieval and use in classifying classification data 116.

[67] The candidate data 120 can store filtering parameters 114 in the memory 106. As described above, the candidate selection engine 110 can retrieve a filtering policy for selecting candidate variants from the antigen-binder classified variants. The candidate selection engine 110 can apply the filtering policy to identify antigen-binder classified variants that have a higher likelihood of having a relatively high affinity for a given antigen. The filtering parameters 114 can each be a data structure that indicates a threshold value for the respective filtering parameter 114. For example, a filtering parameter can indicate that antibody for a given antigen-binder classified variant should have a Fv net charge between about 0 and about 6. Each filtering parameter 114 can indicate a specific parameter and predetermined threshold (e.g., above 2), a predetermined range (e.g., between 0 and 6), an adaptive threshold (e.g., having a predicted affinity within the top 5% of the antigen-binder classified variants), or an adaptive range (e.g., between about the top 1% and 5% of predicted affinities for the antigen-binder classified variants).

[68] The candidate identification system 102 can store classification data 116 in the memory 106. The classification data 116 can be a plurality of variants that are to be classified by the classification engine 108. The classification data 116 can include each variant in the variant space for a given sequence. For example, the candidate identification system 102 can start with a predetermined antibody and calculate all possible variants of the antibody. Each of the variants can be stored in the memory 106 as classification data 116.

[69] The candidate identification system 102 can store training data 118 in the memory 106. The training data 118 can include a data structure that includes indications of a plurality of variants. Each variant of the training data 118 can be stored separately (e.g., as a single string or vector) or collectively (e.g., as a matrix where each column or row corresponds to a different variant). The training data can be labeled training data 118 to indicate whether the respective variant is a binding or non-binding variant. For example, each variant can be stored as a binary file encoding the sequence of the variant. The binary file can include a leading (or trailing) bit that can be set (e.g. set to 1) to indicate that the variant is a binding variant or not set (e.g., set to 0) to indicate that the variant is a non-binding variant.

[70] The training data 118 can be a set of variants that is selected by physical screening of a rationally designed library of variants based on a selected parameter (e.g., antigen binding). For example, in some embodiments, the training data includes numerical values. In some embodiments, the numerical values correspond to binding kinetic values for a set of variants. In some embodiments, the numerical values correspond to numerical value results for biophysical assays (e.g., melting temperature for thermal stability, or AC-SINS for solubility). Exemplary methods for generation of the training data is described in further detail (see, e.g., FIG. 4A).

[71] The classification engine 108 can be trained using the training data 118. The classification engine 108 can be trained, in this example, to predict specificity towards a target antigen. As described further, below, in relation to FIGS. 2 and 3, the training data 118 (like the classification data 116) can be one-hot encoded for input into the classification engine 108. The training data 118 can be divided into training data and testing data. For example, the training data can be used to train the classification engine 108 and the testing data can be reserved to test the accuracy and precision of the trained classification engine 108 instead for the training of the classification engine 108. The testing data can be labeled to enable the classification engine 108 to determine whether the variants of the testing data was properly classified. In one example, 70% of the training data 118 can be set aside for training and 30% can be used for testing or evaluation of the classification engine 108. The testing data can be split to include predetermined proportions of binder to non-binder variants. For example, the testing data can be split to approximately 10/90 binders/non-binders to resemble physiological frequencies.

[72] The candidate selection engine 110 can store candidate variants in the memory 106 as candidate data 120. The candidate data 120 can be a data structure that can indicate each of the antigen-binder classified variants that satisfy the parameters of the filtering policy. The candidate data 120 can be a data structure that can indicate each variant classified as an antigen-binder before or without processing the antigen-binder classified variants with the filtering policy. The data structure can be a text-based file or a binary file that indicates the sequence of the variant. For example, the sequence can be stored as a character string in a

text-based file. The data structure (or file) can include metadata such as which positions were mutated with respect to wild-type and the nature of the mutation. The metadata can include a classification score that indicates the certainty with which the classification engine 108 classified the antigen-binder classified variant as an antigen-binder classified variant.

[73] FIG. 2 illustrates an example neural network 200. The neural network 200 can be an LSTM neural network 200. See FIG 2A. The LSTM neural network 200 can include a plurality of nodes 202, which can also be referred to as neurons 202. The nodes 202 can be arranged in layers. For example, the node 202 can include an input layer of nodes 202, one or more hidden layers of nodes 202, and an output layer of nodes 202. Each of the layers can include one or more nodes 202. For example, the input layer can include 10 nodes 202 (e.g., the number of nodes 202 in the input layer is equal to the length of the input vector 204) and the output layer can include one node 202. The node 202 of the output layer can indicate the probability that the input vector 204 corresponds to an antigen-binder classified variant. The LSTM neural network 200 can include two output nodes 202 - one node 202 that provides the probability that the variant is an antigen-binder classified variant and a second node 202 that provides the probability that variant is a non-antigen-binder classified variant.

[74] The LSTM neural network 200 can include between about 2 and about 10, between about 2 and about 8, between about 2 and about 6, between about 2 and about 4, or between about 2 and about 3 layers. Each layer can include the same number of nodes 202 or a different number of nodes 202. The input layer can include a node 202 for each value on a one-hot encoded matrix input. For example, for a 10x20 one-hot encoded matrix, the input layer can include 200 nodes 202. The number of nodes 202 in the input layer can be based on the number of values in the input sequence (e.g., the number of amino acids in the sequence) times the number of possible values for each value. For example, for a sequence of length 10 with 20 possible amino acids per position, the input layer can include $10 \times 20 = 200$ nodes 202. The LSTM neural network 200 can include a plurality of hidden layers. Each of the hidden layers can include the same or a different number of nodes 202. The hidden layers can include fewer nodes 202 than the input layer. For example, the hidden layers can each include 40 nodes 202.

[75] Each node 202 in a layer can be linked to each node 202 in a subsequent layer. Each node 202 outputs, to the nodes 202 to which it is connected, a weighted sum of the node's inputs. The node 202 can add a bias to the weighted sum to bias the output. The node 202 can include an activation function (e.g., a sigmoid function, a rectified linear unit (ReLU), or leaky rectified linear unit) that determines when the node 202 "fires" or outputs a signal based on the weighted sum. The weights of each link and the bias for each node 202 can be set during the training phase and stored as classifier weights 112. The LSTM neural network 200 can be a recurrent neural network, and each node 202 can provide feedback (or input) to itself. The recurrent neural network can create an internal state to exhibit temporal behaviors.

[76] To classify a variant, the classification engine 108 converts the sequence of the variant into an input vector 204, where each value of the input vector 204 corresponds to a respective amino acid of the sequence. The input vector 204 has a length equal to the length of the input sequence. The classification engine 108 can one-hot encode the input vector 204 to generate a matrix 206. The input vector 204 can include other features of the variant sequence. For example, biophysical properties of the variant sequence can be encoded into the input vector 204. Each row of the matrix 206 corresponds to a respective value (e.g., position) of the input vector 204. Each column of the matrix 206 corresponds to a different possible amino acid that can fill each respective value of the input vector 204. In this example, as there are twenty amino acids, the matrix 206 includes twenty columns. Each row of the matrix 206 includes a 1 in the column corresponding to the amino acid present in the respective value of the input vector 204. The matrix 206 can be flattened to a vector and each value from the vector can be provided to one of the nodes 202 of the input layer. The matrix 206 can be sequentially provided to the nodes 202 of the input layer. For example, the input layer can include 10 input nodes 202 and the columns (e.g., the 10 values of each column) of the matrix 206 can be sequentially provided to the input nodes 202.

[77] To classify a variant, the classification engine 108 can convert the sequence of the variant into an input vector 204, where each value of the input vector 204 corresponds to a respective amino acid of the sequence. The input vector 204 has a length equal to the length of the input sequence. Encoding of the input vector can also take place based on protein physical

properties, as each individual amino acid is represented with a collection of physical properties (e.g., charge, hydrophobicity, volume).

[78] FIG. 2B illustrates the receiver operating characteristic (ROC) curve 208 for the LSTM neural network 200 on a test data set and the precision-recall (PR) curve 210 for the LSTM neural network 200. The ROC curve 208 and the PR curve 210 indicate the accuracy of the LSTM neural network 200. The curves 208 and 210 were generated by providing the LSTM neural network 200 a test data set of unseen variants at a 50/50 split of binders to non-binders.

[79] FIG. 3 illustrates an example neural network 300. The neural network 300 can be a convolutional neural network 300. See FIG. 3A The convolutional neural network 300 can include a plurality of nodes 202. The convolutional neural network 300 can include a plurality of layers 302. Unlike the neural network 200, each of the layers 302 in the convolutional neural network 300 may not be fully connected. For example, a node 202 of a given layer 302 may not be connected to each node 202 in a subsequent layer 302. The convolutional neural network 300 can include a plurality of filters. The convolutional neural network 300 can convolve the matrix 206 with each of the plurality of filters to generate a plurality of feature maps. Each filter can be configured to detect predetermined patterns in the matrix 206. The filter can be 1D convolutional filters with a dilation rate of and a stride size of 1 with a kernel size of 3, which can result in a filter of size 20x3. The convolution neural network 300 can include between about 100 and about 400 filters. The numbers of filters can be selected by cross-validation, or splitting the data into train/validation/test sets and choosing the optimal configuration via a random/grid-search. The convolutional neural network 300 can include one or more max pooling layers to reduce the spatial size of the feature maps. The convolutional neural network 300 can include a flattening layer that flattens the max pooled layer into an input vector for a fully connected layer of nodes. Each value in the flattened layer can act as an input to each of the nodes 202 in the dense (or fully connected) layer. The convolutional neural network 300 can include 50 nodes 202 in the dense layer. The number of nodes can be selected based on a limited cross-validation/grid-

search procedure. As with the LSTM neural network 200, each node 202 in the dense layer can serve as an input to an output node 202.

[80] FIG. 3B illustrates the ROC curve 308 for the convolutional neural network 300 on a test data set and the PR curve 310 for the convolutional neural network 300. The ROC curve 308 and the PR curve 310 indicate the accuracy of the convolutional neural network 300. The curves 308 and 310 were generated by providing the convolutional neural network 300 unseen variants at a 50/50 split of binders to non-binders.

[81] Referring to FIGS. 2 and 3, among others, the LSTM neural network 200 and convolutional neural network 300 architecture and hyper-parameters were selected by performing a grid search across various parameters. For example, the LSTM neural network 200 the grid search was performed to determine the nodes 202 per layer, the batch size, the number epochs, and optimizing function. For the convolutional neural network 300, classification engine 108 determines the number of filters, the kernel size, the dropout rate, the number of nodes 202 in the dense layer nodes based on a k-fold cross-validation of the data set.

[82] FIG. 4A illustrates a flow process 400 for generating the training data 118. The training data 118 can be a set of variants that is selected by physical screening of a rationally designed library of variants based on a selected parameter (e.g., antigen binding). The flow process 400 can include generating a point mutation library using, for example, homology-directed mutagenesis (HDM) or any other suitable mutagenesis method. In some aspects, the set of variants is selected in a two-step screening process that includes single-site (i.e. point mutation) and combinatorial deep mutational scanning (DMS) processes, an example of which is illustrated in flow process 400. The amino acid sequence of an antibody's heavy chain complementarity determining region 3 (CDRH3) is a key determinant of antigen specificity. Thus, two-step DMS process can be performed on this selected region (e.g., 10 amino acids of the CDRH3) to resolve the specificity determining amino acid positions. In some aspects, a mutant full-length antibody that has a variant CDRH3 sequence (e.g., a mutated CDRH3 sequence) such that the antibody no longer binds to its antigen can be used

as a starting sequence. Starting with mutant non-binding variant can provide advantages in the selection of binders from the library by reducing the background from the original sequence. In some alternate implementations, the process can start with a variant that still binds to its antigen.

[83] While FIG. 4A exemplifies training data for the CDRH3 of an antibody, the methods described herein are not so limited and can be applied to a set of variants for one or more regions of interest in an antibody or other binding protein, such as a receptor that binds to a ligand. For example, the set of variants can represent other CDR regions of an antibody, such as CDRH1, CDRH2, CDRL1, CDRL2, CDRL3, combinations of two or more CDR regions, a region within the framework domains of the antibody (e.g., FR1, FR2, FR3, FR4), or regions within the constant domains of the antibody (e.g., CH1, CH2, CH3) for which improvement of one or more properties of the antibody is desired. In some aspects, the variant is a full-length antibody. In some aspects, the variant is a fragment of an antibody of a recombinant antibody comprising an antigen binding domain, such as an scFv or an Fc fusion protein. In some aspects, the training data is derived from variants of a binding protein, such as a receptor, that binds to a ligand.

[84] In the first step of the exemplary flow process 400, a mutagenesis method is applied to the CDRH3 sequence to generate a library of variants as single sites at each position of the CDRH3 sequence (referred to herein as single-site DMS). Any suitable method of producing single point mutations can be employed. In some aspects, a hybridoma cell line expressing a full-length antibody variant sequence is used. Libraries of variant antibody sequences can be generated by CRISPR-Cas9-mediated homology-directed mutagenesis (HDM) (See, e.g., PCT Publication No. WO 2017/174329, which is incorporated by reference in its entirety). For example, gRNA for Cas9 targeting of CDRH3 and a pool of homology templates in the form of single-stranded oligonucleotides (ssODNs) containing NNK degenerate codons at single amino acid positions across the CDRH3 can be used to introduce point mutations at single sites in the CDRH3 of the antibody. Alternatively, any suitable mutagenesis method can be used to generate variants, for example, error-prone PCR, recombination mutagenesis, alanine scanning mutagenesis, structure-guided mutagenesis. In some aspects, the

mutagenesis can be performed on the nucleic acid sequence encoding the amino acid sequence of interest using *in vitro* techniques (e.g. PCR) and then the variant nucleic acids introduced into mammalian cells (e.g., by CRISPR-Cas9 HDR).

[85] Libraries of cells expressing the variant full-length antibodies can then be screened by a suitable method to detect antigen binding, such as by fluorescence-activated cell sorting (FACS). Exemplary FACS results for the first step of the screening process are shown in the first step of process 400. Populations of cells expressing the antibody and selected for binding or not binding antigen can then be subjected to deep sequencing to determine the antibody sequences expressed by the selected cells.

[86] The flow process 400 can include deep mutational scanning to determine enrichment scores for each amino acid position assayed to determine which positions are more or less amenable to accepting mutations. For example, the variant libraries were screened by FACS, and populations expressing antibody and binding or not binding antigen were subjected to deep sequencing. In some aspects, populations of cells that bind to two or more antigens are selected (e.g. cross-reactive or multispecific antibodies). The enrichment scores, which can be referred to as enrichment ratios (ER), can be the ratio of clonal frequencies of variants enriched for antigen specificity by FACS, f_i,Ag_+, to the clonal frequencies of the variants present in the original library, f_i,Ab_+. More particularly:

$$ER = \frac{f_i, Ag_+}{f_i, Ab_+} \quad (1)$$

[87] In some implementations, a minimum value of -2 was designated to variants with log[ER] values less than or equal -2 and variants not present in the dataset were disregarded in the calculation. A clone was defined based on the specific amino acid sequence of the CDRH3. Heatmaps and their corresponding sequence logo plots can then be generated based on the enrichment scores from the first step of the screening process. The heatmaps and sequence logo plots can then be used to rationally design a combinatorial mutagenesis library for screening. Degenerate codons can be selected per position based on their amino acid frequencies which most closely resemble the degree of enrichment or enrichment score found

in the analysis of the DMS data. For example, the codon selection for the rational library design can be based on the below equation. The amino acid positions identified in DMS analysis that have a positive enrichment score (e.g., ER > 1, or log[ER] > 0) were normalized according to their enrichment ratios and were converted to theoretical frequencies. Degenerate codon schemes were then selected which most closely reflect these frequencies as calculated by the mean squared error between the degenerate codon and the target frequencies.

$$Optimal\ Codon = arg_x min\left(\frac{1}{n}\sum_{i=1}^{n} \quad w_n(Y_{n,deg} - Y_{n,target})^2\right) \tag{2}$$

[88] For example, the heatmap and sequence logo plots indicate that position 103 (FIG. 5) is highly acceptable of glycine (G) and serine (S) residues, and to a lesser extent alanine (A). The enrichment scores for these residues correspond to normalized frequencies of approximately 66% G, 25% S, and 9% A. These frequencies are then input values to the above optimal codon equation (e.g., Equation 2) and compared against all 3,375 possible degenerate codon schemes. In this example, the degenerate codon scheme 'RGY' was selected as it represents the degenerate codon scheme with the closest frequencies (50% G, 50% S) to the target frequencies defined by the normalized enrichment scores. Combining degenerate codons across multiple positions produces massive theoretical protein spaces. As an example, by taking the product of all potential amino acids per position across all positions, the combinatorial library generated for the trastuzumab antibody described in the Examples provided herein possessed a theoretical protein sequence space of $6.67x10^8$, which is far higher than the single-site DMS library diversity of 200. The combinatorial mutagenesis libraries containing CDRH3 variants can then be physically generated, e.g., in hybridoma cells through HDM. Antigen binding cells can then be isolated by one or more rounds of enrichment by FACS and the binding or non-binding populations subjected to deep sequencing. Sequencing data representing the binding or non-binding populations from this second step can then be employed as the training set for the machine learning model.

[89] FIG. 4B illustrates a process flow 450 for selecting candidate variants. The process flow 450 can include training the models described herein with the trained data generated during the process flow 400. Once the training data is generated and the classification engine 108 is trained, the full sequence space of mutations can be generated *in silico*. The full sequence space can include each possible mutation. The number of variants in the full sequence space can be orders of magnitude larger than the number of variants on which the classification engine 108 was trained. The classification engine 108 can process the variants of the full sequence space to classify the variants as antigen-binder classified variants or non-antigen-binder classified variants. The process flow 450 can include the candidate selection engine 110 filtering the antigen-binder classified variants with multi-parameter optimization to select one or more candidate variants. The candidate selection engine 110 can filter antigen-binder classified variants by determining whether the antigen-binder classified variants satisfy a filtering policy. The filtering policy can include parameter requirements such as model consensus (e.g., did each of the LSTM neural network and convolutional neural network classify the variant as an antigen-binder classified variant), viscosity values, solubility values, stability values, pharmacokinetic values, and immunogenicity values.

[90] FIGS. 5 and 6 illustrate exemplary data for the process flow 400 and 450 as applied to the CDRH3 of exemplary antibody Trastuzumab, which are described in further detail in the Example below.

[91] FIG. 7 illustrates a filtering policy 700 and a plurality of plots of parameters. As described above, for each of the antigen-binder classified variants, the candidate selection engine 110 can calculate parameter values. The system 100 can calculate, for example, a Levenshtein distance value, charge value, hydrophobicity index value, CamSol score, minimum affinity rank, and average affinity ranking for each antigen-binder classified variant. The system 100 can also identify within each of the antigen-binder classified variants sequence motifs associated with manufacturing liabilities, such as n-glycosylation sites, deamidation sites, isomerization sites, methionine oxidation, tryptophan oxidation and paired or unpaired cysteine residues.

[92] The filtering policy 700 can include a plurality of parameter requirements. The candidate selection engine 110 can apply the parameter requirements in parallel. For example, the candidate selection engine 110 can calculate each of the parameter values for each of the antigen-binder classified variants and determine whether the antigen-binder classified variants satisfy the parameter requirements of the filtering policy 700. The candidate selection engine 110 can apply the parameter requirements in series. For example, the candidate selection engine 110 can sequentially calculate a parameter for the antigen-binder classified variants and determine whether the antigen-binder classified variant satisfies the parameters required for the given parameter. The system 100 may then only calculate the next parameter values for the antigen-binder classified variants that satisfied the first parameter requirement. When an antigen-binder classified variant does not satisfy a parameter requirement, the candidate selection engine 110 may not calculate the remaining parameter values for the antigen-binder classified variant. This can reduce the computational resources required to filter the antigen-binder classified variants as the parameter values are not calculated for the antigen-binder classified variants once they are removed by the filtering process. Thus, by determining to not calculate parameter values for antigen-binder classified variants that do not satisfy the parameter requirement, this technical solution can reduce computational resource consumption (e.g., processor utilization, memory utilization, or network bandwidth utilization), while identifying optimal variants.

[93] Still referring to FIG. 7, the candidate selection engine 110 can first determine the antigen-binder classified variants output by the recurrent neural network (RNN) and the convolutional neural network (CNN). The candidate selection engine 110 can select only the variants that were classified by the respective neural network with a predetermined confidence. For example, as illustrated in FIG. 7, the candidate selection engine 110 can identify 4,315,323 antigen-binder classified variants identified by the recurrent neural network and 5,218,706 antigen-binder classified variants identified by the convolution neural network with a confidence or probability above 0.75. The next filter in the filtering policy 700 can include identifying antigen-binder classified variants identified by both the convolutional neural network and the recurrent neural network. The candidate selection engine 110 can identify 3,159 ,373 antigen-binder classified variants identified by both

the convolutional neural network and the recurrent neural network with a probability greater than 0.75 . The candidate selection engine 110 can then identify the antigen-binder classified variants with a charge symmetry parameter greater than 6.61, a net charge less than 6 .2 and a hydrophobicity index less than 4, returning 402,633 antigen-binder classified variants. The candidate selection engine 110 can then identify antigen-binder classified variants with a solubility score greater than 0.5 , returning 14,125 antigen-binder classified variants. The candidate selection engine 110 can then identify the antigen-binder classified variants with a NetMHCII minimum affinity rank greater than 5.5% and an average affinity rank greater than 60.6 %, returning 4,881 antigen-binder classified variants. All remaining antigen-binder classified variants in this example contain values equal or greater than the parameters of the starting candidate sequence of trastuzumab. The candidate selection engine 110 can then identify the antigen-binder classified variants with the best overall developability across all parameters, returning the antigen-binder classified variants within the top percentage of the remaining candidate variants according to a predefined percentage. The system 100 can additionally identify the antigen-binder classified variants with a Levenshtein distance less than 5 .

[94] FIG. 8 illustrates a block diagram of an example method 800 to identify antibodies with an antigen affinity. The method 800 can include generating the training data (ACT 802). The method 800 can include training the classification model (ACT 804). The method 800 can include classifying variants (ACT 806). The method 800 can include filtering the variants (ACT 808). The method 800 can include selecting variants (ACT 810).

[95] As set forth above, the method 800 can include generating training data (ACT 802). Also, with reference to FIG. 1, among others, the classification engine 108 can use the training data 118 for training to determine the classifier weights 112 for classifying unseen variants. The training data 118 can be generated using a two-step process that includes a single-site mutation process followed by a DMS-based combinatorial process.

[96] The method 800 can include training the classification model (ACT 804). As described above, the classification engine 108 can include one or more classification models. For

example, the classification engine 108 can include a recurrent neural network or a convolution neural network. The classification engine 108 can include a recurrent neural network, a convolution neural network, a standard artificial neural network (ANN), a support vector machine (SVM), a random forest ensemble (RF) or logistic regression (LR) model. The training data 118 can be labeled and passed to the neural networks as a one-hot encoded matrix. The classification engine 108 can use back-propagation and gradient descent to minimize the cost or error between the expected result and result determined by the classification engine 108. Once the classification engine 108 has trained its neural network, the classification engine 108 can save the weights and biases to the memory 106 as classifier weights 112.

[97] The method 800 can include classifying variants (ACT 806). In some implementations, for a given antibody, the candidate identification system 102 can *in silico* generate the complete sequence space for the variants of the antibody. For example, the candidate identification system 102 can generate all possible sequence variations for a given antibody or portion thereof. The classification engine 108 can load the classifier weights 112. The classification engine 108 can pass each of the variants of the complete sequence space to the input layers of the convolutional neural network and recurrent neural network. For example, each variant, the classification engine 108 can determine a probability that the variant is an antigen-binder classified variant. The classification engine 108 can save the antigen-binder classified variants with a probability above a threshold as antigen-binder classified variants in the memory 106.

[98] The method 800 can include filtering the antigen-binder classified variants (ACT 808). The candidate selection engine 110 can filter the antigen-binder classified variants to identify candidate variants. The candidate variants can be the antigen-binder classified variants that have the greatest probability of yielding viable antibodies. The candidate selection engine 110 can retrieve a filtering policy from the memory 106. The filtering policy can include a plurality of parameters that the antigen-binder classified variants must satisfy to be selected as a candidate variant. The candidate selection engine 110 can calculate the parameters for

the antigen-binder classified variants and determine if each of the respective antigen-binder classified variants satisfy the parameter requirements of the filtering policy.

[99] The method 800 can include selecting variants (ACT 810). The candidate variants (e.g., the antigen-binder classified variants that satisfy the parameters of the filtering policy) can be selected for further recombinant expression to test the variant produces an antibody with antigen-specific binding. In some implementations, a sub-portion of the candidate variants can be randomly selected for recombinant expression and testing.

[100] While operations are depicted in the drawings in a particular order, such operations are not required to be performed in the particular order shown or in sequential order, and all illustrated operations are not required to be performed. Actions described herein can be performed in a different order.

[101] The separation of various system components does not require separation in all implementations, and the described program components can be included in a single hardware or software product.

I. EXAMPLE

[102] This Example describes an exemplary application of the systems and methods described herein to the CDRH3 of Trastuzumab (Herceptin) antibody and classify antibody binding to the corresponding target HER2 antigen.

A. Results

1) Deep mutational scanning determines antigen-specific sequence landscapes and guides rational antibody library design

[103] As the amino acid sequence of an antibody's CDRH3 is a key determinant of antigen specificity, deep mutational scanning (DMS) was performed on this region to resolve the specificity determining residues. To start, a hybridoma cell-line expressing a trastuzumab

variant that could not bind HER2 antigen (mutated CDRH3 sequence) was used (FIG. 9). Libraries were generated by CRISPR-Cas9-mediated homology-directed mutagenesis (HDM) (Mason et al. (2018) *Nucleic Acids Research* 46 (14): 7436–49), which utilized gRNA for Cas9 targeting of CDRH3 and a pool of homology templates in the form of single-stranded oligonucleotides (ssODNs) containing NNK degenerate codons at single-sites tiled across CDRH3 (FIG.5A). Libraries were then screened by fluorescence activated cell sorting (FACS), and populations expressing antibody and binding or not binding antigen were subjected to deep sequencing (Illumina MiSeq) (FIG. 10). Deep sequencing data was then used to calculate enrichment scores, of the 10 positions investigated, which revealed six positions that were sufficiently amenable to a wide-range of mutations with an additional three positions that were marginally accepting to defined mutations (FIG. 5B and 5C). Although residues 103102D, 103G, 104F, and 105Y appear to be the primary contacting amino acids of the CDRH3 loop with HER2 (PDB ID:1N8Z, Cho et al. (2003) Nature 421 (6924): 756–60, Rose et al. (2018) *Bioinformatics* 34 (21): 3755–58., 105Y is the only residue completely fixed (FIG. 5D).

[104] Heatmaps and their corresponding sequence logo plots generated by DMS were used to guide the rational design of a combinatorial mutagenesis library, which consisted of degenerate codons across all positions (except 105Y) (FIG. 11). Degenerate codons were selected per position based on their amino acid frequencies which most closely resembled the degree of enrichment found in the DMS data (FIG. 5C, Equation 2). This combinatorial library possesses a theoretical protein sequence space of 6.67×10^8 which is far greater than the single-site DMS library diversity of 200. The theoretical diversity can be calculated by taking the product of all possible amino acids per position across all positions (e.g., all 20 amino acids present at all positions results in 20^X, where X is the number of positions). In some implementations, DMS-guided combinatorial mutagenesis libraries can have a reduced subset of amino acids per position, resulting in a reduction of theoretical diversity. Libraries containing CDRH3 variants were again generated in hybridoma cells through HDM in the same non-binding trastuzumab clone described previously (FIG. 6A). Antigen binding cells were isolated by two rounds of enrichment by FACS and the binding/non-binding populations were subjected to deep sequencing. Sequencing data identified 11,300 and

27,539 unique binders and non-binders, respectively (NGS statistics, FIG. 13). These sequence variants represented only a miniscule 0.0058 % of the theoretical protein sequence space of the combinatorial mutagenesis library. Amino acid usage per position was comparatively similar between binding and non-binding populations (FIG. 6B), thus making it difficult to develop any sort of heuristic rules or observable patterns to identify binding sequences.

2) Training deep neural networks to classify antigen-specificity based on antibody sequence

[105] After having compiled deep sequencing data on binding and non-binding CDRH3 variants, deep learning models capable of predicting specificity towards the target antigen HER2 were developed and trained. Amino acid sequences were converted to an input matrix by one-hot encoding, an approach where each column represents a specific residue and each row corresponds to the position in the sequence, thus a 10 amino acid CDRH3 sequence as here results in a 10 x 20 matrix. Each row will contain a single '1' in the column corresponding to the residue at that position, whereby all other columns/rows receive a '0'. LSTM-RNNs and CNN. LSTM-RNNs and CNNs both stem from standard neural networks, where information is passed along neurons that contain learnable weights and biases, however, there are fundamental differences in how the information is processed. LSTM-RNN layers contain loops, enabling information to be retained from one step to the next, allowing models to efficiently correlate a sequential order with a given output; CNNs, on the other hand, apply learnable filters to the input data, allowing it to efficiently recognize spatial dependencies associated with a given output. Model architecture and hyperparameters were selected by performing a grid search across various parameters (LSTM-RNN: nodes per layer, batch size, number epochs, and optimizing function; CNN: number of filters, kernel size, dropout rate, dense layer nodes) using a k-fold cross-validation of the data set (Figure 7). All models were built to assess their accuracy and precision of classifying binders and non-binders from the available sequencing data. 70% of the original data set was used to train the models and the remaining 30% was split into two test data sets used for model evaluation: one test data set contained the same class split of sequences used to train the model and the

other contained a class split of approximately 10/90 binders/non-binders to resemble physiological frequencies (FIGS. 6A and 14). Performance of the LSTM-RNN and CNN were assessed by constructing receiver operating characteristic (ROC) curves and precision-recall (PR) curves derived from predictions on the unseen testing data sets. Based on conventional approaches to training classification models, the data set was adjusted to allow for a 50/50 split of binders and non-binders during training. Under these training conditions, LSTM-RNN and CNN were able to accurately classify unseen test data (ROC curve AUC: 0.9 ± 0.0, average precision: 0.9 ± 0.0, FIG. 17).

[106] Next, the trained LSTM-RNN and CNN models were used to classify a random sample of $1x10^5$ sequences from the potential combinatorial diversity space were used. However, an unexpectedly high occurrence of positive classifications (25,318 ± 1,643 sequences or 25.3 ± 1.6%, FIG. 21) was observed. In view of the knowledge that the physiological frequency of binders should be approximately 10–15%, the classification split of the training data with the hypothesis that models were being subject to some unknown classification bias was adjusted. Additional models were then trained on classification splits of both 20/80, and 10/90 binders/non-binders, as well as a classification split with all available data (approximately 30/70 binders/non-binders). Unbalancing the sequence classification led to a significant reduction in the percentage of sequences classified as binders, but also led to a reduction in the model performance on the unseen test data (FIG. 21). Through this analysis, it was concluded that the optimal data set for training the models was the set inclusive of all known CDRH3 sequences for the following reasons: 1) the percentage of sequences predicted as binders reflects this physiological frequency, 2) this data set maximizes the information the model sees, and 3) model performance on the test data. Final model architecture, parameters, and evaluation are shown in FIG. 2.

3) Multi-parameter optimization for developability by in silico screening of antibody sequence space

[107] Next, the full 3.1×10^6 deep learning predicted antigen-specific sequences were characterized on a number of parameters to identify highly developable candidates compared

to the original trastuzumab sequence. As a preliminary metric, their sequence similarity to the original trastuzumab sequence was investigated by calculating the LD. The majority of sequences showed an edit distance of LD > 4 (FIG. 7A). The first step in filtering was to calculate the net charge and hydrophobicity index in order to estimate the molecule's viscosity and clearance. According to Sharma et al., viscosity decreases with increasing variable fragment (Fv) net charge and increasing Fv charge symmetry parameter (FvCSP); however, the optimal Fv net charge in terms of drug clearance is between 0 and 6.2 with a CDRL1+CDRL3+CDRH3 hydrophobicity index sum (HI sum) < 4. Based on the wide range of values for these parameters in the 3.1×10^6 predicted variants (FIG.7B,C), we filtered any sequences out that had a FvCSP < 6.61 (trastuzumab FvCSP) or if they contained a Fv net charge > 6.2, and an HI sum > 4, < 0. This filtering criteria greatly reduced the sequence space down to 4.02×10^5 variants. We next padded the CDRH3 sequences with 10 amino acids on the 5' and 3' ends and then ran these sequences through CamSol, a protein solubility predictor developed by Sormanni et al., which estimates and ranks sequence variants based on their theoretical solubility. The remaining variants produced a wide-range of protein solubility scores (FIG. 7D) and sequences with a score < 0.5 (trastuzumab score) were filtered out, leaving 14,125 candidates for further analysis. As a last step in the in silico screening process, we aimed at reducing immunogenicity by predicting the peptide binding affinity of the variant sequences to MHC Class II molecules by utilizing NetMHCIIpan, a model previously developed by Jensen et al. One output from the model is a given peptide's % Rank of predicted affinity compared to a set of 200,000 random natural peptides. Typically, molecules with a % Rank < 2 are considered strong binders and those with a % Rank < 10 are considered weak binders to the MHC Class II molecules scanned. All possible 15-mers from the padded CDRH3 sequences were run through NetMHCIIpan. After predicting the affinities for a set of 26 HLA alleles determined to cover over 98% of the global population32, sequences were filtered out if any of the 15-mers contained a % Rank < 5.5 (trastuzumab minimum % Rank) (FIG 7E). The number of 15-mers with a % Rank less than 10 and the average % Rank across all 15-mers for the remaining sequences were also calculated. Sequences with more than two 15-mers with a % Rank < 10 (FIG 7F) and those with an average % Rank < 60.56 (trastuzumab average % Rank) were also filtered out (FIG 7G). All remaining 4,881 variants contain values equal to or greater than the parameters of the original trastuzumab sequence. When applying this same

filtering scheme on the 11,300 experimentally determined binding sequences (obtained from training / test data), only 9 variants remained. Lastly, to determine the best developable sequences, we calculated an overall developability improvement score based on the mean of normalized values for each relevant parameter (see Materials and Methods), where trastuzumab would have a developability improvement score equal to 0. Of the remaining 4,881 predicted binding sequences, 293 variants were identified to have a higher developability score compared to the maximum developability score of the 9 experimentally determined binding sequences (Figure 7H). The filtering parameters and number of remaining variants at each step for the in silico library are provided in Figure 7I.

4) Selected antibody sequences are recombinantly expressed and antigen-specific

[108] To validate the precision of the fully trained LSTM-RNN and CNN models, a subset of 30 CDRH3 sequences predicted to be antigen-specific and optimized across the multiple developability parameters was randomly selected. To further demonstrate the capacity of deep learning to identify novel sequence variants, the criteria that the selected variants have a minimum Levenshtein edit distance of 5 from the original CDRH3 sequence of trastuzumab was also added. CRISPR-Cas9-mediated HDR was used to generate mammalian display cell lines expressing different sequence variants. Flow cytometry was performed and revealed 29 of the 30 variants (96.67%) were antigen-specific (FIG. 23 A–23 O) Further analysis was performed on 104 of the antigen-binding variants to more precisely quantify the binding kinetics via biolayer interferometry (FortéBio Octet) (FIG. 15, FIG. 16B, FIGS. 23A-G). The original trastuzumab sequence was measured to have an affinity towards HER2 of 4.0 x 10^{-10} M (equilibrium dissociation constant, Kd); and although the majority of variants tested had a slight decrease in affinity, 75 % (78 /104) were still in the single-digit nanomolar range, 16 % (17 /104) remained sub-nanomolar, and six variants (5 %) showed an increase in affinity compared to trastuzumab (Kd = 1.4 x 10^{-10} M).

[109] Developability parameters for the selected variants were also experimentally validated. In particular, expression levels of the selected variants were compared to those of

trastuzumab (FIG. 16A). Further, thermal stability of the selected variants were compared to those of trastuzumab. (FIG. 16C). Immunogenicity risk was also compared to trastuzumab, where each tested variant (variants C and F) and trastuzumab were each tested twice (FIG. 16D).

B. Discussion

[110] Addressing the limitation of antibody optimization in mammalian cells, an approach based on deep learning has been developed that enables identification of antigen specific sequences with high precision. Using the clinically approved antibody trastuzumab, single-site DMS was performed followed by combinatorial mutagenesis to determine the antigen-binding landscape of CDRH3. This DMS-based mutagenesis strategy was important for attaining high quality training data that is enriched with antigen-binding variants, in this case nearly 10% of the generated library (FIG. 14). In contrast, if a completely randomized combinatorial mutagenesis strategy was employed (i.e., NNK degenerate codons), it would be unlikely to produce any significant fraction of antigen-binding variants.

[111] A remarkable finding in this study was that experimental screening of a library of only 5 x 10^4 variants, which reflected a tiny fraction (0.0005%) of the total sequence diversity of the DMS-based combinatorial mutagenesis library (6.67 x 10^8), was capable of training accurate neural networks. This suggests that physical library size limitations of mammalian expression systems (or other expression systems such as phage display and yeast display) and deep sequencing read depth will not serve as a limitation in deep learning-guided protein engineering. Another important result was that deep sequencing of antigen-binding and non-binding populations showed nearly no observable difference in their positional amino acid usage (FIG. 6), suggesting that neural networks are effectively capturing high-dimensional patterns.

[112] In the current study, LSTM-RNNs and CNNs were selected as the basis of our classification models, as they represent two state of the art approaches in deep learning. Other machine learning approaches such as k-nearest neighbors, random forests, and support

vector machines are also well-suited at identifying complex patterns from limited input data. Furthermore, deep generative modeling methods such as variational autoencoders can also be used to explore the mutagenesis sequence space from directed evolution.

[113] Approximately 10^8 CDRH3 variants were in silico generated from the DMS-based combinatorial diversity and used the fully trained LSTM-RNN and CNN models to classify each sequence as a binder or non-binder. The ~10^8 sequence variants comprise only a subset of the potential sequence space and was chosen to minimize the computational effort, however it still represents a library size several orders of magnitude greater than what is experimentally achievable in mammalian cells. The screening capacity can be extended through script optimization and employing parallel computing on high performance clusters. Out of all variants classified, the LSTM-RNN and CNN predicted approximately 12-13% to bind the target antigen, showing exceptional agreement with the experimentally observed frequencies by flow cytometry (FIG. 14). With the exception of critical residues determined by DMS, the majority of predicted binders were substantially distant from the original Trastuzumab sequence with 80% of sequences having an edit distances of at least 6 residues. This high degree of sequence variability indicated the potential for a wide range of biomolecular properties.

[114] Once an antibody's affinity for its target antigen is within a desirable range for efficacious biological modification, addressing other biomolecular properties becomes the focus of antibody development. With recent advances in computational predictions, a number of these properties, including viscosity, clearance, stability, specificity, solubility, and immunogenicity can be approximated from sequence information alone. With the aim of selecting antibodies with improved characteristics, the library of predicted binders was subjected to a number of these in silico approaches in order to provide a ranking structure and filtering strategy for developability (FIG. 7). After implementing these methods to remove variants with a high likelihood of having poor viscosity, clearance, or solubility, as well as those with high immunogenic potential, approximately 5,000 multi-parameter optimized antibody variants remained. More stringent or additional filters can also be applied

to address other developability parameters (e.g. stability, specificity, humanization) to further reduce the sequence space down to highly developable therapeutic molecules.

[115] Lastly, to experimentally validate the accuracy of neural networks to predict antigen specificity, we randomly selected and expressed 30 variants from the library of optimized sequences with a minimum edit distance of 5 from trastuzumab. The precision of the LSTM-RNN and CNN models were each estimated to be ~85% (at $P > 0.75$) according to predictions made on the test data sets. By taking the consensus between models, however, it wasexperimentally validated that >96% (29/30) of the antigen-predicted (and developability filtered) sequences were indeed binders. This suggests that potentially thousands of optimized lead candidates, all substantially different from the starting trastuzumab sequence, maintain a binding affinity in the range of therapeutic relevance.

[116] The methods provided herein can be further modified to increase the stringency of selection during screening or investigation of correlations between prediction probability and affinity, which can assist in retaining high target affinities. These methods also can enable the optimization of other functional properties of therapeutic antibodies, such as pH-dependent antibody recycling or pH-dependent antigen binding. Additionally, extending this approach to other regions across the variable light and heavy chain genes, namely other CDRs, can yield deep neural networks that are able to capture long-range, complex relationships between an antibody and its target antigen. In addition, the described neural network predictions can be compared to protein structural modeling predictions.

C. Methods

1) Mammalian cell culture and transfection

[117] Hybridoma cells were cultured and maintained according to the protocols described by Mason et al. (2018) *Nucleic Acids Research* 46 (14): 7436–49. Hybridoma cells were electroporated with the 4D-Nucleofector™System (Lonza) using the SF Cell Line 4D-Nucleofector® X Kit L or X Kit S (Lonza, V4XC-2024, V4XC-2032) with the program CQ-104. Cells were prepared as follows: cells were isolated and centrifuged at 125 x G for 10

minutes, washed with Opti-MEM® I Reduced Serum Medium (Thermo, 31985-062), and centrifuged again with the same parameters. The cells were resuspended in SF buffer (per kit manufacturer guidelines), after which Alt-R gRNA (IDT) and ssODN donor (IDT) were added. All experiments performed utilize constitutive expression of Cas9 from *Streptococcus pyogenes* (SpCas9). Transfections of $1x10^6$ and $1x10^7$ cells were performed in 100 µl, single Nucleocuvettes™ with 0.575 or 2.88 nmol Alt-R gRNA and 0.5 or 2.5 nmol ssODN donor respectively. Transfections of $2x10^5$ cells were performed in 16-well, 20 ul Nucleocuvette™ stips with 115 pmol Alt-R gRNA and 100 pmol ssODN donor.

2) Flow cytometry analysis and sorting

[118] Flow cytometry-based analysis and cell isolation were performed using the BD LSR Fortessa™ (BD Biosciences) and Sony SH800S (Sony), respectively. When labeling with fluorescently conjugated antigen or anti-IgG antibodies, cells were first washed with PBS, incubated with the labeling antibody and/or antigen for 30 minutes on ice, protected from light, washed again with PBS and then analyzed or sorted. The labeling reagents and working concentrations are described in FIG. 23A and 23B. For cell numbers different from 10^6, the antibody/antigen amount and incubation volume were adjusted proportionally.

3) Sample preparation for deep sequencing

[119] Sample preparation for deep sequencing was performed similar to the antibody library generation protocol of the primer extension method described previously (Menzel, et al. (2014) *PloS One* 9 (5): e96727). Genomic DNA was extracted from $1-5x10^6$ cells using the Purelink™ Genomic DNA Mini Kit (Thermo, K182001). All extracted genomic DNA was subjected to a first PCR step. Amplification was performed using a forward primer binding to the beginning of the VH framework region and a reverse primer specific to the intronic region immediately 3' of the J segment. PCRs were performed with Q5® High-Fidelity DNA polymerase (NEB, M0491L) in parallel reaction volumes of 50 ml with the following cycle conditions: 98°C for 30 seconds; 16 cycles of 98°C for 10 sec, 70°C for 20 sec, 72°C for 30 sec; final extension 72°C for 1 min; 4°C storage. PCR products were concentrated using

DNA Clean and Concentrator (Zymo, D4013) followed by 0.8X SPRIselect (Beckman Coulter, B22318) left-sided size selection. Total PCR1 product was amplified in a PCR2 step, which added extension-specific full-length Illumina adapter sequences to the amplicon library. Individual samples were Illumina-indexed by choosing from 20 different index reverse primers. Cycle conditions were as follows: 98°C for 30 sec; 2 cycles of 98°C for 10 sec, 40°C for 20 sec, 72°C for 1 min; 6 cycles of 98°C for 10 sec, 65°C for 20 sec, 72°C for 1 min; 72°C for 5 min; 4°C storage. PCR2 products were concentrated again with DNA Clean and Concentrator and run on a 1% agarose gel. Bands of appropriate size (~550bp) were gel-purified using the Zymoclean™ Gel DNA Recovery kit (Zymo, D4008). Concentration of purified libraries were determined by a Nanodrop 2000c spectrophotometer and pooled at concentrations aimed at optimal read return. The quality of the final sequencing pool was verified on a fragment analyzer (Advanced Analytical Technologies) using DNF-473 Standard Sensitivity NGS fragment analysis kit. All samples passing quality control were sequenced. Antibody library pools were sequenced on the Illumina MiSeq platform using the reagent kit v3 (2x300 cycles, paired-end) with 10% PhiX control library. Base call quality of all samples was in the range of a mean Phred score of 34.

4) Bioinformatics analysis and graphics

[120] The MiXCR v2.0.3 program was used to perform data pre-processing of raw FASTQ files (Bolotin et al. (2015) *Nature Methods* 12 (5): 380–81). Sequences were aligned to a custom germline gene reference database containing the known sequence information of the V- and J-gene regions for the variable heavy chain of the trastuzumab antibody gene. Clonotype formation by CDRH3 and error correction were performed as described by Bolotin et al. Functional clonotypes were discarded if: 1) a duplicate CDRH3 amino acid sequence arising from MiXCR uncorrected PCR errors, or 2) a clone count equal to one. Downstream analysis was performed using R v3.2.2 (Cite R Development Core Team (2008)) and Python v3.6.5 (Van Rossumet al. (2011) The Python Language Reference Manual. Network Theory). Graphics were generated using the R packages ggplot2 (Wilkinson (2011) *Biometrics*, found at https://doi.org/10.1111/j.1541-0420.2011.01616.x., RColorBrewer (Brewer et al. (2003) *Cartography and Geographic Information Science*,

found at https://doi.org/10.1559/152304003100010929., and ggseqlogo (Wagih (2017) *Bioinformatics* 33 (22): 3645–47).

5) Calculation of enrichment ratios (ERs) in DMS

The ERs of a given variant was calculated according to previous methods (Fowler et al. (2010) *Nature Methods* 7 (9): 741–46). Clonal frequencies of variants enriched for antigen specificity by FACS, $f_{i,Ag+}$, were divided by the clonal frequencies of the variants present in the original library, $f_{i,Ab+}$, according to Equation 1, above.

[121] A minimum value of -2 was designated to variants with log[ER] values less than or equal -2 and variants not present in the dataset were disregarded in the calculation. A clone was defined based on the exact amino acid sequence of the CDRH3.

6) Redesign of trastuzumab in Rosetta for diversity of sequences

[122] The Rosetta program (Leaver-Fay et al.) was used to redesign the trastuzumab antibody in complex with the extracellular domain of HER2 (PDB id: 1N8Z) (Cho et al.). Ten residues in the CDRH3 loop of trastuzumab (residues 98-108 of the heavy chain) were allowed to mutate to any natural amino acid, while all other residues were allowed to change rotameric conformation. A RosettaScript invoked the PackRotamersMover, a stochastic MonteCarlo algorithm, to optimize the sequence of the antibody to CDRH3 according to the Rosetta energy function, followed by backbone minimization. Energies were computed using Rosetta's ddG filter. Rosetta was run to generate 5000 sequences stochastically, and this resulted in 48 sequences. Rosetta's output files were processed using RS-Toolbox (Bonet et al., 2019).

7) Classification of experimentally-determined sequences in Rosetta

[123] Each of the 11,300 binding and 27,539 non-binding sequences from the combinatorial library were modelled in Rosetta. For each experimentally-determined binding or non-binding sequence, the structure of the HER2:trastuzumab complex was used as input and the residues diverging from the wildtype were mutated using the PackRotamersMover in RosettaScripts

(Fleishman et al.). The backbone and the side chains were minimized with Rosetta's MinMover after the sequence was modeled to optimized intra- and inter-chain contacts. Rosetta's predicted interface score (ddG) was used as the relative classification score.

8) Codon selection for rational library design

[124] Codon selection for rational library design was based off the equation provided by Mason et al. (2018) *Nucleic Acids Research* 46 (14): 7436–49, (Equation 2). Residues identified in DMS analysis to have a positive enrichment (ER > 1, or log[ER] > 0) were normalized according to their enrichment ratios and were converted to theoretical frequencies. Degenerate codon schemes were then selected which most closely reflect these frequencies as calculated by the mean squared error between the degenerate codon and the target frequencies.

[125] In certain instances, if the selected degenerate codon did not represent desirable amino acid frequencies or contained undesirable amino acids, a mixture of degenerate codons was selected and pooled together to achieve better coverage of the functional sequence space.

9) Machine learning model construction

[126] Machine learning models were built in Python v3.6.5. K-nearest neighbor models and support vector machine models were built using the Scikit-learn libraries. Artificial neural networks, LSTM-RNNs, and CNNs were built using the Keras Sequential model as a wrapper for TensorFlow. Model architecture and hyperparameters were optimized by performing a grid search of relevant variables for a given model. These variables include nodes per layer, activation function(s), optimizer, loss function, dropout rate, batch size, number of epochs, number of filters, kernel size, stride length, and pool size. Grid searches were performed by implementing a k-fold cross validation of the data set.

10) Machine learning model training and testing

[127] Data sets for antibody expressing, non-binding, and binding sequences (Sequencing statistics: FIGS. 12 and 13) were aggregated to form a single, binding/non-binding data set where antibody expressing sequences were classified as non-binders, unless also identified among the binding sequences. Sequences from one round of antigen enrichment were excluded from the training data set. The complete, aggregated data set was then randomly arranged and appropriate class labeled sequences were removed to achieve the desired classification ratio of binders to non-binders (50/50, 20/80, 10/90, and non-adjusted). The class adjusted data set was further split into a training set (70%), and two testing sets (15% each), where one test set reflected the classification ratio observed for training and the other reflected a classification ratio of approximately 10/90 to resemble the physiological expected frequency of binders.

11) Sequence similarity and model attribution analysis of predicted variants

[128] Sequence similarity networks of sequences predicted to be antigen positive and antigen negative were constructed for Levenshtein Distance 1-6 were constructed using the igraph R package v1.2.4 (Csardi and Nepusz 2006). The resulting networks were analyzed with respect to their overall connectivity, the composition of their largest clusters and the overall degree distribution between the classes.

[129] The Integrated Gradients technique (Sundararajan et al. 2017) was used to assess the relative attribution of each feature of a given input sequence towards the final prediction score. First, a baseline was obtained by zeroing out the input vector and the path integral of the gradients from baseline to the input vector was then approximated with a step size of 100. Integrated gradients were visualized as sequence logos. Sequence logos were created by the python module Logomaker (Tareen and Kinney 2019).

12) In silico sequence classification and sequence parameters

[130] All possible combinations of amino acids present in the DMS-based combinatorial mutagenesis libraries were used to calculate the total theoretical sequence space of 7.17×10^8. 7.2×10^7 sequence variants were generated in silico by taking all possible combinations of the amino acids used per position in the combinatorial mutagenesis library designed from the DMS data following three rounds of enrichment for antigen binding variants; Alanine was also selected to be included at position 103. All in silico sequences were then classified as a binder or non-binder by the trained LSTM-RNN and CNN models. Sequences were selected for further analysis if they were classified in both models with a prediction probability (P) of more than 0.75.

[131] The Fv net charge and Fv charge symmetry parameter (FvCSP) were calculated as described by Sharma et al. Briefly, the net charge was determined by first solving the Henderson-Hasselbalch equation for each residue at a specified pH (here 5.5) with known amino acid pKas. The sum across all residues for both the VL and VH was then calculated as the Fv net charge. The FvCSP was calculated by taking the product of the VL and VH net charges. The hydrophobicity index (HI) was also calculated as described by Sharma et al., according to the following equation: $HI = -(\sum niEi / \sum njEj)$. E represents the Eisenberg value of an amino acid, n is the number of an amino acid, and i and j are hydrophobic and hydrophilic residues respectively.

[132] The protein solubility score was determined for each, full-length CDRH3 sequence (15 a.a.) padded with 10 amino acids on both the 5' and 3' ends (35 a.a.) by the CamSol method at pH 7.0.

[133] The binding affinities for a reference set of 26 HLA alleles were determined for each 15-mer contained within the 10 amino acid padded CDRH3 sequence (35 a.a.) by NetMHCIIpan 3.2. The output provides for each 15-mer a predicted affinity in nM and the % Rank which reflects the 15-mer's affinity compared to a set of random natural peptides. The % Rank measure is unaffected by the bias of certain molecules against stronger or weaker affinities and is used to classify peptides as weak or strong binders towards the specified MHC Class II allele. The minimum % Rank, the number of 15-mers with % Rank less than 10 (classification of

weak binder), and the average % Rank were calculated across all 21 15-mers for a single CDRH3 sequence across all 26 HLA alleles.

[134] Overall developability improvement of the antibody sequences was determined by first normalizing the FvCSP, CamSol score, and average NetMHCII % Rank according to the range of values observed in the remaining sequences post-filtering. The normalized CamSol protein solubility score was then weighted by a factor of 2 for its importance in determining developability. Lastly, the mean across these three parameters was taken to produce the overall developability improvement score. Since the sequences were filtered with the calculated values for trastuzumab, trastuzumab would have an overall developability improvement equal to 0.

$$Overall\ developability = \frac{1}{3}\left(\left(\frac{FvCSP - \min(FvCSP)}{\max(FvCSP) - \min(FvCSP)}\right) + 2 * \left(\frac{CamSol - \min(CamSol)}{\max(CamSol) - \min(CamSol)}\right) + \left(\frac{avgNetMHC - \min(avgNetMHC)}{\max(avgNetMHC) - \min(avgNetMHC)}\right)\right) \quad (3)$$

[135]

13) Expression and affinity measurements by biolayer interferometry

[136] Monoclonal populations of the individual variants were isolated by performing a single-cell sort. Following expansion, supernatant for all variants was collected and filtered through a 0.20 µm filter (Sartorius, 16534-K). Affinity measurements were then performed on an Octet RED96e (FortéBio) with the following parameters. Anti-human capture sensors (FortéBio, 18-5060) were hydrated in conditioned media diluted 1 in 2 with kinetics buffer (FortéBio, 18-1105) for at least 10 minutes before conditioning through 4 cycles of regeneration consisting of 10 seconds incubation in 10 mM glycine, pH 1.52 and 10 seconds in kinetics buffer. Conditioned sensors were then loaded with 0 ug/mL (reference sensor), 10 ug/mL trastuzumab (reference sample), or hybridoma supernatant (approximately 20 µg/mL) diluted 1 in 2 with kinetics buffer followed by blocking with mouse IgG (Rockland, 010-0102) at 50 µg/mL in kinetics buffer. After blocking, loaded sensors were equilibrated in

kinetics buffer and incubated with either 5 nM or 25 nM HER2 protein (Sigma-Aldrich, SRP6405-50UG). Lastly, sensors were incubated kinetics buffer to allow antigen dissociation. Antibody expression and kinetics analysis was performed in analysis software Data Analysis HT v11.0.0.50.

14) Thermal stability measurements by fluorescence

[137] Monoclonal antibodies of the individual variants were purified by Protein A column chromatography from the supernatant of their respective monoclonal cell line and eluted into 200 mM sodium dihydrogen phosphate, 140 mM sodium chloride, pH 2.5. Protein purity was verified by SDS-PAGE prior to downstream analysis. Purified antibody was loaded into Unchained Lab's UNcle instrument and static light scattering (SLS) and fluorescence measurements were taken while exposing the antibody to a thermal ramp from 20°C to 95°C at a rate of 0.5°C per minute. The melting temperature (Tm) is identified as the inflection point of the first derivative of the barycentric mean (BCM) as a function of the temperature.

15) Immunogenicity risk assessment by T-cell proliferation assay

[138] Immunogenicity risk was assessed by ProImmune's ProMap® T Cell Proliferation assay. Briefly, 15-mer peptides for specified variant sequences were synthesized and used for the in vitro assessment of potential antigenicity. Each 15-mer peptide is pulsed into donor antigen presenting cells which are then co-cultured with the donor's CD4+ T cells. CD4+ T cell proliferation is then measured by flow cytometry. The assay was performed by testing the peptides against 20 healthy donor cell samples. Donor cell samples were CD8-depleted prior to use, to eliminate CD8+ responses from the analysis. Detection of proliferation of CD4+ T cells was performed by labeling cells with CFSE and co-staining with anti-human CD4 antibody.

[139] Where technical features in the drawings, detailed description or any claim are followed by reference signs, the reference signs have been included to increase the intelligibility of the drawings, detailed description, and claims. Accordingly, neither the reference signs nor their absence has any limiting effect on the scope of any claim elements.

[140] The systems and methods described herein may be embodied in other specific forms without departing from the characteristics thereof. The foregoing implementations are illustrative rather than limiting of the described systems and methods. Scope of the systems and methods described herein is thus indicated by the appended claims, rather than the foregoing description, and changes that come within the meaning and range of equivalency of the claims are embraced therein.

[141] Having now described some illustrative implementations, it is apparent that the foregoing is illustrative and not limiting, having been presented by way of example. In particular, although many of the examples presented herein involve specific combinations of method acts or system elements, those acts and those elements may be combined in other ways to accomplish the same objectives. Acts, elements, and features discussed in connection with one implementation are not intended to be excluded from a similar role in other implementations or implementations.

[142] The phraseology and terminology used herein is for the purpose of description and should not be regarded as limiting. The use of "including" "comprising" "having" "containing" "involving" "characterized by" "characterized in that" and variations thereof herein, is meant to encompass the items listed thereafter, equivalents thereof, and additional items, as well as alternate implementations consisting of the items listed thereafter exclusively. In one implementation, the systems and methods described herein consist of one, each combination of more than one, or all of the described elements, acts, or components.

[143] As used herein, the term "about" and "substantially" will be understood by persons of ordinary skill in the art and will vary to some extent depending upon the context in which it is used. If there are uses of the term which are not clear to persons of ordinary skill in the art given the context in which it is used, "about" will mean up to plus or minus 10% of the particular term.

[144] Any references to implementations or elements or acts of the systems and methods herein referred to in the singular may also embrace implementations including a plurality of these

elements, and any references in plural to any implementation or element or act herein may also embrace implementations including only a single element. References in the singular or plural form are not intended to limit the presently disclosed systems or methods, their components, acts, or elements to single or plural configurations. References to any act or element being based on any information, act or element may include implementations where the act or element is based at least in part on any information, act, or element.

[145] Any implementation disclosed herein may be combined with any other implementation or embodiment, and references to "an implementation," "some implementations," "one implementation" or the like are not necessarily mutually exclusive and are intended to indicate that a particular feature, structure, or characteristic described in connection with the implementation may be included in at least one implementation or embodiment. Such terms as used herein are not necessarily all referring to the same implementation. Any implementation may be combined with any other implementation, inclusively or exclusively, in any manner consistent with the aspects and implementations disclosed herein.

[146] The indefinite articles "a" and "an," as used herein in the specification and in the claims, unless clearly indicated to the contrary, should be understood to mean "at least one."

[147] References to "or" may be construed as inclusive so that any terms described using "or" may indicate any of a single, more than one, and all of the described terms. For example, a reference to "at least one of 'A' and 'B'" can include only 'A', only 'B', as well as both 'A' and 'B'. Such references used in conjunction with "comprising" or other open terminology can include additional items.

[148] The terminology used in the description herein is for the purpose of describing particular embodiments only and is not intended to be limiting of the invention. All publications, patent applications, patents and other references mentioned herein are incorporated by reference in their entirety.

WHAT IS CLAIMED:

1. A method, comprising:

providing an input amino acid sequence that represents an antigen binding portion of an antigen binding molecule;

generating a first training data set comprising a first plurality of variant sequences, each of the first plurality of variant sequences comprising a single site mutation in the input amino acid sequence of the antigen binding molecule;

generating a second training data set comprising a second plurality of sequences, each of the second plurality of sequences comprising a plurality of variants at positions based on enrichment scores of the first training data set comprising the first plurality of variant sequences;

providing the second training data set to a classification engine comprising a first machine learning model to generate a plurality of weights and biases for the first machine learning model;

determining, by the classification engine based on the plurality of weights and bias for the first machine learning model, a first affinity binding score for a proposed amino acid sequence to an antigen; and

selecting the proposed amino acid sequence for expression based on the first affinity binding score satisfying a threshold.

2. The method of claim 1, wherein the antigen binding molecule comprises an antibody, or an antigen binding fragment thereof.

3. The method of claim 1, wherein the antigen binding molecule comprises a chimeric antigen receptor.

4. The method of any one of claims 1 to 3, comprising:

determining, by the classification engine, a second affinity binding score for the proposed amino acid sequence using a second machine learning model of the classification engine; and

selecting the proposed amino acid sequence for expression based on the first affinity binding score and the second affinity binding score satisfying the threshold.

5. The method of any one of claims 1 to 4, comprising:

determining, by the classification engine, an affinity binding score for each of a plurality of proposed amino acid sequences;

determining, by a candidate selection engine, one or more parameters for each of the plurality of proposed amino acid sequences; and

selecting, by the candidate selection engine, candidate variants from the plurality of proposed amino acid sequences based on the affinity binding score and the one or more parameters for each of the plurality of proposed amino acid sequences.

6. The method of claim 5, wherein the candidate selection engine selects only the variants that were classified with a predetermined confidence or probability level.

7. The method of claim 6, wherein the predetermined confidence or probability level is above 0.5.

8. The method of any one of claims 5 to 7, wherein the candidate selection engine selects variants based on the proposed amino acid sequence satisfying a threshold for at least of one of the one or more additional parameters.

9. The method of claim 5, wherein the candidate selection engine selects variants based on the proposed amino acid sequence satisfying a threshold for each of the one or more additional parameters.

10. The method of claim 9, wherein one or more of the threshold values are a value threshold.

11. The method of claim 9 or 10, wherein one or more of the threshold values are a variable or relative threshold.

12. The method of any one of claims 9 to 11, wherein the threshold for one or more of the additional parameters is a parameter value in the top 5% or top 10%.

13. The method of any one of claims 9 to 12, wherein the threshold for one or more of the additional parameters is based on a number of standard deviations above the average for the one or more parameters.

14. The method of any one of claims 5 to 13, wherein the one or more parameters comprise viscosity values, solubility values, stability values, pharmacokinetic values, and/or immunogenicity values.

15. The method of any one of claims 5 to 14, wherein the one or more parameters comprise a Levenshtein distance value.

16. The method of any one of claims 5 to 15, wherein the one or more parameters comprise charge value.

17. The method of claim 16, wherein the charge value is a variable fragment (Fv) charge value.

18. The method of claim 17, wherein the Fv charge value is between about 0 and about 6.2.

19. The method of claim 16, wherein the charge value is a variable fragment charge symmetry parameter (FvCSP) value.

20. The method of claim 19, wherein the FvCSP value is greater than 0.

21. The method of any one of claims 5 to 20, wherein the one or more parameters comprise hydrophobicity index value.

22. The method of claim 21, wherein the hydrophobicity index sum value is less than 4.0.

23. The method of any one of claims 5 to 22, wherein the one or more parameters comprise a protein solubility score.

24. The method of claim 23, wherein the protein solubility score is a CamSol score.

25. The method of claim 23 or 24, wherein the protein solubility score is greater than 0.5.

26. The method of claim 25, wherein the protein solubility score is greater than 1.

27. The method of any one of claims 5 to 26, wherein the one or more parameters comprise minimum affinity rank.

28. The method of any one of claims 5 to 27, wherein the one or more parameters comprise average affinity rank.

29. The method of any one of claims 5 to 28, wherein the one or more parameters comprise sequence motifs associated with manufacturing liabilities.

30. The method of claim 29, wherein the one or more parameters comprise n-glycosylation sites.

31. The method of claim 29 or 30, wherein the one or more parameters comprise deamidation sites.

32. The method of any one of claims 29 to 31, wherein the one or more parameters comprise isomerization sites.

33. The method of any one of claims 29 to 32, wherein the one or more parameters comprise n-glycosylation methionine oxidation sites.

34. The method of any one of claims 29 to 33, wherein the one or more parameters comprise tryptophan oxidation sites.

35. The method of any one of claims 29 to 34, wherein the one or more parameters comprise paired or unpaired cysteine residues.

36. The method of any one of claims 5 to 35, wherein the one or more parameters comprise protein structured based metrics.

37. The method of claim 36, wherein the one or more parameters comprise solvent accessible surface area (SASA).

38. The method of claim 36 or 37, wherein the one or more parameters comprise patches positive charges (PPC).

39. The method of claim 38, wherein the PPC value is less than 1.

40. The method of any one of claims 36 to 39, wherein the one or more parameters comprise patches negative charges (PNC).

41. The method of claim 40, wherein the PNC value is less than 1.5.

42. The method of any one of claims 36 to 41, wherein the one or more parameters comprise patches surface hydrophobicity (PSH).

43. The method of claim 42, wherein the PSH value is between about 100 and about 150.

44. The method of any one of claims 36 to 43, wherein the one or more parameters comprise surface Fv charge symmetry parameter (SFvCSP).

45. The method of claim 44, wherein the SFvCSP value is greater than 0.

46. The method of any one of claims 5 to 45, wherein the candidate selection engine calculates an affinity binding score for binding of the candidate variants to MHC Class II molecules.

47. The method of claim 46, wherein the MHC Class II molecules comprise MHC class II isotypes HLA-DR, HLA-DP and HLA-DQ.

48. The method of claim 46 or 47, wherein the affinity binding rank for binding of the candidate variants to MHC II molecules is a NetNHCII % Rank.

49. The method of claim 48, wherein the NetMHCII % Rank has a threshold value of greater than 10%.

50. The method of any one of claims 1 to 49, wherein the first machine learning model comprises a recurrent neural network (RNN).

51. The method of any one of claims 1 to 49, wherein the first machine learning model comprises a convolution neural network (CNN).

52. The method of any one of claims 1 to 49, wherein the first machine learning model comprises a standard artificial neural network (ANN).

53. The method of any one of claims 1 to 49, wherein the first machine learning model comprises a support vector machine (SVM).

54. The method of any one of claims 1 to 49, wherein the first machine learning model comprises a random forest ensemble (RF).

55. The method of any one of claims 1 to 49, wherein the first machine learning model comprises a logistic regression model (LR).

56. The method of any one of claims 2 and 4-55, wherein the input amino acid sequence is a portion of a complementarity determining region (CDR) of the antibody.

57. The method of claim 56, wherein the input amino acid sequence comprises a CDRH3 sequence.

58. The method of claim 56 or 57, wherein the input amino acid sequence comprises a CDRH1 sequence.

59. The method of any one of claims 56 to 58, wherein the input amino acid sequence comprises a CDRH2 sequence.

60. The method of any one of claims 56 to 59, wherein the input amino acid sequence comprises a CDRL1 sequence.

61. The method of any one of claims 56 to 60, wherein the input amino acid sequence comprises a CDRL2 sequence.

62. The method of any one of claims 56 to 61, wherein the input amino acid sequence comprises a CDRL3 sequence.

63. The method of any one of claims 2 and 4 to 62, wherein the input amino acid sequence comprises a framework domain, or a region within the framework domain of the antibody.

64. The method of claim 63, wherein the input amino acid sequence comprises FR1, FR2, FR3 or FR4 of the antibody.

65. The method of any one of claims 2 and 4 to 64, wherein the input amino acid sequence comprises a constant domain, or a region with a constant domain, of the antibody.

66. The method of any one of claims 2 and 4 to 65, wherein the input amino acid sequence comprises a full length heavy chain sequence of the antibody.

67. The method of any one of claims 2 and 4 to 66, wherein the input amino acid sequence comprises a full length light chain sequence of the antibody.

68. The method of any one of claims 2 and 4 to 67, wherein the antibody is a therapeutic antibody.

69. The method of claim 68, wherein the therapeutic antibody is selected from abciximab (Reopro); adalimumab (Humira, Amjevita); alefacept (Amevive); alemtuzumab (Campath); basiliximab (Simulect); belimumab (Benlysta); bezlotoxumab (Zinplava); canakinumab (Ilaris); certolizumab pegol (Cimzia); cetuximab (Erbitux); daclizumab (Zenapax, Zinbryta); denosumab (Prolia, Xgeva); efalizumab (Raptiva); golimumab (Simponi, Simponi Aria); inflectra (Remicade); ipilimumab (Yervoy); ixekizumab (Taltz); natalizumab (Tysabri); nivolumab (Opdivo); olaratumab (Lartruvo); omalizumab (Xolair); palivizumab (Synagis); panitumumab (Vectibix); pembrolizumab (Keytruda); rituximab (Rituxan); tocilizumab (Actemra); trastuzumab (Herceptin); secukinumab (Cosentyx); and ustekinumab (Stelara).

70. The method of any one of claims 1 to 69, wherein the first training data set is generated by deep mutational scanning.

71. The method of claim 70, wherein deep mutational scanning comprises generating a first library of variant sequences wherein each variant sequence is modified at a single amino acid position relative to the input amino acid sequence.

72. The method of claim 71, wherein the first library comprises variant sequences representing each amino acid position of the input amino acid sequence.

73. The method of claim 71 or 72, wherein the first library comprises variant sequences representing all 20 standard amino acids at each position of the input amino acid sequence.

74. The method of any one of claims 71 to 73, wherein the first library of variant sequences is generated by mutagenesis of an nucleic acid encoding the input amino acid sequence.

75. The method of any one of claims 71 to 74, wherein the first library of variant sequences is generated by high throughput mutagenesis in a mammalian cell.

76. The method of claim 75, wherein the high throughput mutagenesis comprises error-prone PCR, recombination mutagenesis, alanine scanning mutagenesis, structure-guided mutagenesis, or homology-directed repair (HDR).

77. The method of claim 76, wherein the first library of variant sequences is generated by CRISPR/Cas9-mediated homology-directed repair (HDR).

78. The method of any one of claims 70 to 77, wherein deep mutational scanning comprises generating a plurality of antibodies comprising the first library of variant sequences.

79. The method of claim 78, wherein deep mutational scanning further comprises screening the plurality of antibodies comprising the first library of variant sequences for binding to an antigen and determining a sequence of variants selected for binding to the antigen, thereby obtaining the first training data set.

80. The method of any one of claims 1 to 79, wherein the second training data set is generated by deep mutational scanning-guided combinatorial mutagenesis.

81. The method of claim 80, wherein deep mutational scanning-guided combinatorial mutagenesis comprises generating a second library of variant sequences wherein each variant sequence is modified at two or more amino acid positions based on the first training data set.

82. The method of claim 81, wherein the second library of variant sequences is generated by mutagenesis of an nucleic acid encoding the first training data of input amino acid sequences.

83. The method of claim 81 or 82, wherein the second library of variant sequences is generated by high throughput mutagenesis in a mammalian cell.

84. The method of any one of claims 81 to 83, wherein the second librarylibrary of variant sequences is generated by CRISPR/Cas9-mediated homology-directed repair (HDR).

85. The method of any one of claims 81 to 84, wherein deep mutational scanning-guided combinatorial mutagenesis comprises generating a plurality of antibodies comprising the second library of variant sequences.

86. The method of claim 85, wherein combinatorial deep mutational scanning further comprises screening the plurality of antibodies comprising the second library of variant sequences for binding to the antigen and determining the sequence of variants selected for binding to the antigen, thereby obtaining the second training data set.

87. The method of any one of claims 5 to 86, wherein the candidate variants have one or more parameter values equal to or greater than the input amino acid sequence.

88. A system comprising one or more processors and a memory storing processor-executable instructions, the one or more processors execute the processor-executable instructions to:

receive an input amino acid sequence that represents an antigen binding portion of an antibody;

receive a first training data set comprising a first plurality of variant sequences, each of the first plurality of variant sequences comprising a single site mutation in the input amino acid sequence of the antibody;

receive a second training data set comprising a second plurality of sequences, each of the second plurality of sequences comprising a plurality of variants at positions based on enrichment scores of the first training data set comprising the first plurality of variant sequences;

provide the second training data set to a classification engine comprising a first machine learning model to generate a plurality of weights and bias for the first machine learning model;

determine, based on the plurality of weights and bias for the first machine learning model, a first affinity binding score for a proposed amino acid sequence to an antigen; and

select the proposed amino acid sequence for expression based on the first affinity binding score satisfying a threshold.

89. A protein or peptide, wherein the amino acid sequence of the protein or peptide is generated by the method of any one of claims 1 to 87, or the system of claim 88.

90. The protein or peptide of claim 89, wherein the protein or peptide binds to an antigen.

91. The protein or peptide of claim 90, wherein the protein or peptide is a chimeric antigen receptor.

92. The protein or peptide of claim 89 or 90, wherein the amino acid sequence comprises a CDRH3 sequence.

93. The protein or peptide of claim 89, 90 or 92, wherein the protein or peptide comprises an antibody or antigen binding fragment thereof.

94. The protein or peptide of claim 93, wherein the protein or peptide is a fusion protein comprising one or more portions of an antibody.

95. The protein or peptide of any one of claims 89 to 94, wherein the protein or peptide comprises an scFv or an Fc fusion protein.

96. The protein or peptide of any one of claims 90 to 95, wherein the antigen is associated with a disease or condition.

97. The protein or peptide of claim 96, wherein the antigen is a tumor antigen.

98. The protein or peptide of claim 96, wherein the antigen is an anti-inflammatory antigen.

99. The protein or peptide of claim 96, wherein the antigen is a parasitic antigen.

100. The protein or peptide of any one of claims 89 to 99, wherein the protein or peptide has one or more improved properties compared to a protein or peptide comprising the input amino acid sequence.

101. The protein or peptide of any one of claims 89 to 100, wherein the protein or peptide has improved biophysical properties for manufacturing compared to a protein or peptide comprising the input amino acid sequence.

102. The protein or peptide of any one of claims 89 to 101, wherein the protein or peptide has improved affinity for an antigen compared to a protein or peptide comprising the input amino acid sequence.

103. The protein or peptide of any one of claims 89 to 102, wherein the protein or peptide has reduced immunogenic risk compared to a protein or peptide comprising the input amino acid sequence.

104. A protein or peptide, comprising an amino acid sequence depicted in FIGURE 15A to 15D or in FIGURE 23A to 23O.

105. A protein or peptide according to claim 104, wherein the protein or peptide comprises an antibody, or antigen binding fragment thereof.

106. A protein or peptide according to claim 105, wherein the protein or peptide comprises a full length antibody.

107. A protein or peptide according to claim 104 or 105, wherein the protein or peptide comprises an scFv or an Fc fusion protein.

108. A protein or peptide according to claim 104, wherein the protein or peptide comprises a chimeric antigen receptor.

109. A protein or peptide according to any one of claims 104 to 108, wherein the protein or peptide is a fusion protein.

110. A protein or peptide according to any one of claims 89 to 109, wherein the protein or peptide binds to HER2 (human epidermal growth factor receptor 2).

111. A protein or peptide according to claim 110, wherein the protein or peptide has improved affinity for the HER2 antigen compared to the trastuzumab (Herceptin) antibody.

112. A cell comprising a protein or peptide according to any one of claims 89 to 111.

113. A cell comprising a nucleic acid sequence encoding a protein or peptide according to any one of claims 89 to 112.

114. The cell according to claim 112 or 113, wherein the cell is a mammalian cell, a bacterial cell, a yeast cell, an insect cell, or a eukaryotic cell.

115. The cell according to any one of claims 112 to 114, wherein the cell is an immune cell.

116. The cell according to claim 115, wherein the immune cell is a T cell.

117. The cell according to claim 116, wherein the T cell is a CAR-T cell.

118. The protein or peptide of any one of claims 89 to 111 or a cell according to any one of claims 112 to 117, wherein the protein or peptide, or cell, is administered to a subject to treat an inflammatory disease, infectious disease, cancer, genetic disorder, organ transplant rejection, autoimmune disease or an immunological disorder.

119. The protein or peptide of any one of claims 89 to 111 or a cell according to any one of claims 112 to 117, wherein the protein or peptide, or cell, is administered to a subject to treat a HER2 positive cancer.

120. The protein or peptide of any one of claims 89 to 111 or a cell according to any one of claims 112 to 117, wherein the protein or peptide, or cell, is used for the manufacture of a medicament to treat an inflammatory disease, infectious disease, cancer, genetic disorder, organ transplant rejection, autoimmune disease or an immunological disorder.

121. The protein or peptide of any one of claims 89 to 111 or a cell according to any one of claims 112 to 117, wherein the protein or peptide, or cell, is used for the manufacture of a medicament to treat a HER2 positive cancer.

122. The use of the protein or peptide of any one of claims 89 to 111, for detecting an antigen in a biological sample.

123. The use of the protein or peptide of any one of claims 89 to 111, for detecting an antigen in a subject *in vivo*.

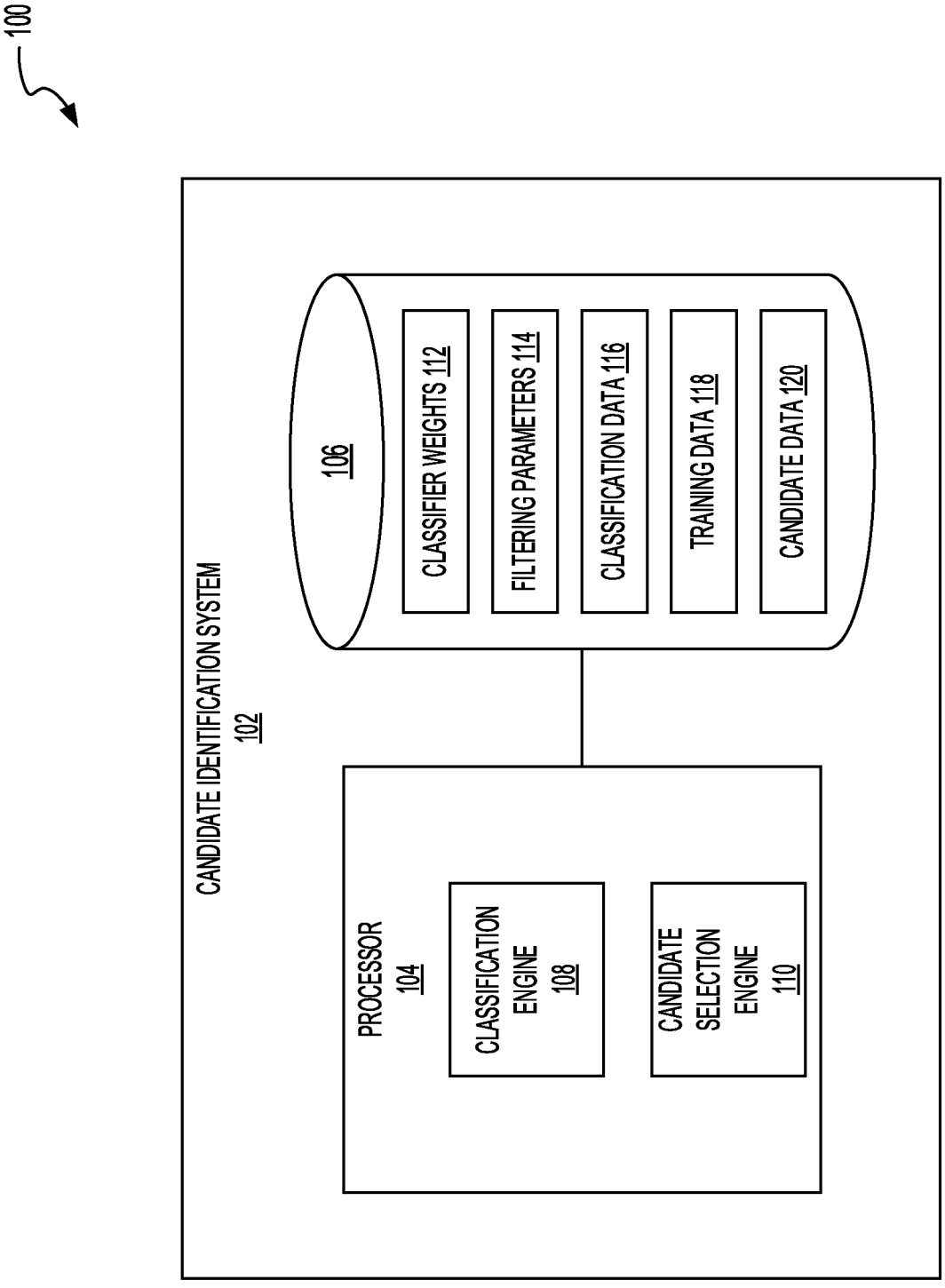

100
CANDIDATE IDENTIFICATION SYSTEM
102
106
CLASSIFIER WEIGHTS 112
FILTERING PARAMETERS 114
CLASSIFICATION DATA 116
TRAINING DATA 118
CANDIDATE DATA 120
PROCESSOR
104
CLASSIFICATION ENGINE
108
CANDIDATE SELECTION ENGINE
110

FIG. 1

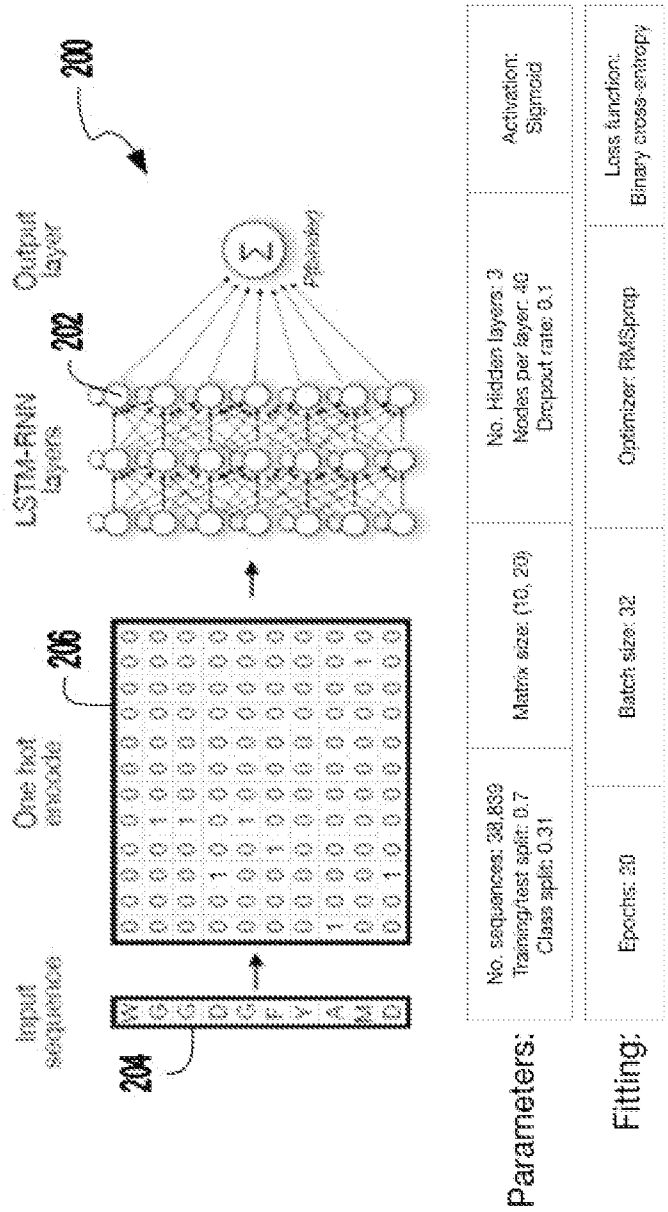

200
Output layer
Polarity
202
LSTM-RNN layers
One hot encode
206
Input sequences
204
Parameters:
No. sequences: 38,859
Training/test split: 0.7
Class split: 0.31
Matrix size: (10, 20)
No. Hidden layers: 2
Nodes per layer: 40
Dropout rate: 0.1
Activation: Sigmoid
Fitting:
Epochs: 30
Batch size: 32
Optimizer: RMSprop
Loss function: Binary cross-entropy

FIG. 2A

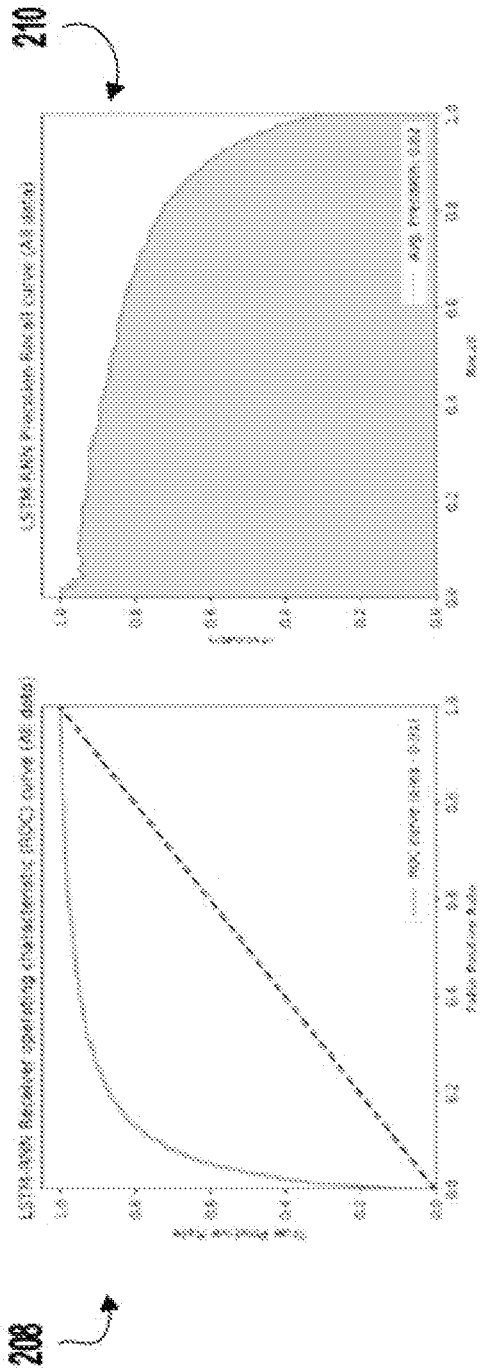


FIG. 2B

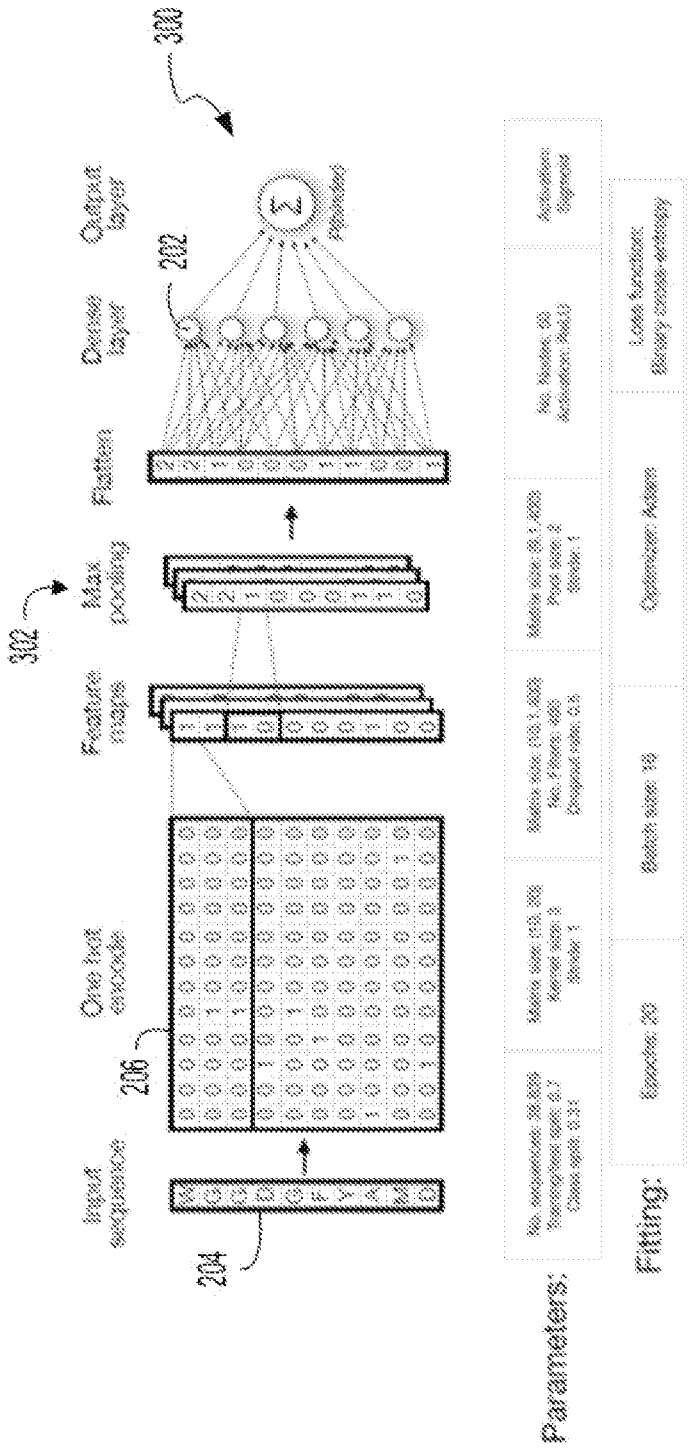

300
302
Input sequence
One hot encode
206
204
Feature maps
Max pooling
Flatten
Dense layer
202
Output layer
Parameters:
Fitting:
Optimizer: Adam

FIG. 3A

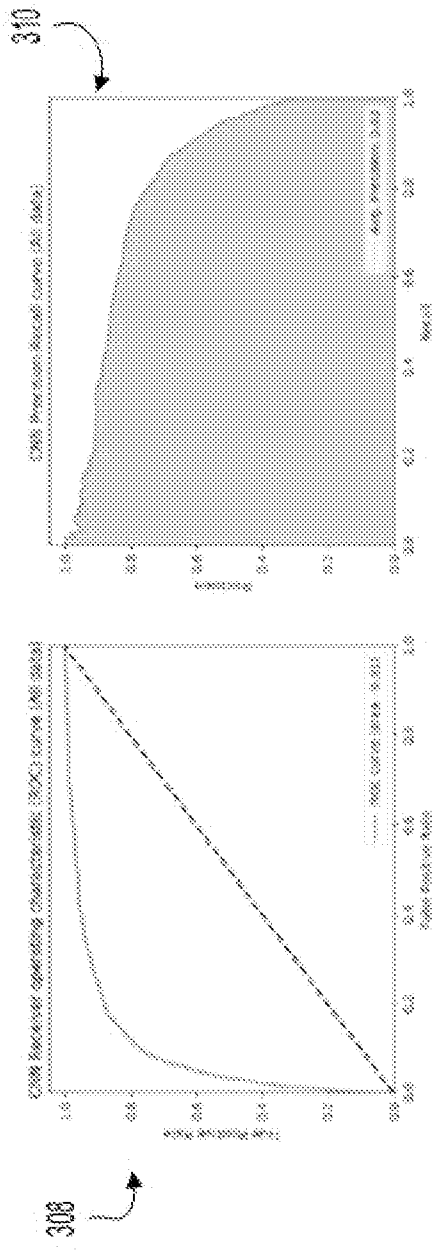


FIG. 3B

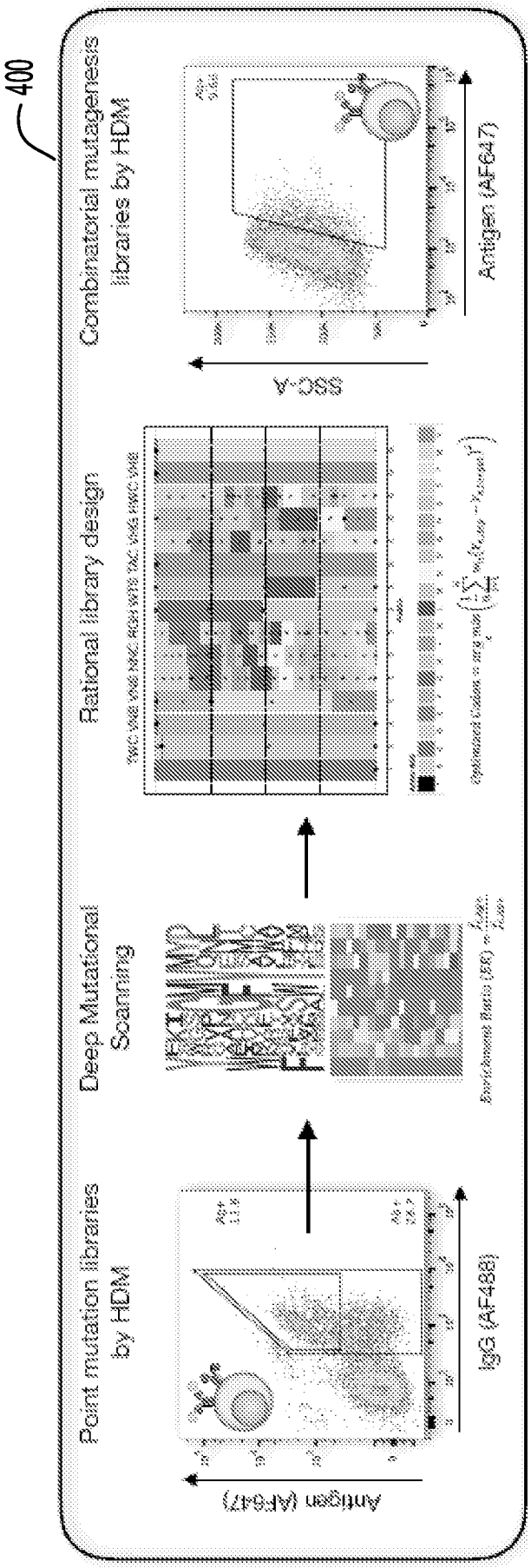

400
Point mutation libraries by HDM
Deep Mutational Scanning
Rational library design
Combinatorial mutagenesis libraries by HDM
Antigen (AF647)
IgG (AF488)
SSC-A
Antigen (AF647)

FIG. 4A

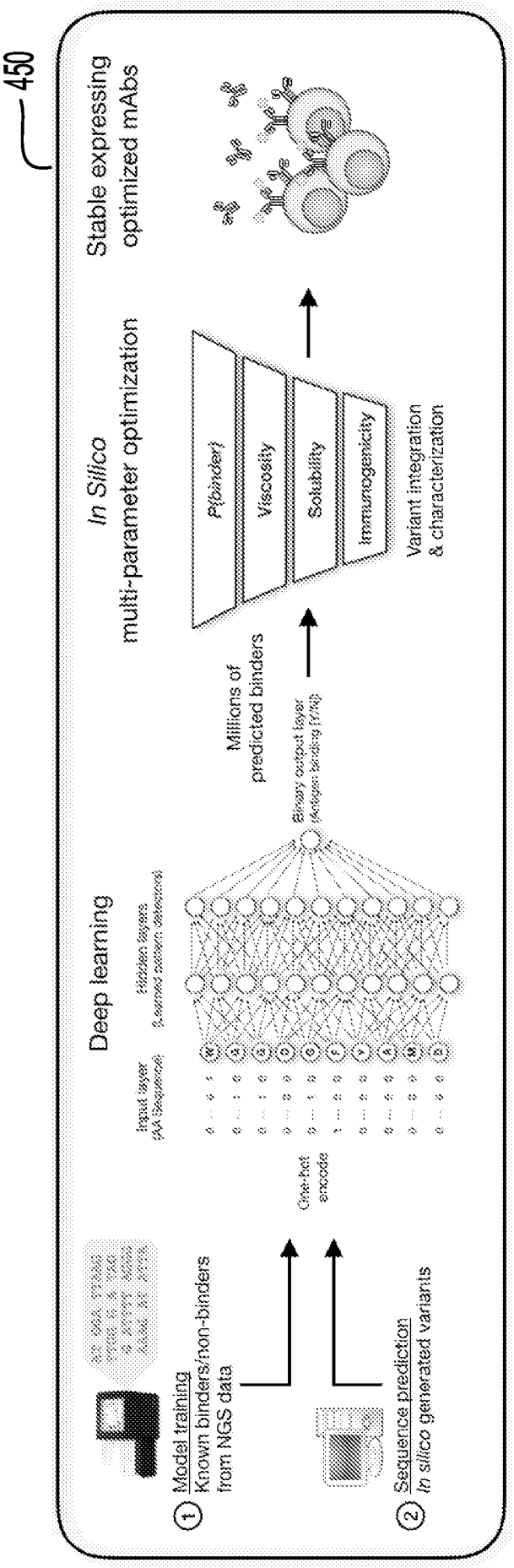

450
Deep learning
In Silico
multi-parameter optimization
Stable expressing
optimized mAbs
Input layer
(AA Sequence)
Hidden layers
(Learned pattern detectors)
Binary output layer
(Antigen binding (Y/N))
Millions of
predicted binders
P(binder)
Viscosity
Solubility
Immunogenicity
Variant integration
& characterization
One-hot
encode
1 Model training
Known binders/non-binders
from NGS data
2 Sequence prediction
In silico generated variants

FIG. 4B

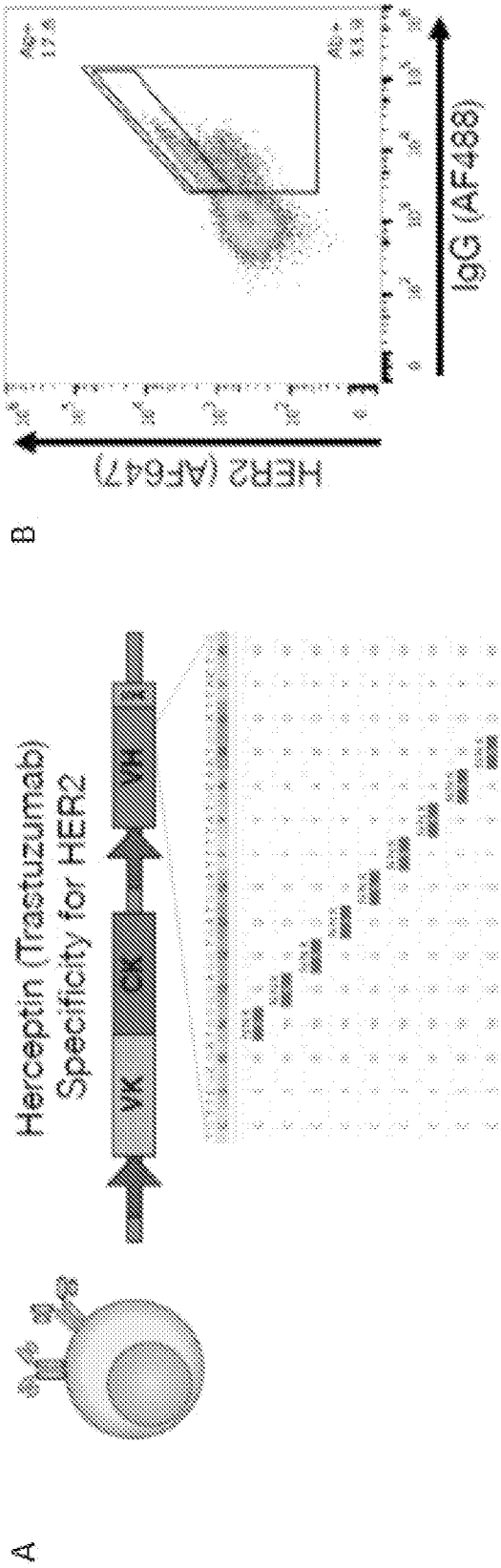

A
Herceptin (Trastuzumab)
Specificity for HER2
VK
CK
VH
H
B
HER2 (AF647)
IgG (AF488)

FIG. 5A

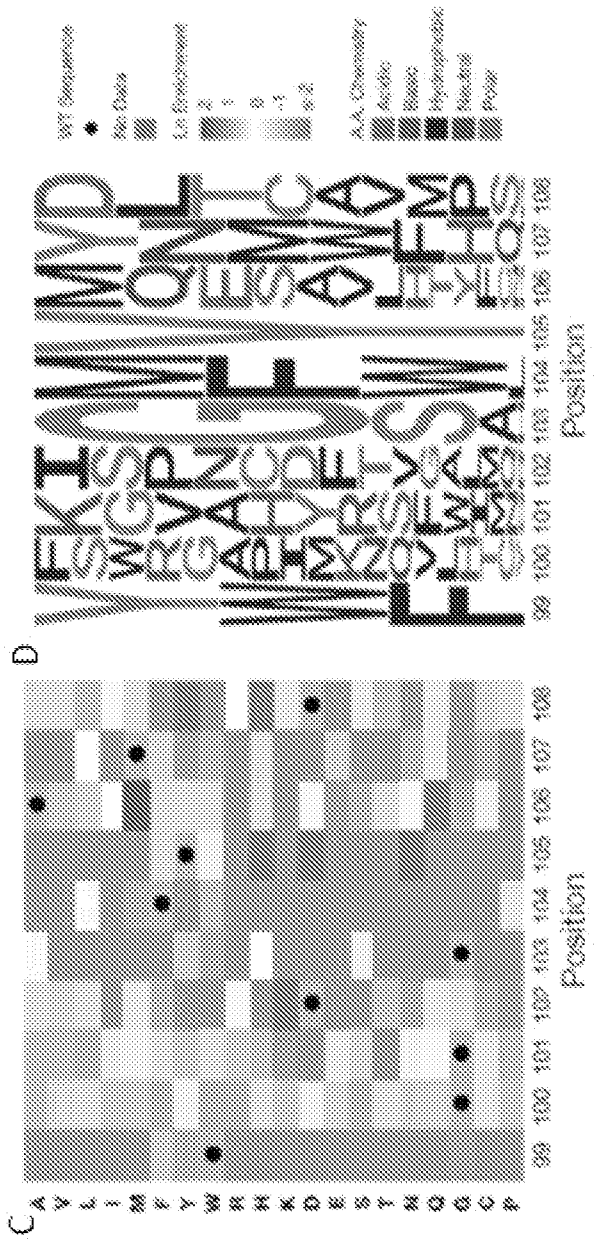


FIG. 5B

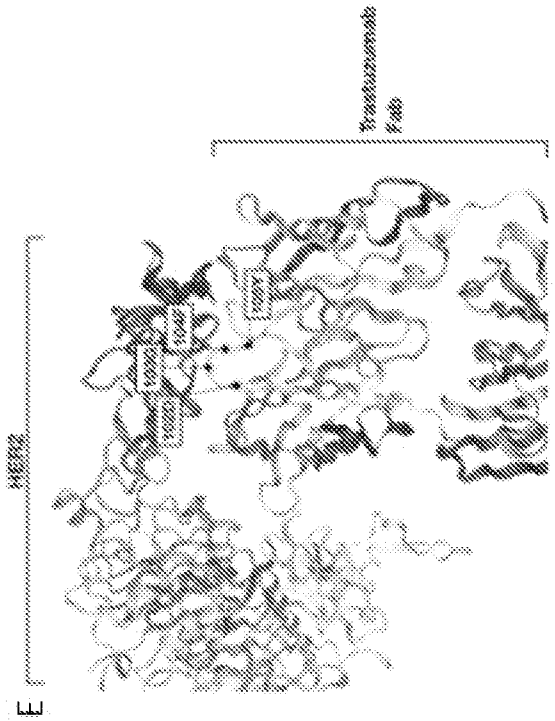


FIG. 5C

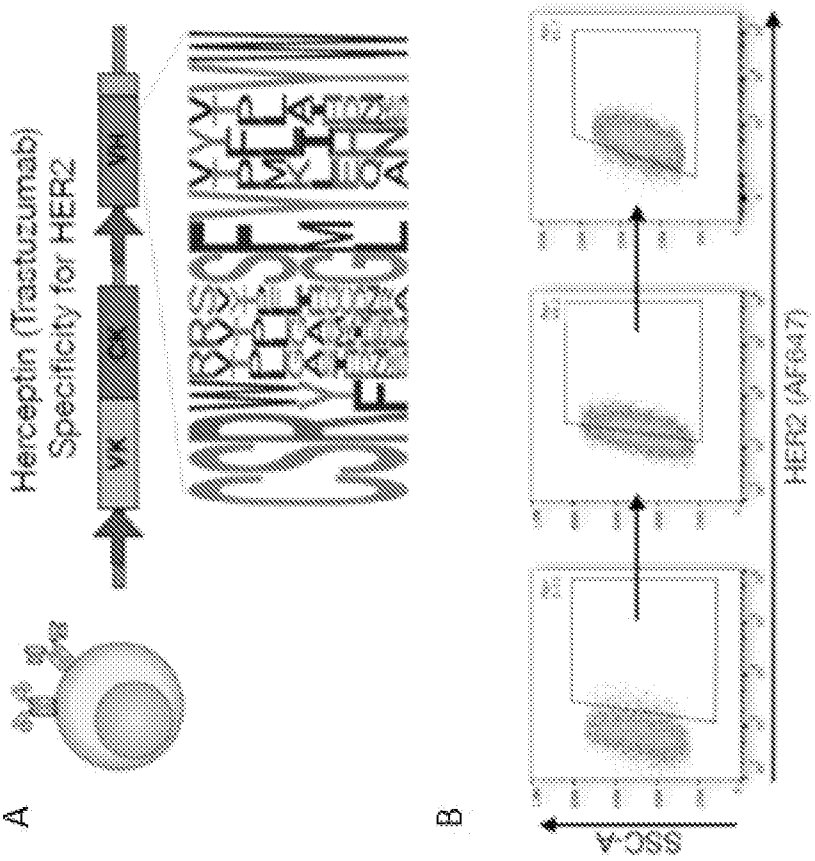

A
Herceptin (Trastuzumab)
Specificity for HER2
VH
CH
VL
CL
B
SSC-A
HER2 (AF647)

FIG. 6A

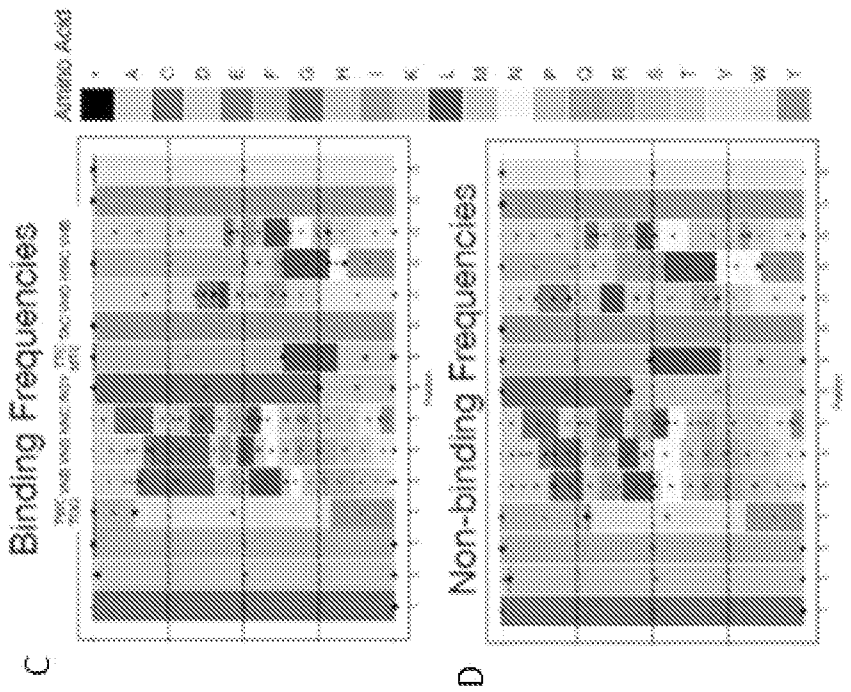

C
Binding Frequencies
D
Non-binding Frequencies
Amino Acid

FIG. 6B

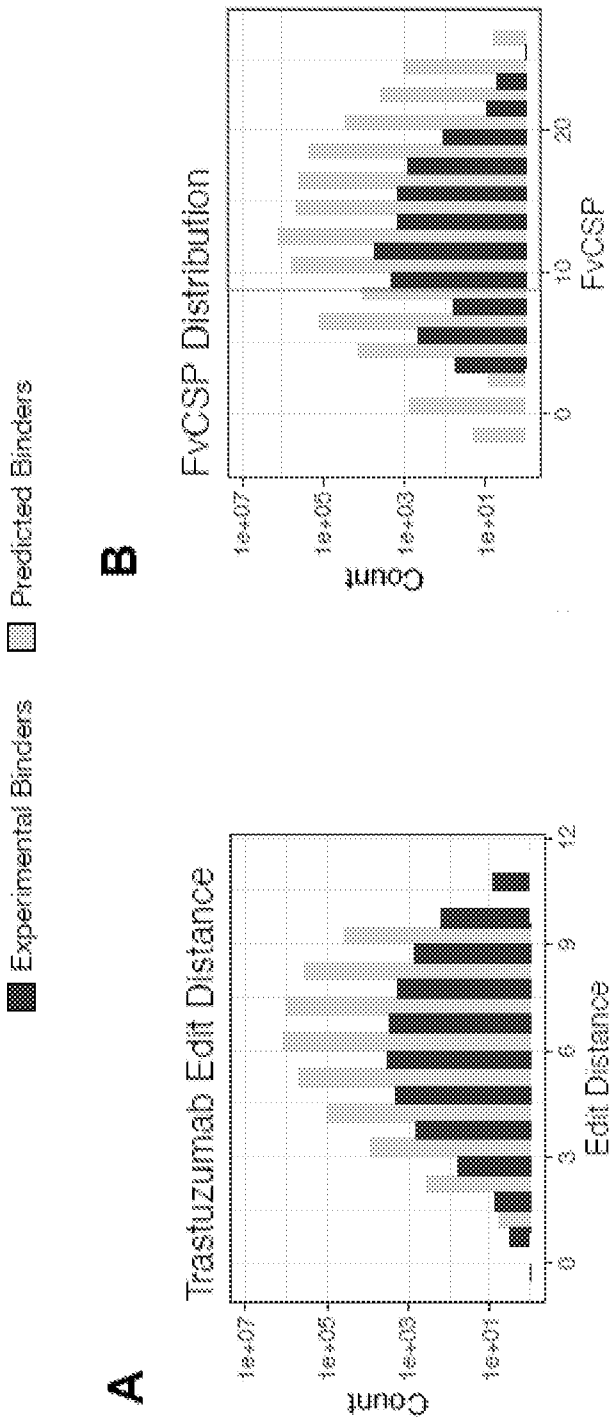

A
Trastuzumab Edit Distance
Experimental Binders
Predicted Binders
Count
1e+07
1e+05
1e+03
1e+01
0
3
6
9
12
Edit Distance
B
FvCSP Distribution
Count
1e+07
1e+05
1e+03
1e+01
0
10
20
FvCSP

FIG. 7A

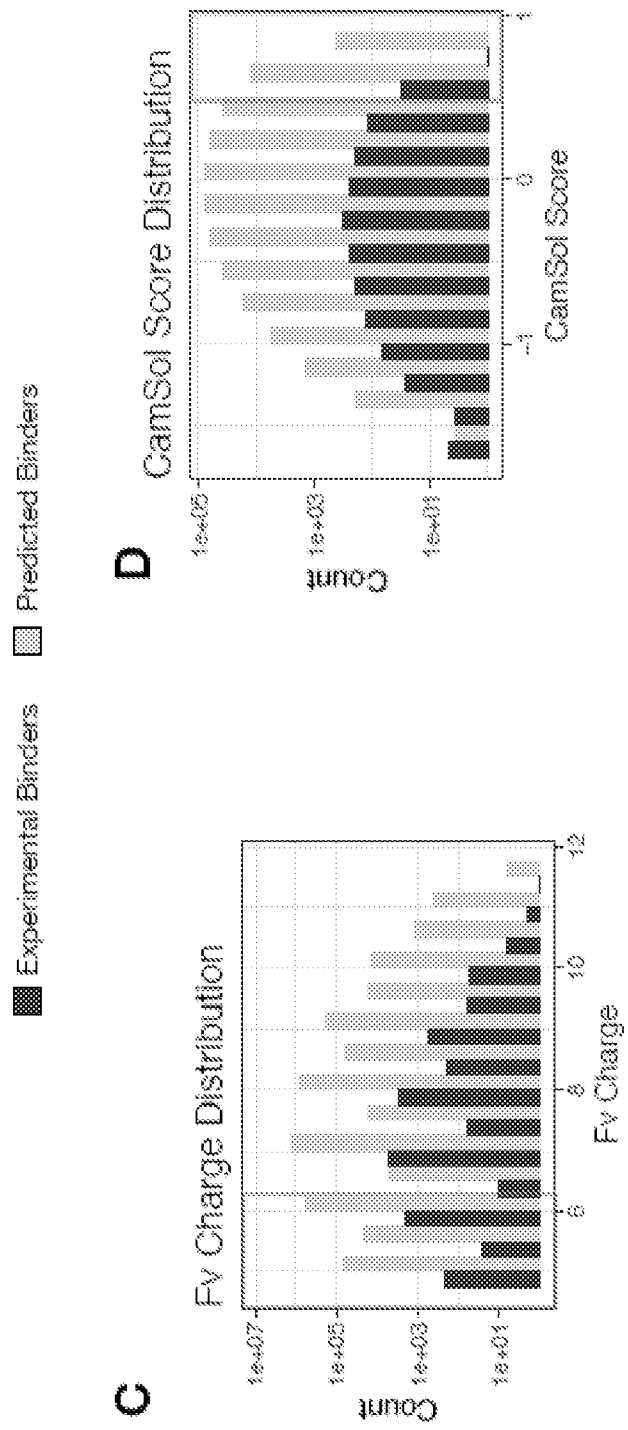

Experimental Binders
Predicted Binders
C
Fv Charge Distribution
Count
1e+07
1e+05
1e+03
1e+01
6
8
10
12
Fv Charge
D
CamSol Score Distribution
Count
1e+05
1e+03
1e+01
−1
0
1
CamSol Score

FIG. 7B

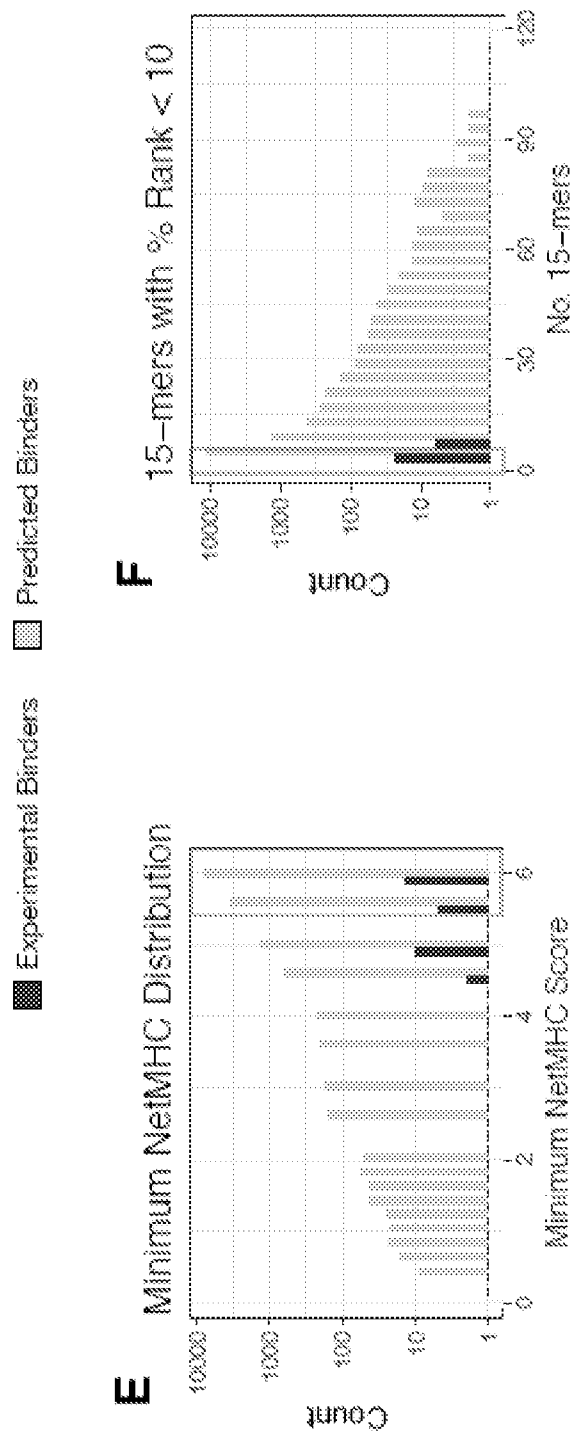

Experimental Binders
Predicted Binders
E
Minimum NetMHC Distribution
Count
10000
1000
100
10
1
0
2
4
6
Minimum NetMHC Score
F
15-mers with % Rank < 10
Count
10000
1000
100
10
1
0
30
60
90
120
No. 15-mers

FIG. 7C

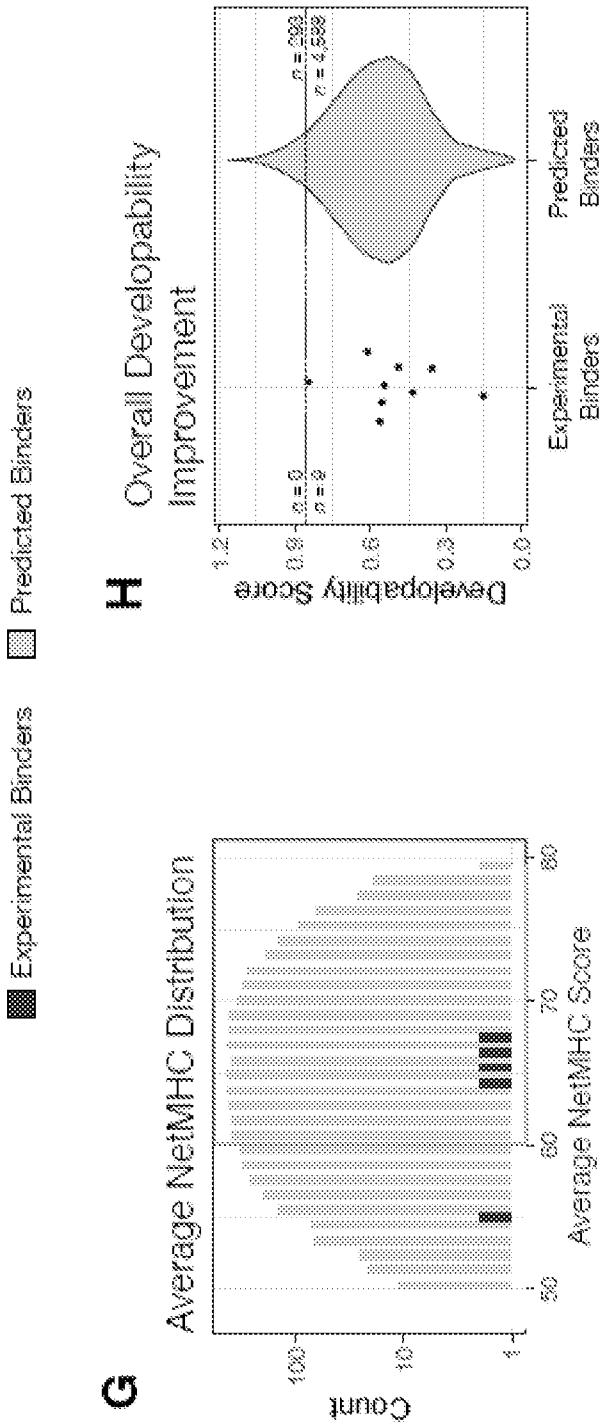

G
Experimental Binders
Predicted Binders
Average NetMHC Distribution
Count
100
10
1
50
60
70
80
Average NetMHC Score
H
Overall Developability Improvement
Developability Score
1.2
0.9
0.6
0.3
0.0
n = 0
n = 0
n = 293
n = 4,589
Experimental Binders
Predicted Binders

FIG. 7D

Experimental Binders   Predicted Binders

Filtering Parameters	No. Predicted Binders
RNN P(binder) > 0.5 CNN P(binder) > 0.5	8,552,268 9,519,642
RNN P(binder) > 0.75 CNN P(binder) > 0.75	4,316,323 5,218,706
RNN, CNN consensus	3'159'373
FvCSP > 6.61 Fv charge < 6.2 HI Sum > 0, < 4	402'633
Solubility score > 0.5	14'126
Minimum % Rank > 5.5 No. 15-mers w/ % Rank < 10 ≤ 2 Average % Rank > 60.8	4'881

FIG. 7E

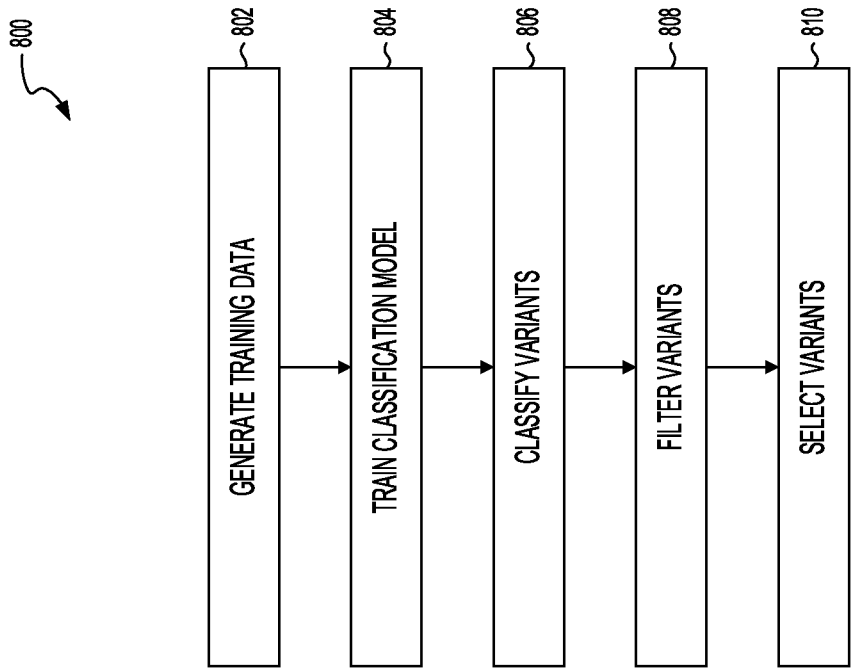

800
802
GENERATE TRAINING DATA
804
TRAIN CLASSIFICATION MODEL
806
CLASSIFY VARIANTS
808
FILTER VARIANTS
810
SELECT VARIANTS

FIG. 8

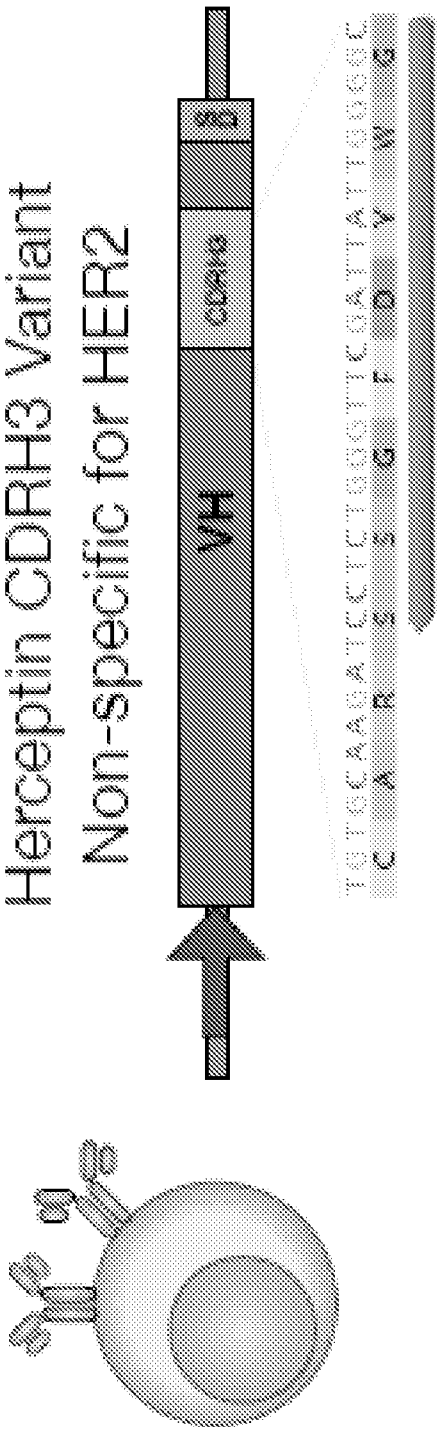

Herceptin CDRH3 Variant
Non-specific for HER2
VH
CDRH3
WG
TGTGCAAGATCCTCTGGTTTCGATTATTGGGGC
C A R S S G F D Y W G

FIG. 9A

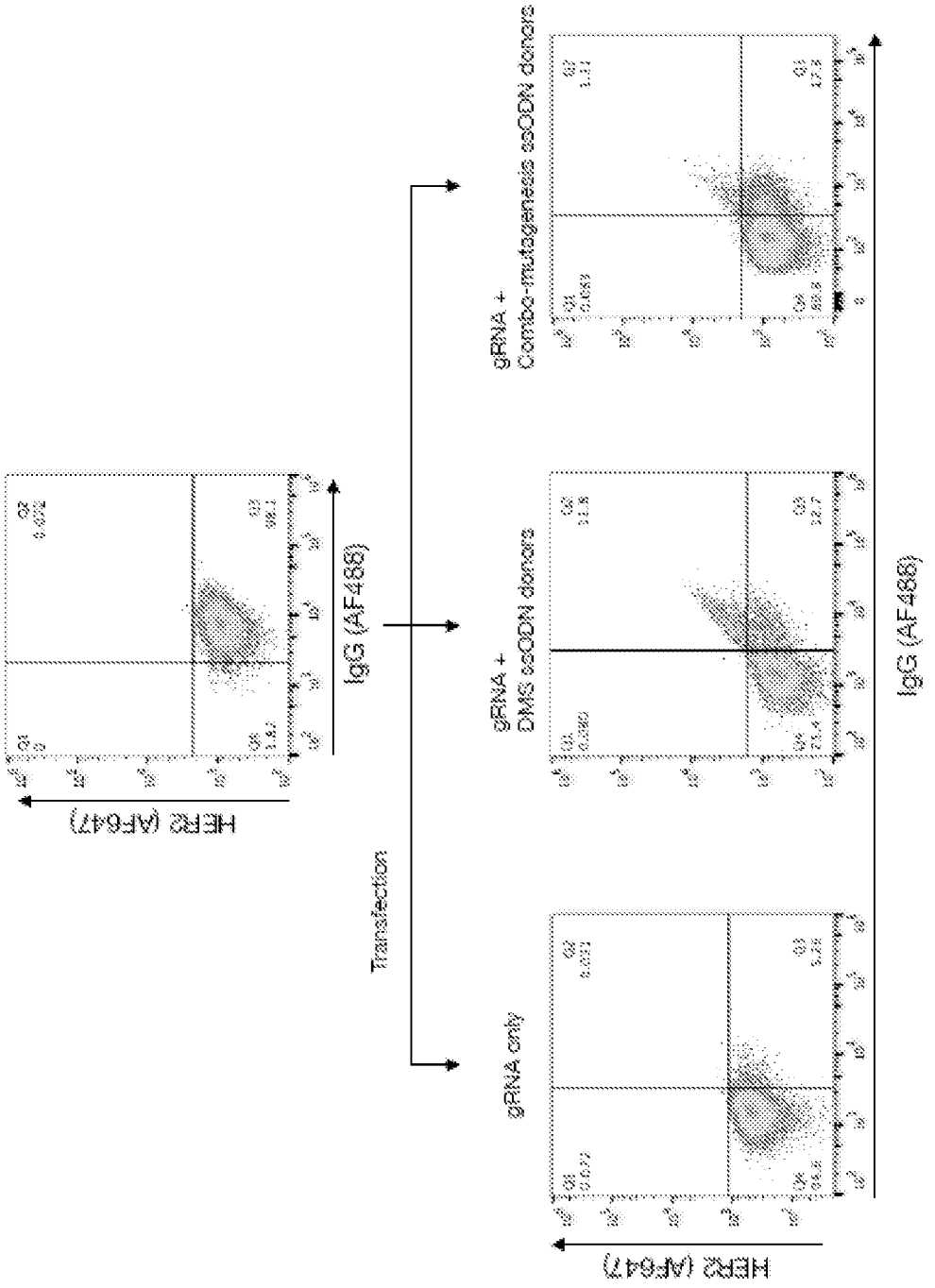

HER2 (AF647)
IgG (AF488)
Transfection
gRNA only
gRNA +
DMS ssODN donors
gRNA +
Combo-mutagenesis ssODN donors
IgG (AF488)

FIG. 9B

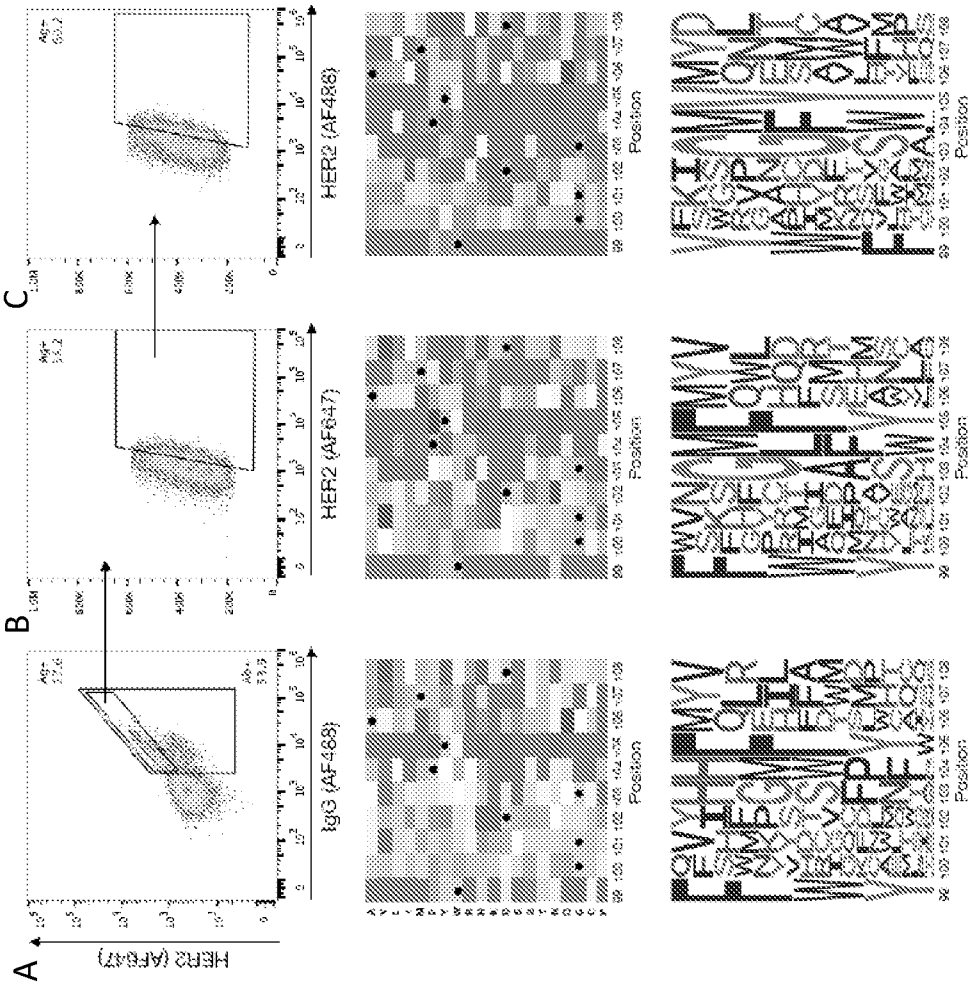

A
B
C
HER2 (AF647)
HER2 (AF488)
Ig3 (AF488)
HER2 (AF647)
HER2 (AF488)
Position

FIG. 10

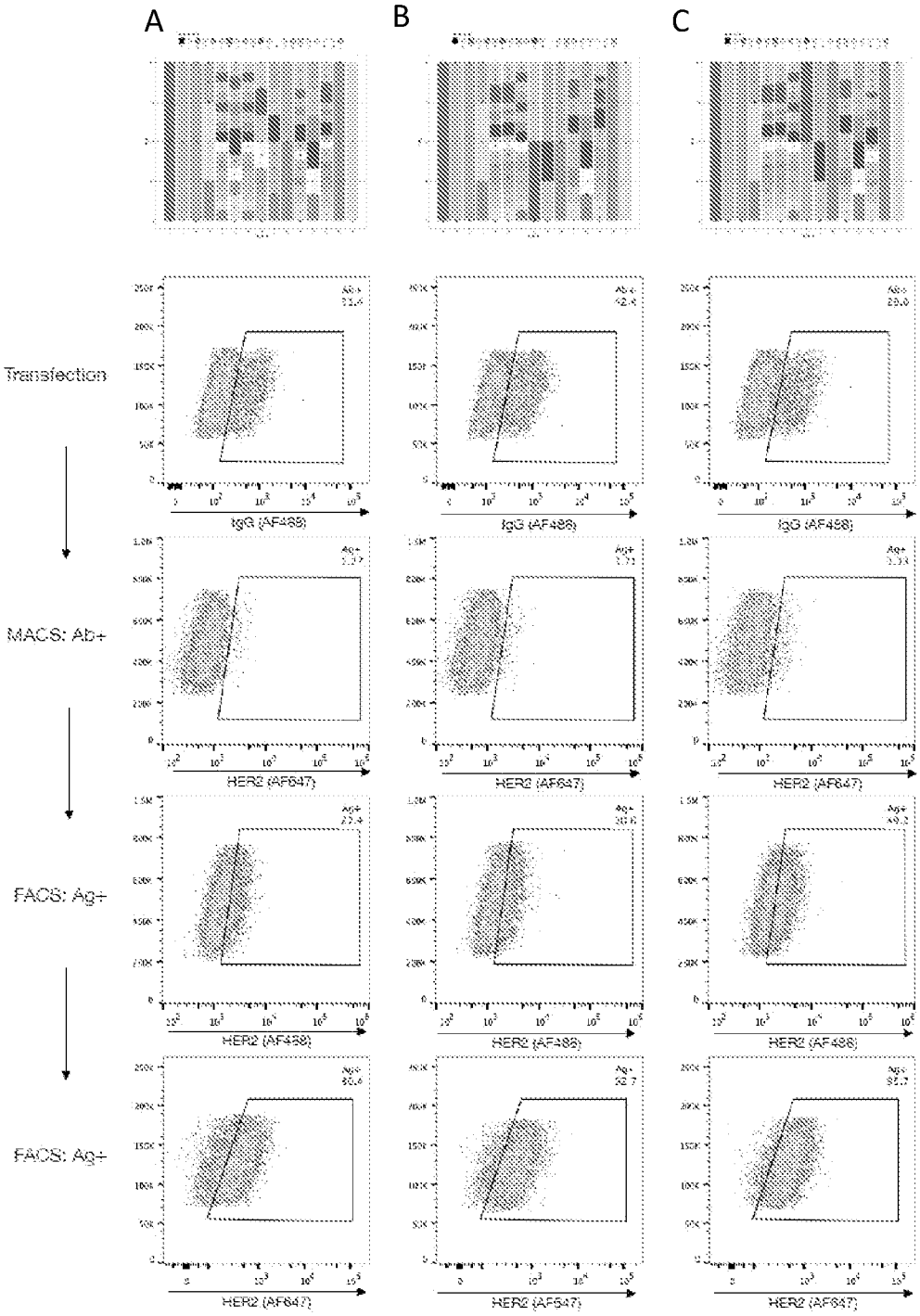

A
B
C
Transfection
MACS: Ab+
FACS: Ag+
FACS: Ag+
IgG (AF488)
HER2 (AF647)
HER2 (AF488)
HER2 (AF647)

FIG. 11

Sample	Description	Raw Read count (post-merge)	Aligned Reads	Unique clonals
DMS-H3-Ab	Library of antibody expressing hybridomas following transfection with NNK-tiled mutagenesis ssODN donors	1,060,062	992,772 (93.66%)	2,788
DMS-H3-Ag1	Library of antigen specific antibodies expressed in hybridomas following transfection with NNK-tiled mutagenesis ssODN donors	2,122,279	1,982,104 (93.4%)	1,648
DMS-H3-Ag2	Library of antigen specific antibodies in hybridomas following two rounds of enrichment for antigen binding variants	1,604,880	1,505,189 (93.79%)	790
DMS-H3-Ag3	Library of antigen specific antibodies in hybridomas following three rounds of enrichment for antigen binding variants	1,490,213	1,407,744 (94.09)	663

FIG. 12

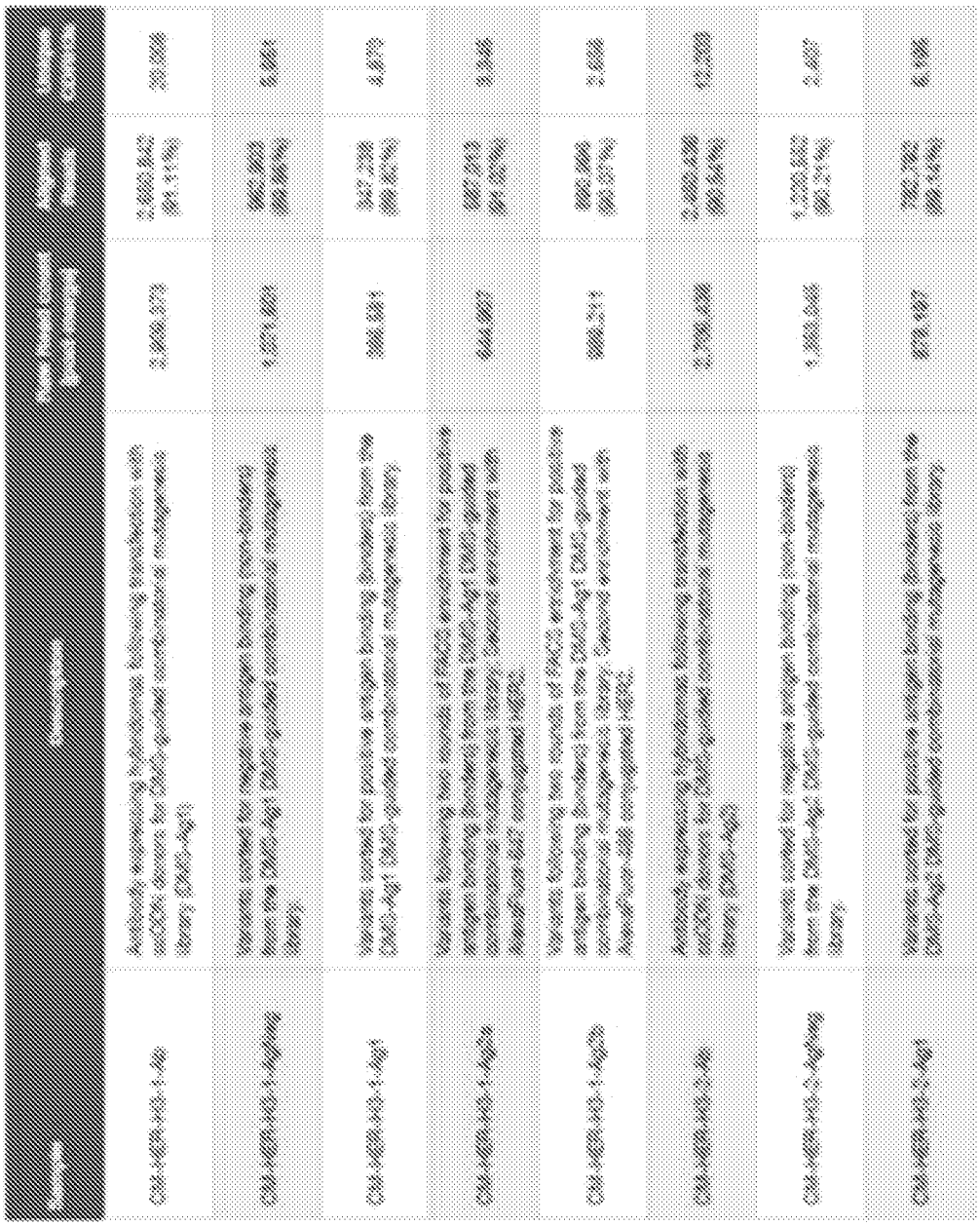


FIG. 13A

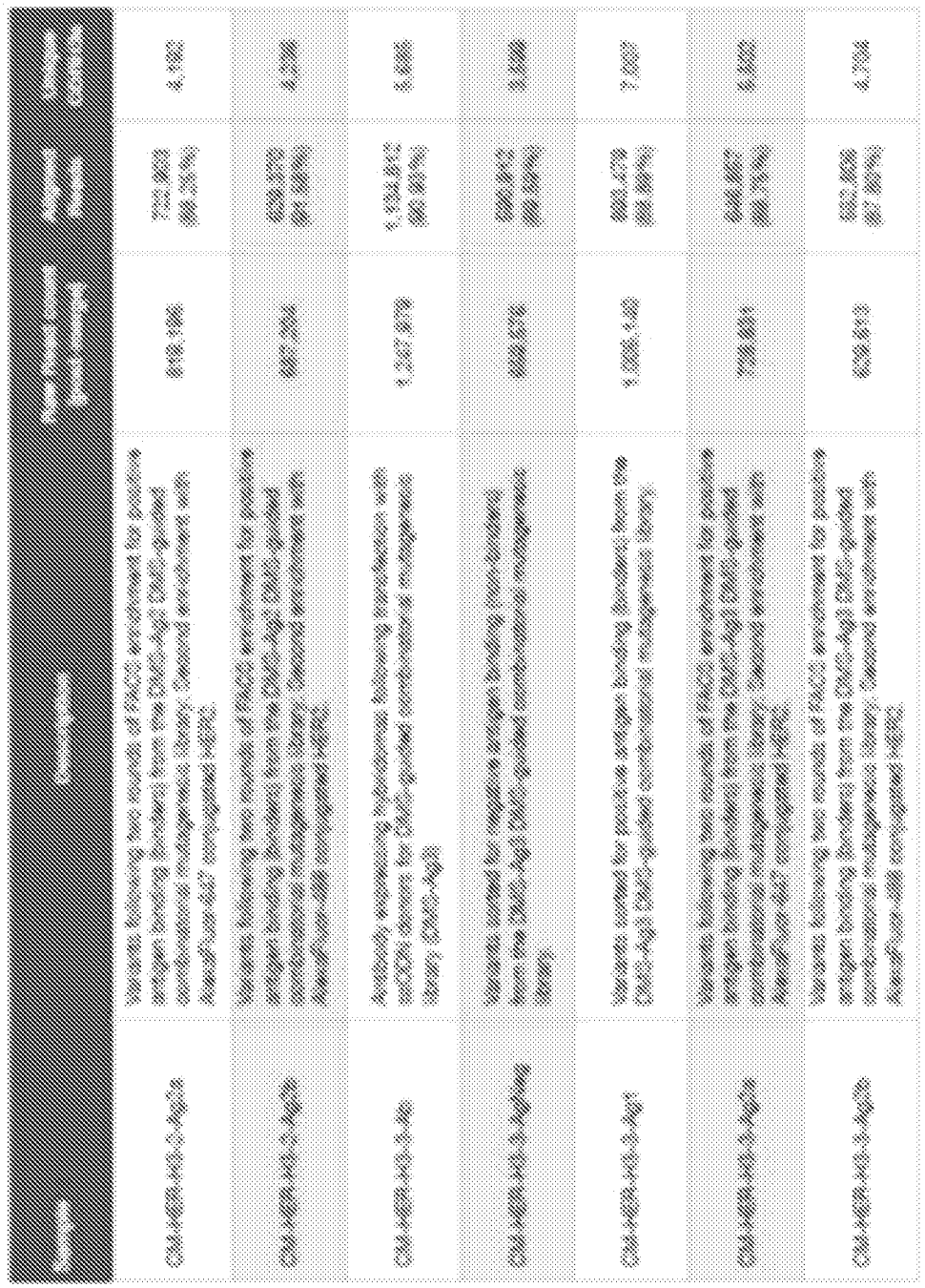


FIG. 13B

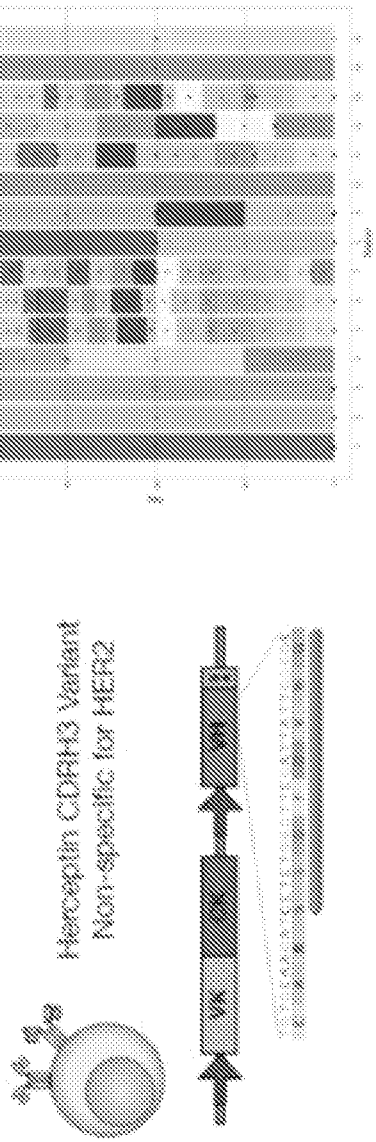


FIG. 14A

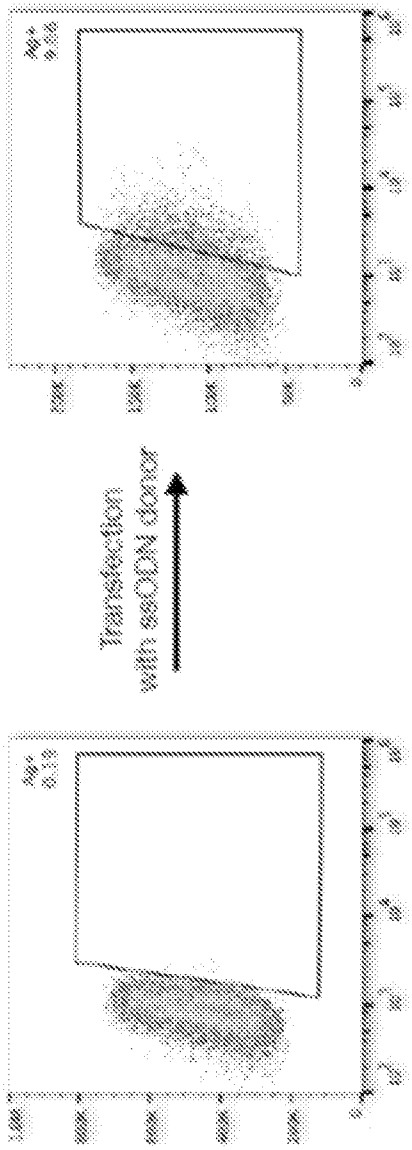


FIG. 14B

Variant ID	CDRH3 Sequence	FcγR1 F(ab')2	CHO FcγR1 F(ab')2	K_D (nM)	k_a (1/Ms)	k_d (1/s)
A	CSRWSEPGFYTNDYW	0.9118	0.9696	0.093	1.46E+05	1.21E-05
B	CSRWMETGFYTNDYW	0.8946	0.9656	0.137	1.06E+05	1.45E-05
C	CSRWKGDGFYEHDYW	0.8727	0.8698	0.335	1.38E+05	4.61E-05
D	CSRWPAPGFYTNDYW	0.9439	0.9101	0.368	2.30E+05	8.36E-05
E	CSRWPGPGMYTNDYW	0.9396	0.7702	0.381	2.87E+05	1.02E-04
F	CSRWPGPGMYANDYW	0.9530	0.9526	0.389	2.40E+05	9.33E-05
G	CSRWRGDGFYEHDYW	0.8887	0.9487	0.491	1.88E+05	9.28E-05
H	CSRWDRGGFYVNDYW	0.9561	0.9628	0.508	2.51E+05	1.28E-04
I	CSRWGHNSFYEHDYW	0.8444	0.8694	0.538	1.79E+05	9.63E-05
J	CSRWREPGFYEHDYW	0.8620	0.9809	0.638	1.49E+05	9.53E-05
K	CSRWRDRGFYEYDYW	0.8581	0.9739	0.711	3.63E+05	1.67E-04
L	CSRWGQAGFYEHDYW	0.9303	0.9823	0.737	2.01E+05	1.48E-04
M	CSRWWEPGFYEHDYW	0.8869	0.9834	0.772	9.59E+04	7.41E-05
N	CSRWWDPGFYEHDYW	0.8973	0.9053	0.774	1.16E+05	8.96E-05
O	CSRWGACGFYENDYW	0.9189	0.9526	0.903	1.69E+05	1.52E-04
P	CSRYRAVGFYTNDYW	0.7828	0.8707	0.909	2.66E+05	2.41E-04
Q	CSRYQPPGMYTNDYW	0.9085	0.8373	0.963	2.94E+05	2.83E-04
R	CSRWRDDGLYAHDYW	0.8857	0.9383	1.011	1.83E+05	1.85E-04
S	CSRWNDRGFYEHDYW	0.7857	0.9184	1.067	9.01E+04	9.62E-05
T	CSRWNPHGLYVNDYW	0.9385	0.8741	1.111	2.41E+05	2.67E-04
U	CSRWPGQGMYANDYW	0.9016	0.9544	1.148	3.72E+05	3.12E-04
V	CSRYQEYGMYVNKYW	0.8876	0.8696	1.259	3.23E+05	4.01E-04
W	CSRWAPYGLYAHDYW	0.9483	0.9590	1.275	2.92E+05	3.72E-04
X	CSRWGGPGMYTNDYW	0.9527	0.9103	1.321	3.07E+05	4.05E-04
Y	CSRWRDPGFYEHDYW	0.8874	0.8804	1.367	2.00E+05	2.73E-04
Z	CSRWRDPGFYEHDYW	0.8874	0.8804	1.402	1.69E+05	2.37E-04

FIG. 15A

Variant ID	CDRH3 Sequence	RNN P(Binder)	CNN P(Binder)	K_D (nM)	ka (1/Ms)	kd (1/s)
AA	CSRWGPSGLYVNDYW	0.9635	0.9322	1.424	2.73E+05	3.89E-04
AB	CSRWKQDGFYEHDYW	0.8921	0.7652	1.436	5.41E+04	7.79E-05
AC	CSRYGMRGLYAYEYW	0.9423	0.9480	1.470	3.10E+05	4.55E-04
AD	CSRYGGPGMYTNDYW	0.9584	0.9665	1.571	3.06E+05	4.80E-04
AE	CSRYGGRGMYQLEYW	0.9062	0.9729	1.735	2.90E+05	5.02E-04
AF	CSRYGGDGMYQHDYW	0.9307	0.9345	1.772	2.19E+05	3.88E-04
AG	CSRYGGTGMYEHDYW	0.9488	0.8175	1.833	2.38E+05	4.36E-04
AH	CSRWGSNGMYVNDYW	0.9074	0.8267	1.857	3.09E+05	5.74E-04
AI	CSRWEEYGLYVHDYW	0.9325	0.9920	1.979	7.78E+04	1.54E-04
AJ	CSRYGNGGMYANDYW	0.9536	0.8200	2.233	5.54E+05	1.24E-03
AK	CSRWAMLGMYAHDYW	0.9474	0.7788	2.315	3.01E+05	6.96E-04
AL	CSRWPKVGLYTNDYW	0.8529	0.8850	2.459	2.57E+05	6.32E-04
AM	CSRYGPGGMYEHDYW	0.9411	0.7793	2.474	2.87E+05	7.10E-04
AN	CSRWGGPGMYEHDYW	0.9096	0.8570	2.526	2.01E+05	5.07E-04
AO	CSRYRGDGFYEHDYW	0.7909	0.8612	2.849	2.49E+05	7.09E-04
AP	CSRYGGPGLYANDYW	0.9686	0.9846	2.881	2.90E+05	8.35E-04
AQ	CSRYGGPGMYQNDYW	0.9419	0.9819	3.021	2.57E+05	7.76E-04
AR	CSRWPQPGMYANDYW	0.8478	0.9096	3.024	2.64E+05	7.98E-04
AS	CSRYGGPGMYEHDYW	0.9478	0.9148	3.121	1.88E+05	5.87E-04
AT	CSRWGGIGLYELDYW	0.9416	0.8890	3.202	1.95E+05	6.24E-04
AU	CSRWGEAAFYAHDYW	0.8329	0.9299	3.306	2.22E+05	7.34E-04
AV	CSRYGDPGMYQHDYW	0.8319	0.9502	3.308	1.99E+05	6.57E-04
AW	CSRYGGPGLYTNDYW	0.9523	0.7923	3.319	3.12E+05	1.04E-03
AX	CSRWSDSGMYAHDYW	0.6239	0.5839	3.334	2.81E+05	9.36E-04
AY	CSRWNGLGMYVNDYW	0.8740	0.8884	3.355	2.23E+05	7.47E-04
AZ	CSRYAGPGMYTNDYW	0.8702	0.8297	3.371	2.43E+05	8.20E-04

FIG. 15B

Variant ID	CDRH3 Sequence	ZDHX Polyreact	CDRH Polyreact	K_D (nM)	k_a (1/Ms)	k_d (1/s)
BA	CSRYGPQGLYTNDYW	0.9542	0.8692	3.449	3.39E+05	1.17E-03
BB	CSRYGGPGLYENDYW	0.9410	0.8709	3.512	1.90E+05	6.67E-04
BC	CSRWAEAGMYEFDYW	0.8803	0.8150	3.600	1.25E+05	4.52E-04
BD	CSRYSMPGMYTNAYW	0.9480	0.9077	3.601	2.65E+05	9.52E-04
BE	CSRWGNPGMYANDYW	0.9287	0.9277	3.652	3.44E+05	1.26E-03
BF	CSRYRGDGFYENDYW	0.7909	0.8512	3.679	2.45E+05	9.02E-04
BG	CSRVRHDGFYENDYW	0.9468	0.9784	3.705	9.77E+04	3.62E-04
BH	CSRWGSPGLYTFDYW	0.9583	0.9639	3.777	2.79E+05	1.05E-03
BI	CSRYGPGGMYANDYW	0.9601	0.9842	3.854	3.50E+05	1.35E-03
BJ	CSRVPMDGMYTHDYW	0.8347	0.8548	3.855	1.14E+05	4.38E-04
BK	CSRWGNVSFYENDYW	0.8708	0.8728	3.900	2.04E+05	7.97E-04
BL	CSRWGNGGMYANDYW	0.9477	0.9022	3.968	5.81E+05	2.29E-03
BM	CSRYGERGFYENDYW	0.7983	0.9256	4.020	2.48E+05	9.97E-04
BN	CSRYGQNGMYTNDYW	0.9493	0.8075	4.023	3.51E+05	1.41E-03
BO	CSRYGSPGMYTNDYW	0.9316	0.8950	4.397	3.49E+05	1.54E-03
BP	CSRYGGPGMYTNDYW	0.9512	0.9525	4.647	3.80E+05	1.77E-03
BQ	CSRYGEPGMYQHDYW	0.7950	0.9549	4.789	2.43E+05	1.16E-03
BR	CSRYGDAGMYALKYW	0.9734	0.9950	4.844	3.72E+05	1.80E-03
BS	CSRWQGNGMYANDYW	0.9469	0.8551	4.869	4.01E+05	1.95E-03
BT	CSRYASAGMYTHDYW	0.9273	0.8825	4.922	3.36E+05	1.65E-03
BU	CSRYGPTGMYQNDYW	0.9211	0.9304	4.941	3.18E+05	1.57E-03
BV	CSRYGDRGFYENDYW	0.8380	0.9528	5.350	3.11E+05	1.66E-03
BW	CSRYGTPGMYQNDYW	0.9692	0.8047	5.409	3.16E+05	1.71E-03
BX	CSRYGPNGLYANDYW	0.9504	0.9442	5.471	3.63E+05	1.99E-03
BY	CSRYGGNGMYQNDYW	0.9304	0.8686	5.485	3.17E+05	1.74E-03
BZ	CSRYGNPGMYQNDYW	0.8763	0.9261	5.858	2.83E+05	1.66E-03

FIG. 15C

Variant ID	CDRH3 Sequence	RBD1 P(Bound)	CHO P(Bound)	KD (nM)	ka (1/Ms)	kd (1/s)
CA	CSRYQSQGLYEHDYW	0.9150	0.8585	9.27	5.19E+05	3.26E-03
CB	CSRYQARGFYQNDYW	0.8922	0.7607	6.33	4.36E+05	2.77E-03
CC	CSRWSESGPYTHDYW	0.7969	0.6666	9.51	3.99E+04	2.60E-04
CD	CSRYQGPGMYANDYW	0.9067	0.9609	9.65	3.82E+05	2.54E-03
CE	CSRYSGPGMYANDYW	0.9601	0.9370	8.86	4.23E+05	2.90E-03
CF	CSRYGGNGMYQNDYW	0.8304	0.8686	7.14	2.78E+05	1.99E-03
CG	CSRWNSDGLYAYEYW	0.8640	0.9615	7.22	3.31E+05	2.39E-03
CH	CSRYGSNGLYANDYW	0.9139	0.9076	7.25	3.14E+05	2.28E-03
CI	CSRYGSNGMYQNDYW	0.8388	0.7937	7.31	4.17E+05	3.04E-03
CJ	CSRWNAPSFYANDYW	0.8847	0.9770	7.43	3.02E+05	2.25E-03
CK	CSRWQLGGMYTHDYW	0.9195	0.9409	7.55	1.65E+05	1.25E-03
CL	CSRYGGNGMYANDYW	0.9545	0.9145	7.68	4.42E+05	3.39E-03
CM	CSRWERPGLYEHDYW	0.9175	0.8697	8.66	2.73E+05	2.37E-03
CN	CSRWRPDGFYEHDYW	0.8980	0.9108	8.70	2.95E+05	2.57E-03
CO	CSRWPSPSFYTNDYW	0.8664	0.5772	9.06	3.61E+05	3.27E-03
CP	CSRWERPGLYEHDYW	0.8175	0.8697	9.12	2.94E+05	2.68E-03
CQ	CSRYQANGLYAYEYW	0.9587	0.8859	9.77	4.34E+05	4.24E-03
CR	CSRYGPNGMYQNDYW	0.9491	0.8365	10.31	3.28E+05	3.38E-03
CS	CSRWREPGLYEHDYW	0.7946	0.7996	11.40	1.37E+05	1.56E-03
CT	CSRYGNPGMYEHDYW	0.8653	0.7579	11.69	1.81E+05	2.11E-03
CU	CSRWDRPGLYEHDYW	0.7934	0.7723	11.81	6.07E+04	7.17E-04
CV	CSRWKEPGLYEHDYW	0.8261	0.8086	13.28	9.14E+04	1.21E-03
CW	CSRYQGPGMYEHDYW	0.9387	0.8688	13.32	1.34E+05	1.79E-03
CX	CSRYGGPGLYEHDYW	0.9435	0.8559	18.04	7.59E+04	1.36E-03
CY	CSRWDGPAFYELDYW	0.8598	0.9362	31.08	1.03E+05	3.21E-03
CZ	CSRWRGDGLYEHDYW	0.8142	0.8495	53.39	1.22E+04	6.39E-04

FIG. 15D

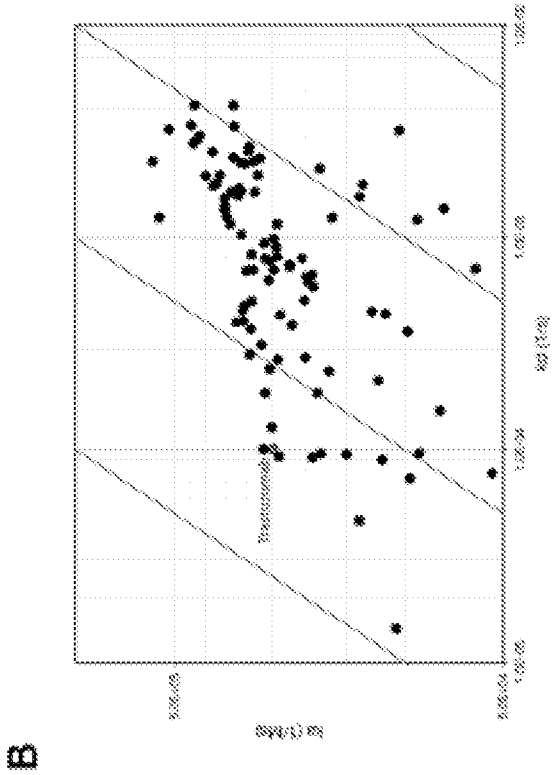


FIG. 16B

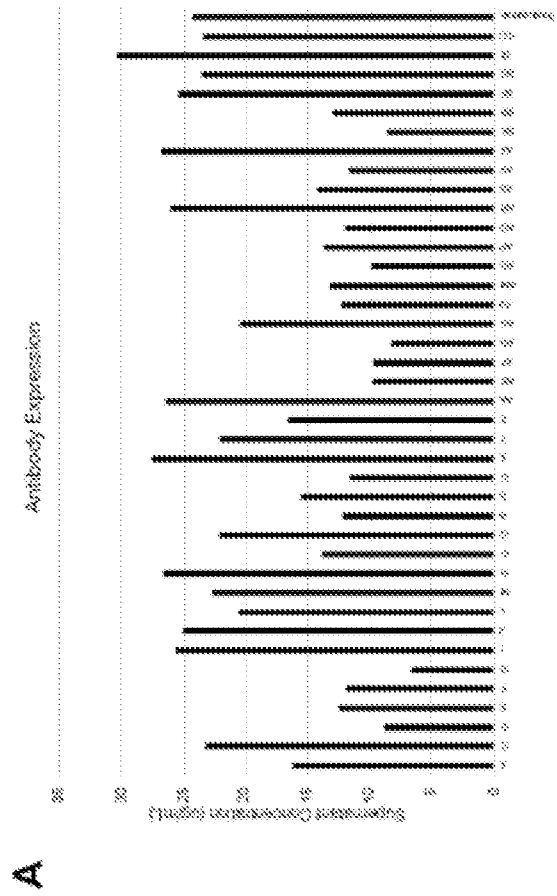


FIG. 16A

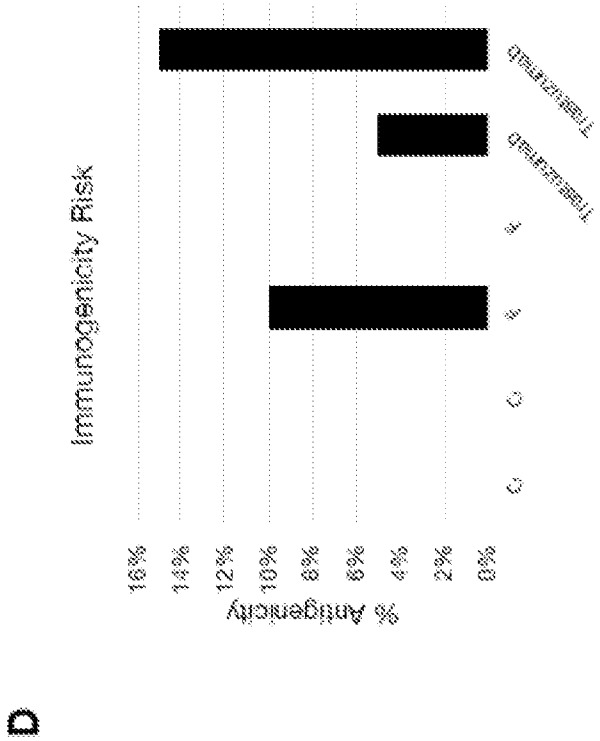


FIG. 16D

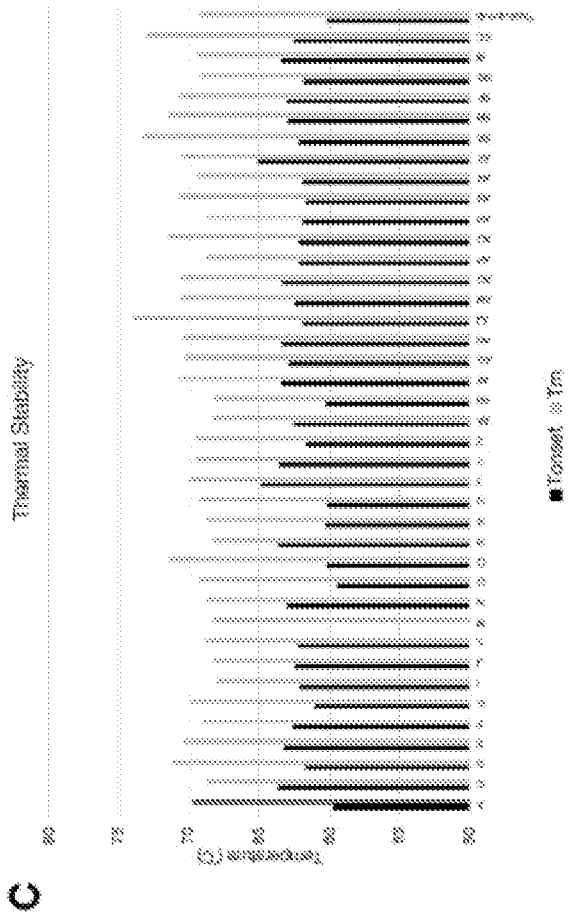


FIG. 16C

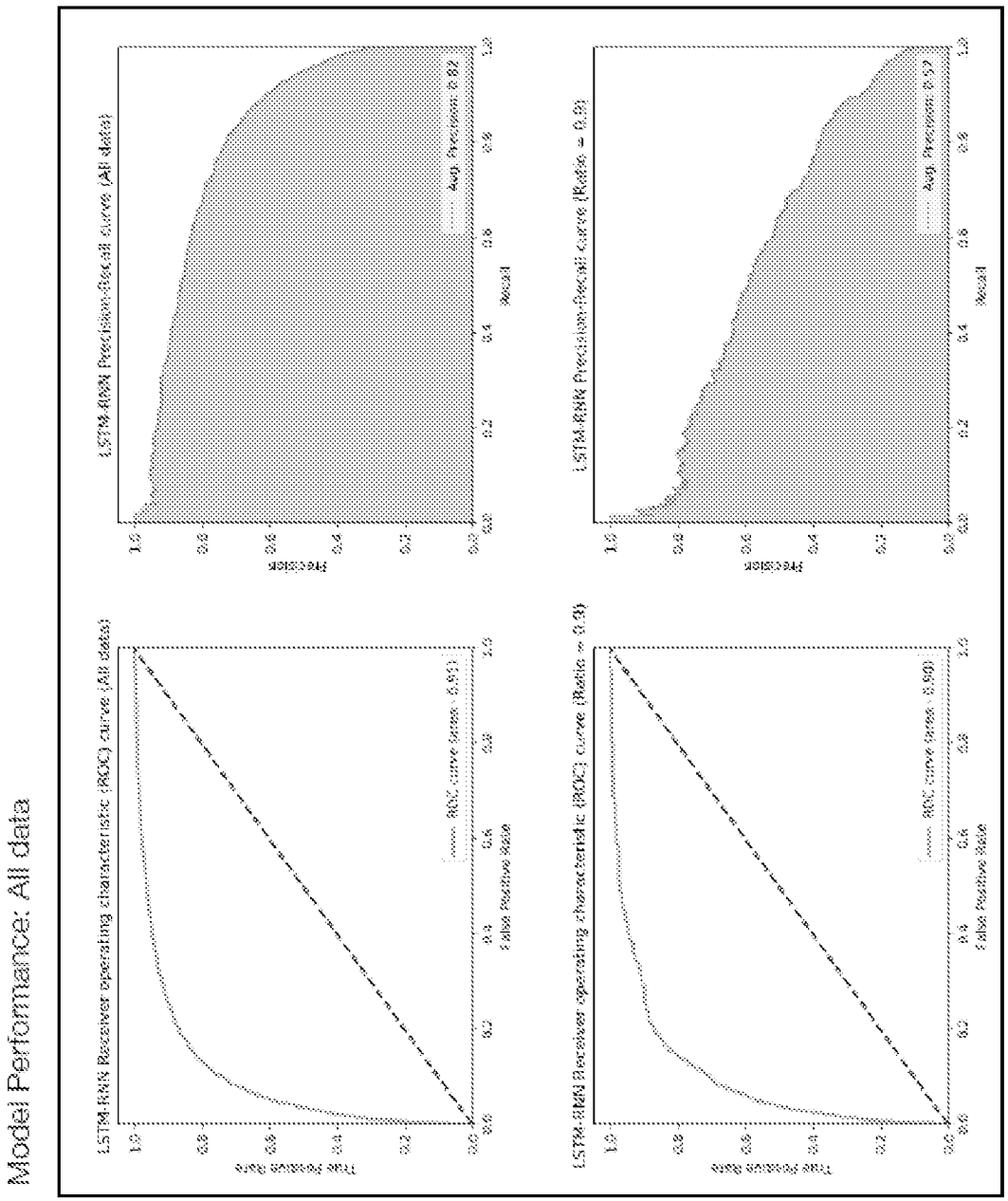

Model Performance: All data
LSTM-RNN Receiver operating characteristic (ROC) curve (All data)
True Positive Rate
False Positive Rate
ROC curve (area = 0.95)
LSTM-RNN Precision-Recall curve (All data)
Precision
Recall
Avg. Precision: 0.82
LSTM-RNN Receiver operating characteristic (ROC) curve (Ratio = 0.3)
ROC curve (area = 0.96)
LSTM-RNN Precision-Recall curve (Ratio = 0.3)
Avg. Precision: 0.93

FIG. 17A

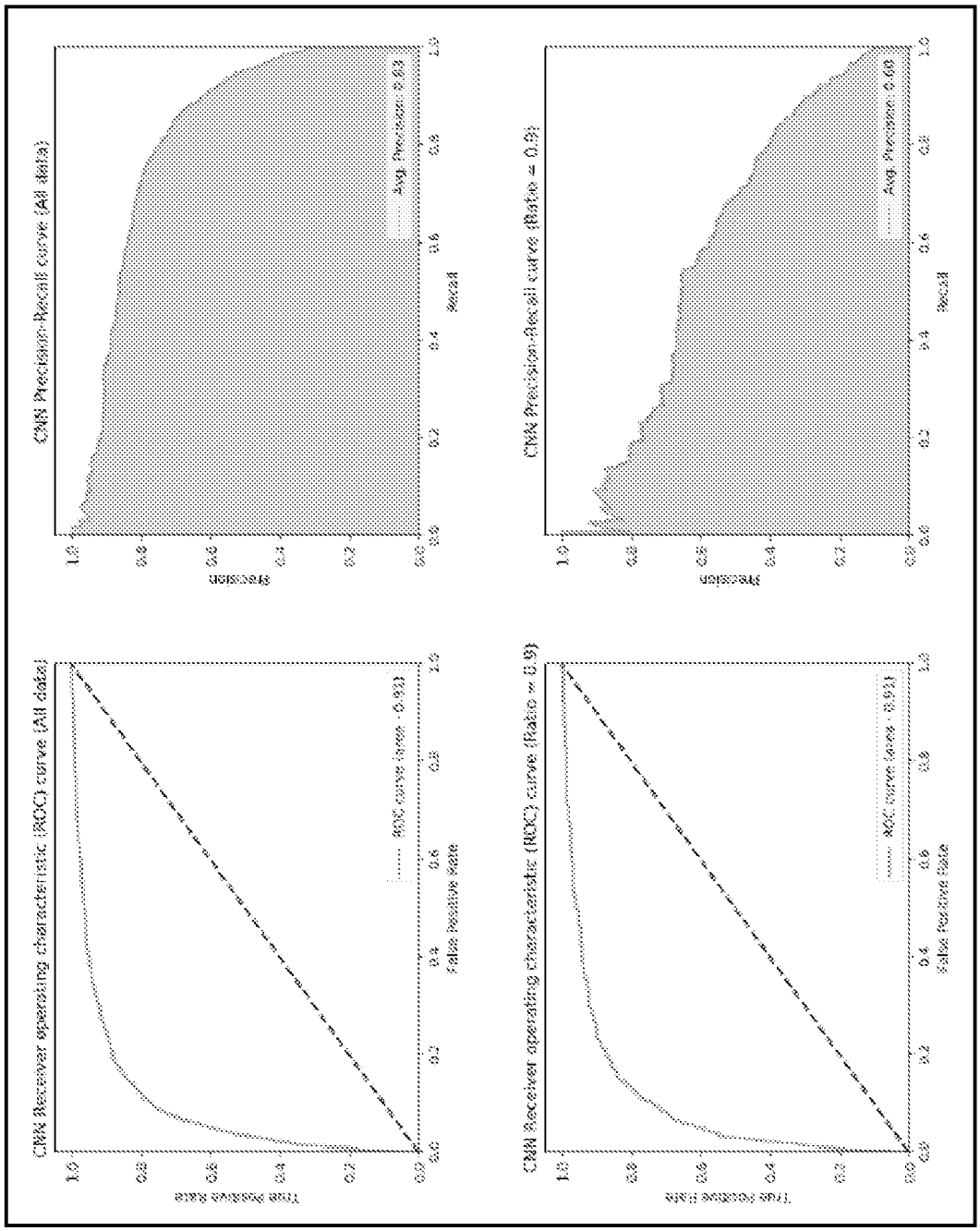

CNN Receiver operating characteristic (ROC) curve (All data)
True Positive Rate
False Positive Rate
ROC curve (area = 0.93)
CNN Precision-Recall curve (All data)
Precision
Recall
Avg Precision: 0.83
CNN Receiver operating characteristic (ROC) curve (Ratio = 0.5)
True Positive Rate
False Positive Rate
ROC curve (area = 0.93)
CNN Precision-Recall curve (Ratio = 0.5)
Precision
Recall
Avg Precision: 0.68

FIG. 17B

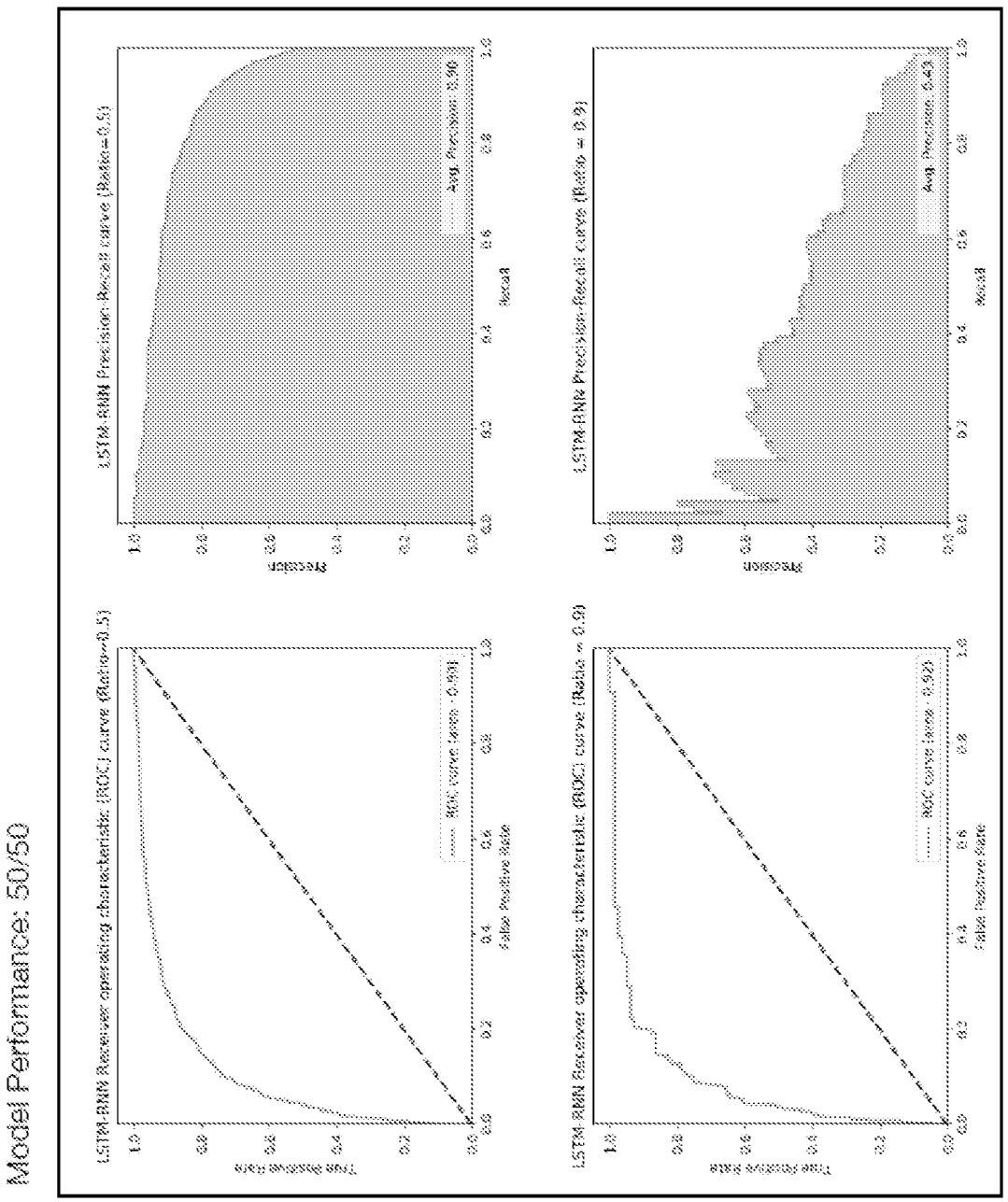

Model Performance: 50/50
LSTM-RNN Receiver operating characteristic (ROC) curve (Ratio=0.5)
True Positive Rate
False Positive Rate
ROC curve (area = 0.93)
LSTM-RNN Precision-Recall curve (Ratio=0.5)
Precision
Recall
Avg Precision: 0.90
LSTM-RNN Receiver operating characteristic (ROC) curve (Ratio = 0.9)
True Positive Rate
False Positive Rate
ROC curve (area = 0.92)
LSTM-RNN Precision-Recall curve (Ratio = 0.9)
Precision
Recall
Avg Precision: 0.43

FIG. 18A

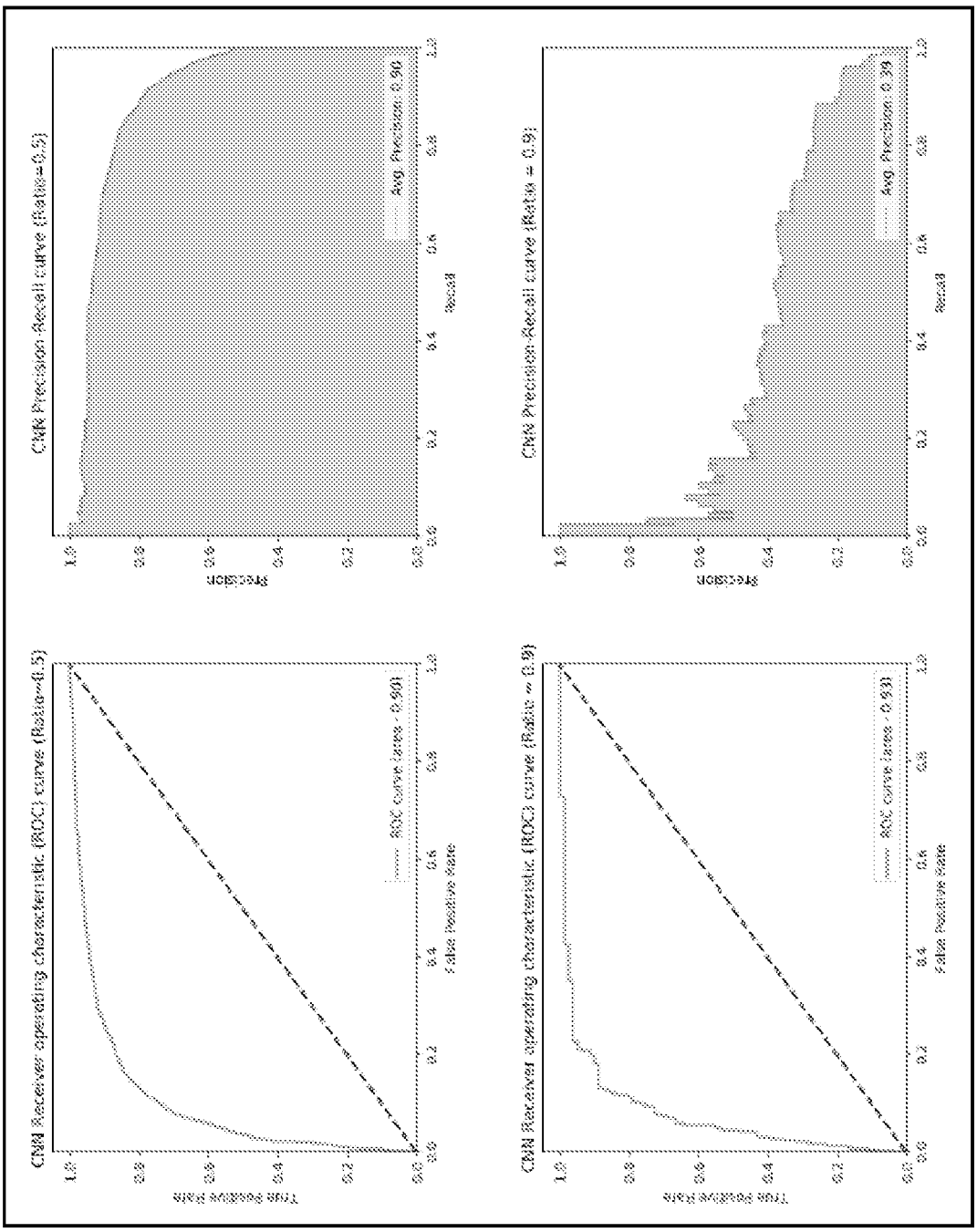

CNN Receiver operating characteristic (ROC) curve (Ratio=0.5)
True Positive Rate
False Positive Rate
ROC curve (area = 0.90)
CNN Precision-Recall curve (Ratio=0.5)
Precision
Recall
Avg. Precision: 0.90
CNN Receiver operating characteristic (ROC) curve (Ratio = 0.9)
True Positive Rate
False Positive Rate
ROC curve (area = 0.93)
CNN Precision-Recall curve (Ratio = 0.9)
Precision
Recall
Avg. Precision: 0.89

FIG. 18B

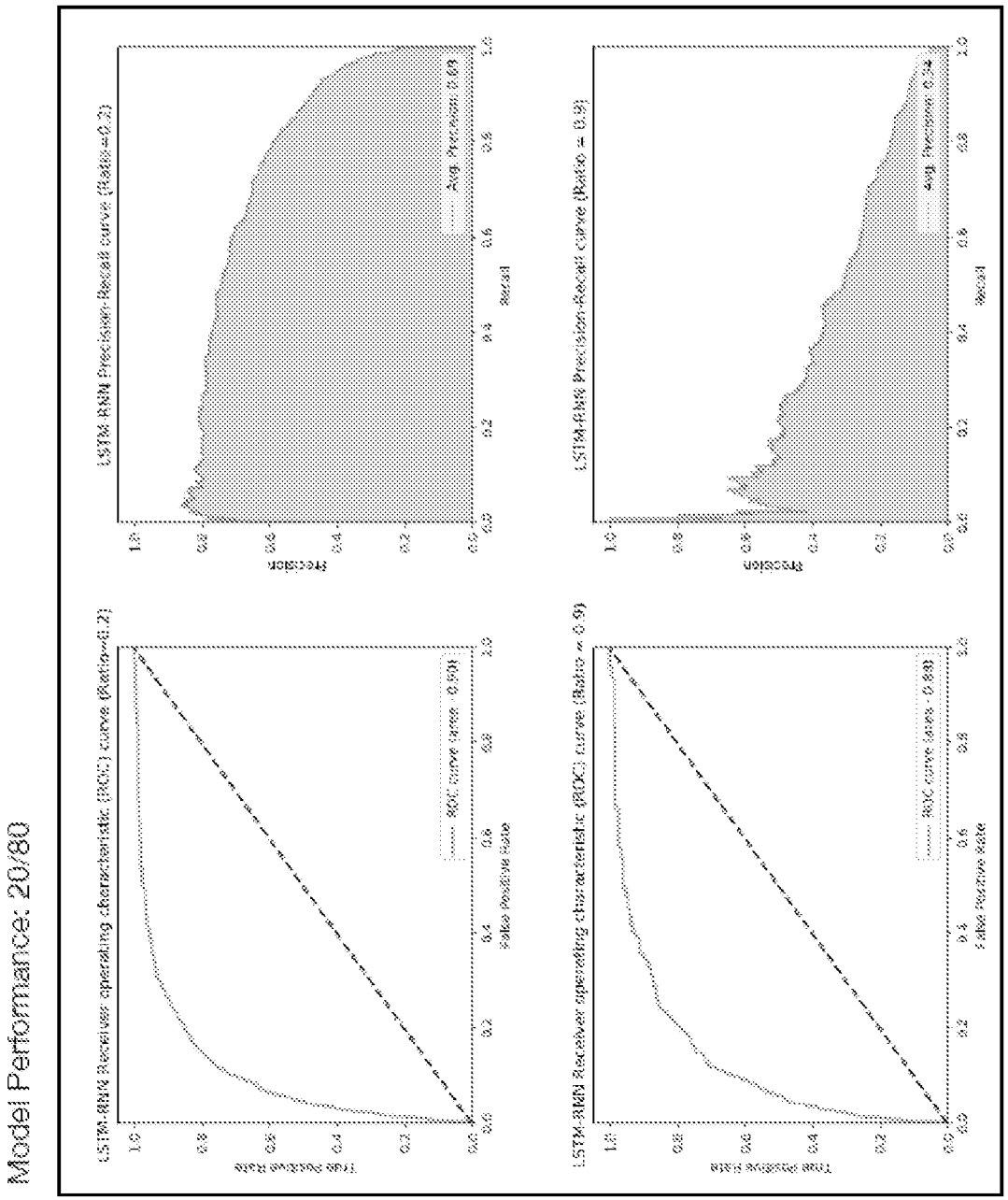

Model Performance: 20/80
LSTM-RNN Receiver operating characteristic (ROC) curve (Ratio=0.2)
True Positive Rate
False Positive Rate
ROC curve (area = 0.90)
LSTM-RNN Precision-Recall curve (Ratio=0.2)
Precision
Recall
Avg Precision: 0.69
LSTM-RNN Receiver operating characteristic (ROC) curve (Ratio = 0.5)
True Positive Rate
False Positive Rate
ROC curve (area = 0.88)
LSTM-RNN Precision-Recall curve (Ratio = 0.5)
Precision
Recall
Avg Precision: 0.74

FIG. 19A

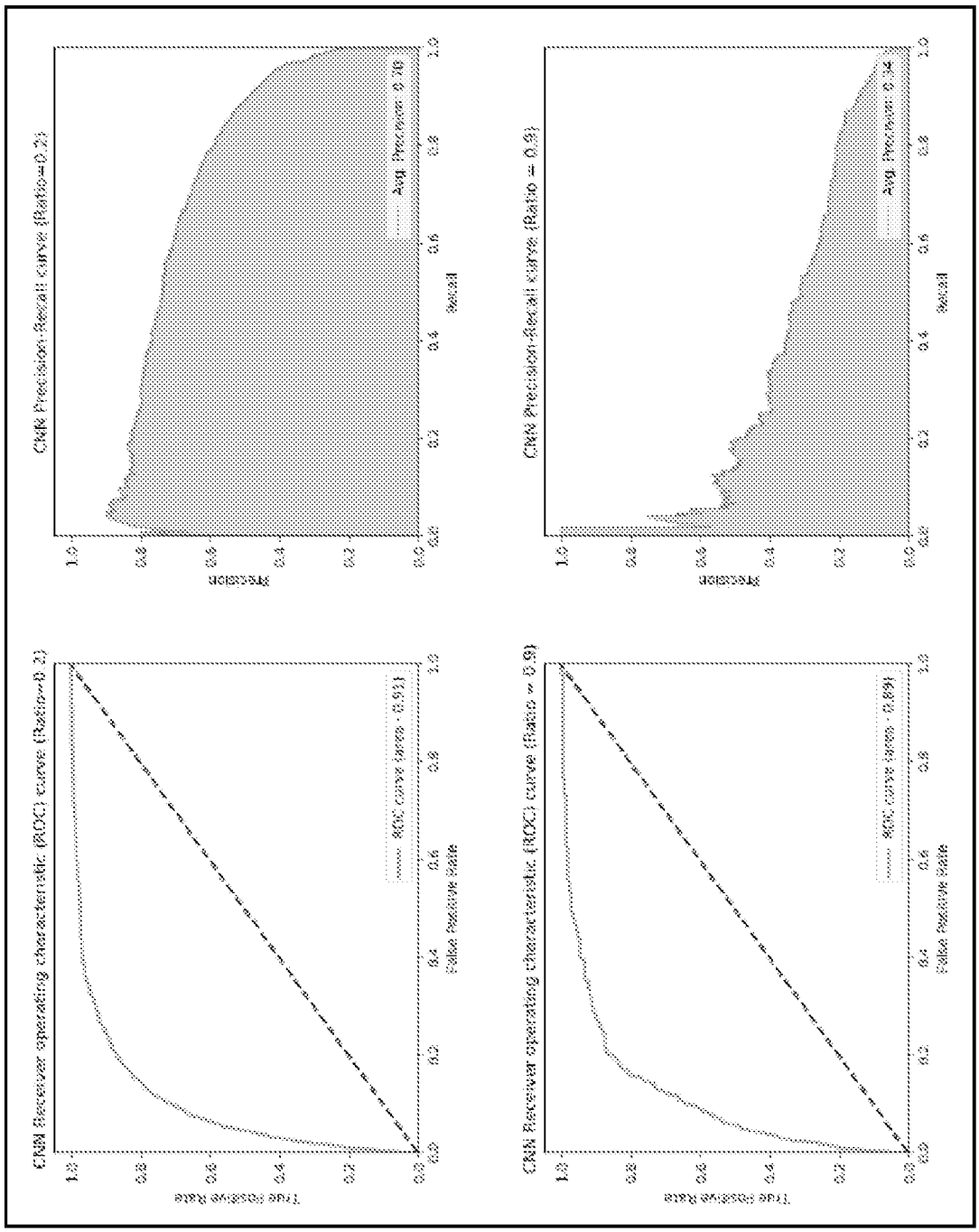

CNN Precision-Recall curve (Ratio=0.2)
Precision
Recall
Avg. Precision: 0.98
CNN Precision-Recall curve (Ratio = 0.9)
Precision
Recall
Avg. Precision: 0.94
CNN Receiver operating characteristic (ROC) curve (Ratio=0.2)
True Positive Rate
False Positive Rate
ROC curve (area = 0.91)
CNN Receiver operating characteristic (ROC) curve (Ratio = 0.9)
True Positive Rate
False Positive Rate
ROC curve (area = 0.89)

FIG. 19B

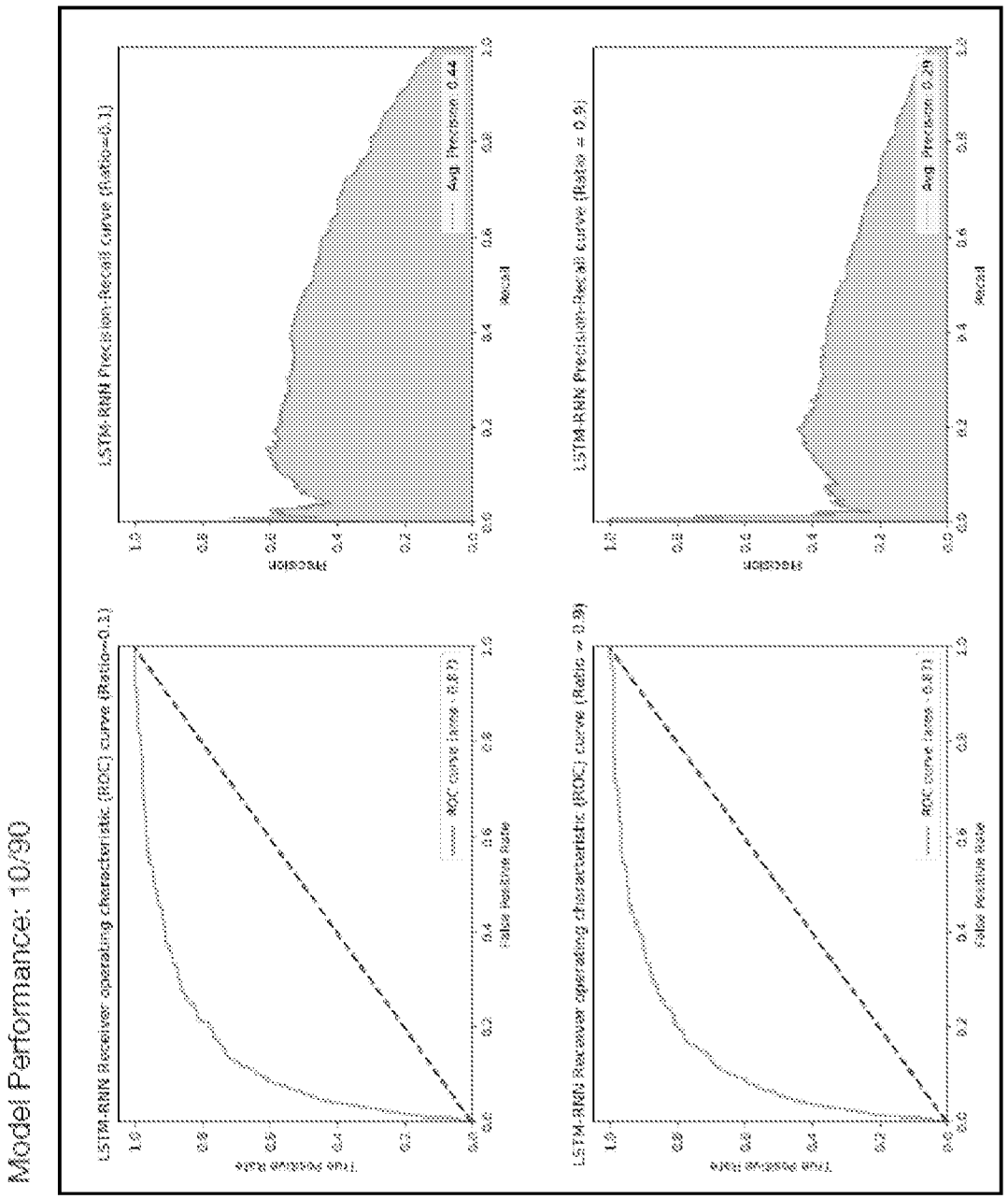

Model Performance: 10/30
LSTM-RNN Receiver operating characteristic (ROC) curve (Ratio=0.1)
True Positive Rate
False Positive Rate
ROC curve (area = 0.83)
LSTM-RNN Precision-Recall curve (Ratio=0.1)
Precision
Recall
Avg Precision: 0.44
LSTM-RNN Receiver operating characteristic (ROC) curve (Ratio = 0.9)
True Positive Rate
False Positive Rate
ROC curve (area = 0.83)
LSTM-RNN Precision-Recall curve (Ratio = 0.9)
Precision
Recall
Avg Precision: 0.29

FIG. 20A

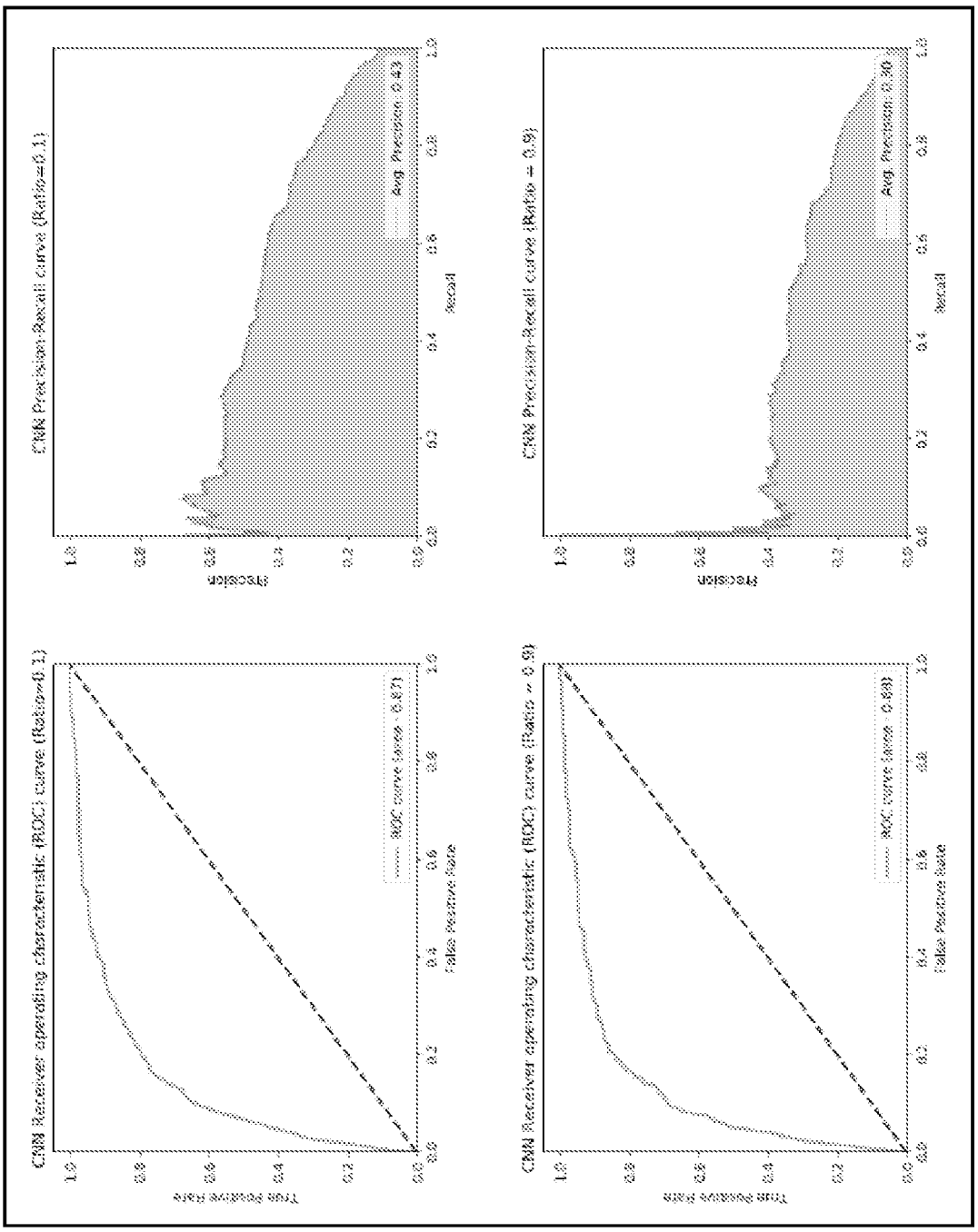

CNN Precision-Recall curve (Ratio=0.1)
Avg. Precision: 0.43
Precision
Recall
CNN Precision-Recall curve (Ratio = 0.9)
Avg. Precision: 0.30
Precision
Recall
CNN Receiver operating characteristic (ROC) curve (Ratio=0.1)
ROC curve (area = 0.87)
True Positive Rate
False Positive Rate
CNN Receiver operating characteristic (ROC) curve (Ratio = 0.9)
ROC curve (area = 0.68)
True Positive Rate
False Positive Rate

FIG. 20B

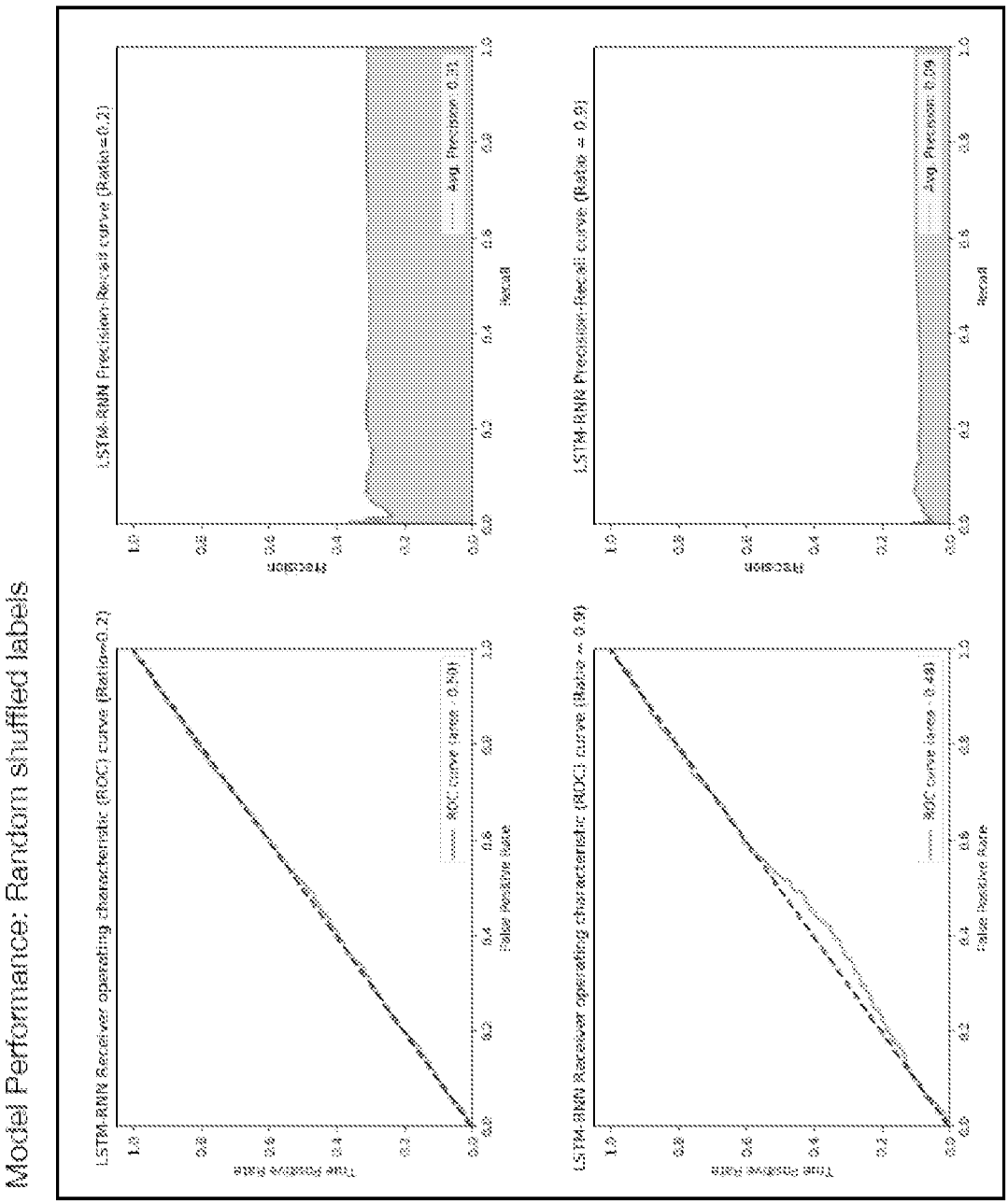

Model Performance: Random shuffled labels
LSTM-RNN Receiver operating characteristic (ROC) curve (Ratio=0.2)
True Positive Rate
False Positive Rate
ROC curve (area = 0.50)
LSTM-RNN Precision-Recall curve (Ratio=0.2)
Precision
Recall
Avg Precision: 0.33
LSTM-RNN Receiver operating characteristic (ROC) curve (Ratio = 0.9)
True Positive Rate
False Positive Rate
ROC curve (area = 0.49)
LSTM-RNN Precision-Recall curve (Ratio = 0.9)
Precision
Recall
Avg Precision: 0.89

FIG. 21A

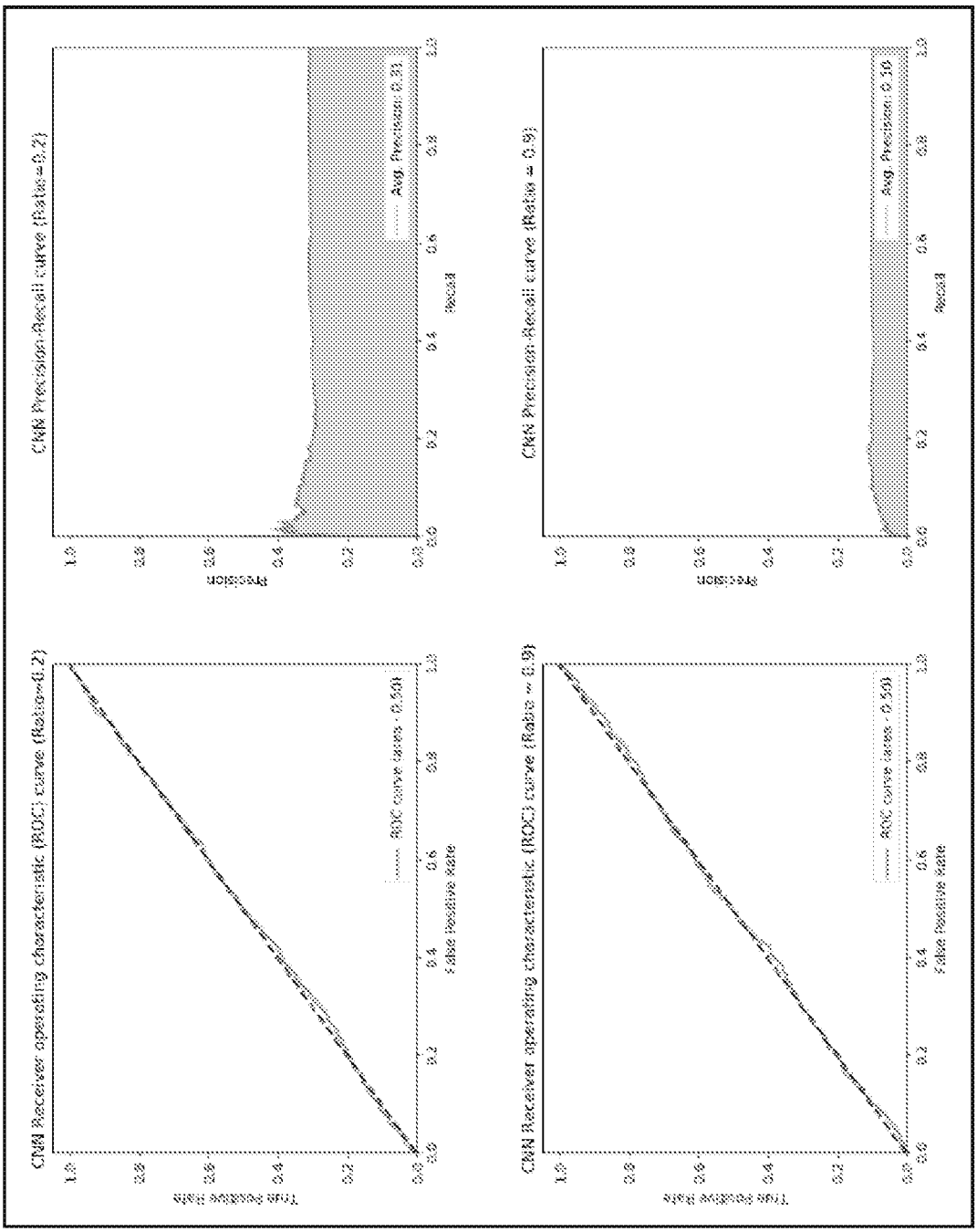

CNN Precision-Recall curve (Ratio=0.2)
Avg. Precision: 0.31
Precision
Recall
CNN Precision-Recall curve (Ratio = 0.9)
Avg. Precision: 0.50
Precision
Recall
CNN Receiver operating characteristic (ROC) curve (Ratio=0.2)
ROC curve (area = 0.50)
True positive rate
False positive rate
CNN Receiver operating characteristic (ROC) curve (Ratio = 0.9)
ROC curve (area = 0.50)
True positive rate
False positive rate

FIG. 21B

Model / Class split	ROC curve AUC	Average PR
RNN All data	0.91	0.82
CNN All data	0.91	0.83
RNN 50/50	0.90	0.90
CNN 50/50	0.90	0.90
RNN 20/80	0.90	0.69
CNN 20/80	0.91	0.70
RNN 10/90	0.87	0.44
CNN 10/90	0.87	0.43

Model / Class split	Predicted Binders
RNN 50/50	26,480
CNN 50/50	24,156
RNN All data	17,489
CNN All data	17,920
RNN 20/80	13,893
CNN 20/80	11,336
RNN 10/90	4,296
CNN 10/90	8,041

FIG. 22

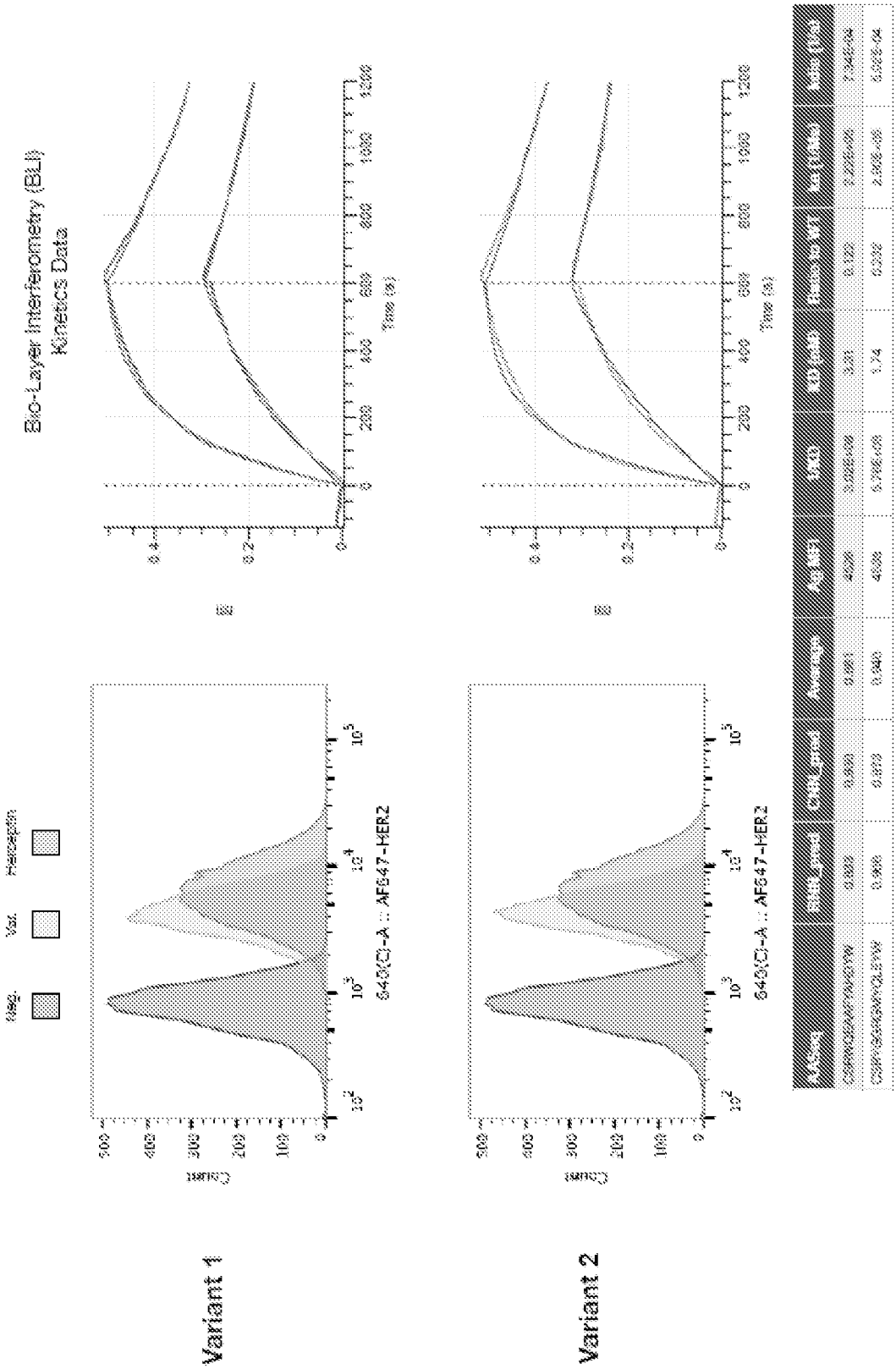

Bio-Layer Interferometry (BLI)
Kinetics Data
Time (s)
Variant 1
Variant 2
Count
640(C)-A :: AF647-HER2

FIG. 23A

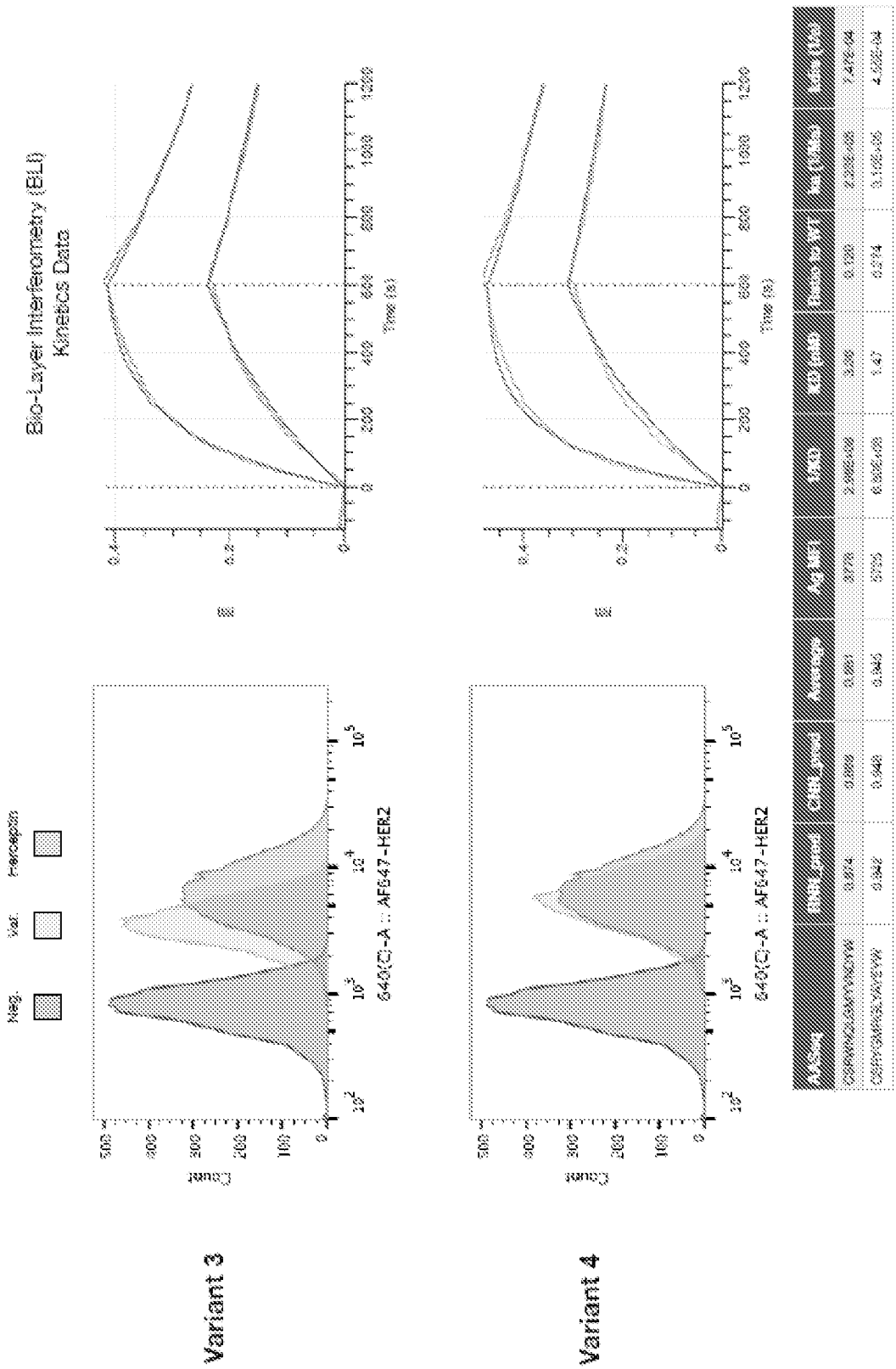

Variant 3
Variant 4
Neg.
Var.
Herceptin
Count
640(C)-A :: AF647-HER2
Bio-Layer Interferometry (BLI)
Kinetics Data
nm
Time (s)

FIG. 23B

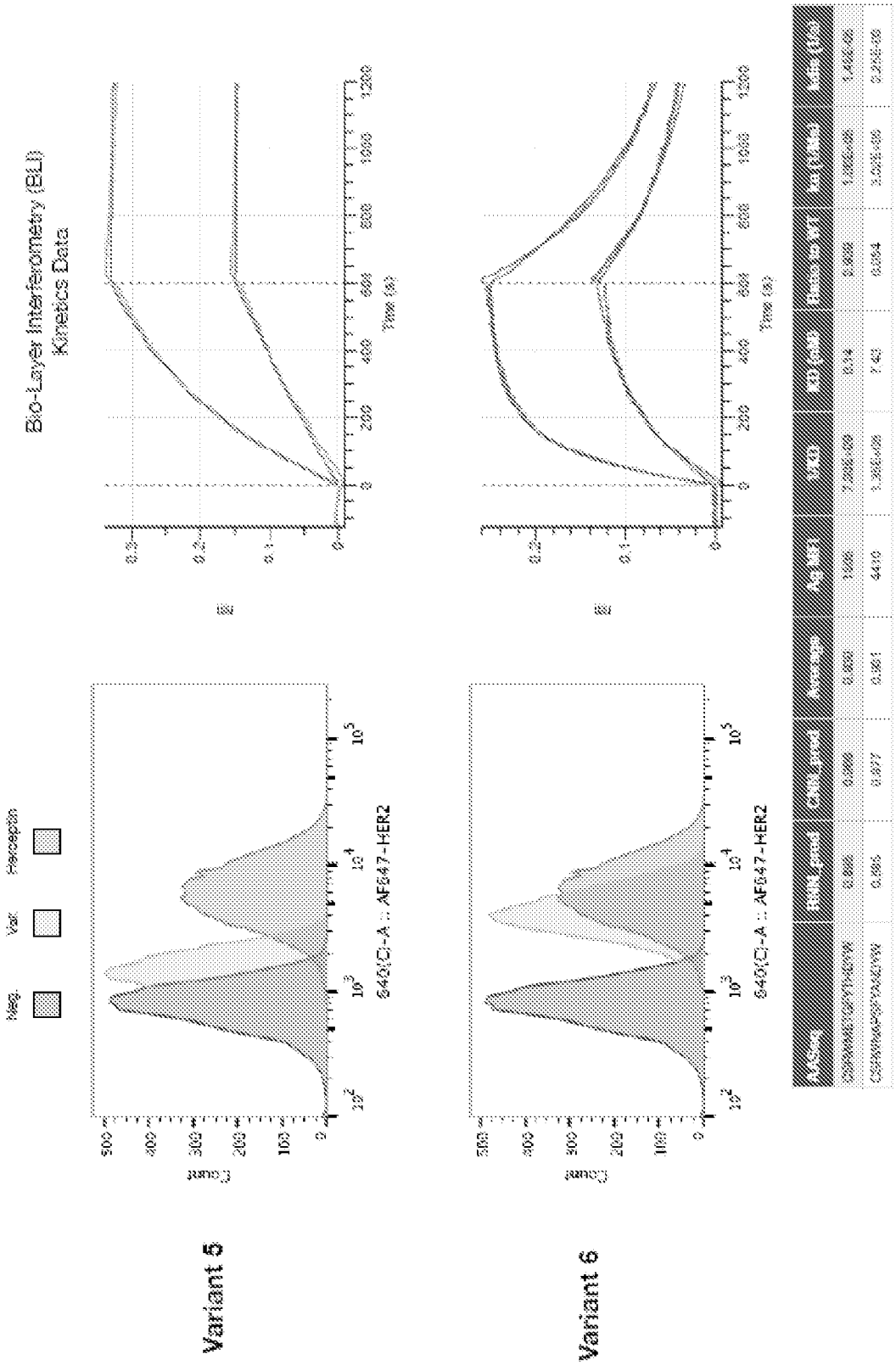

Bio-Layer Interferometry (BLI)
Kinetics Data
Time (s)
Variant 5
Variant 6
Count
640(C)-A :: AF647-HER2

FIG. 23C

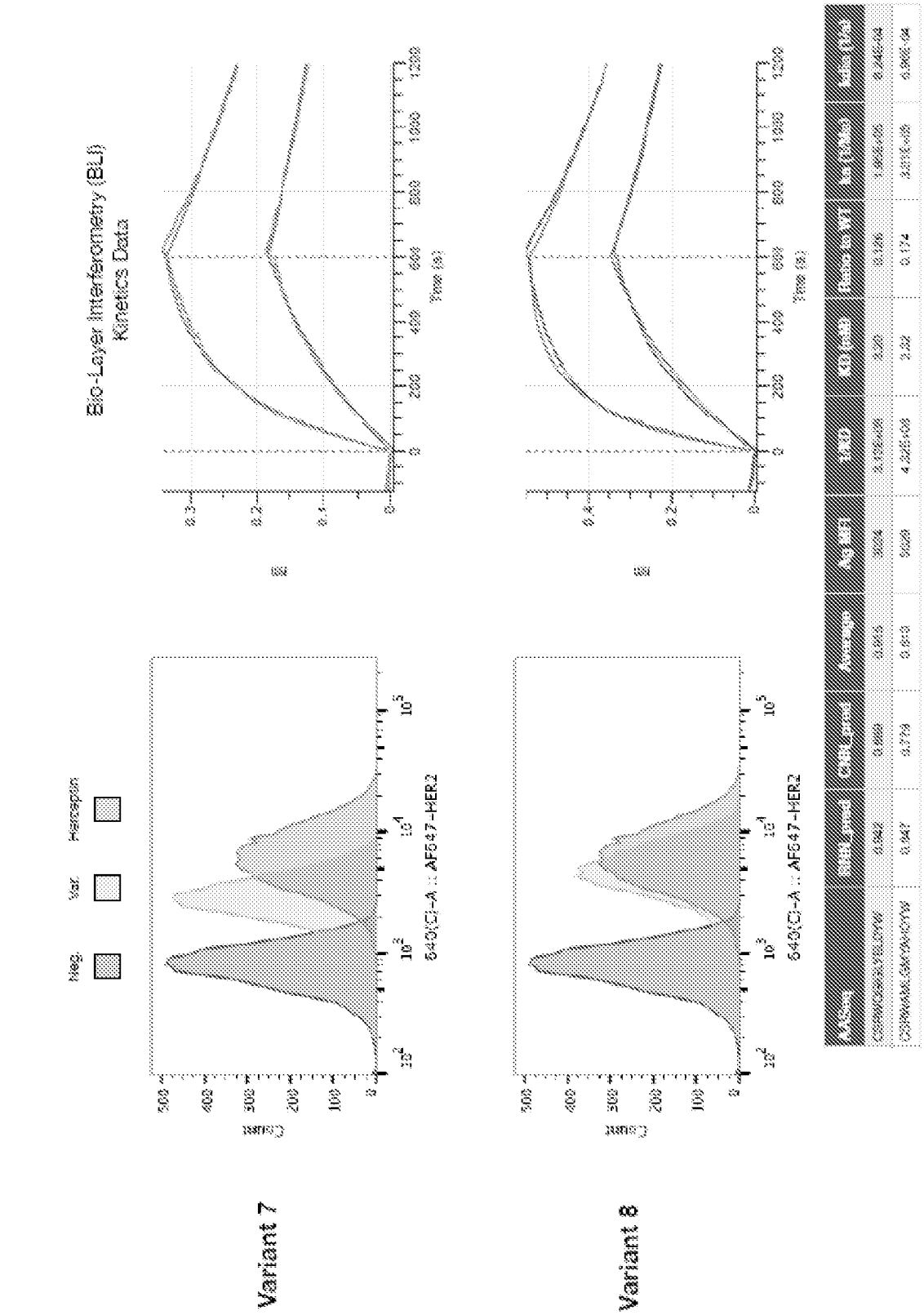

Bio-Layer Interferometry (BLI)
Kinetics Data
Neg.
Var.
Receptor
Variant 7
Variant 8
Count
640(C)-A :: AF647-HER2
nm
Time (s)

FIG. 23D

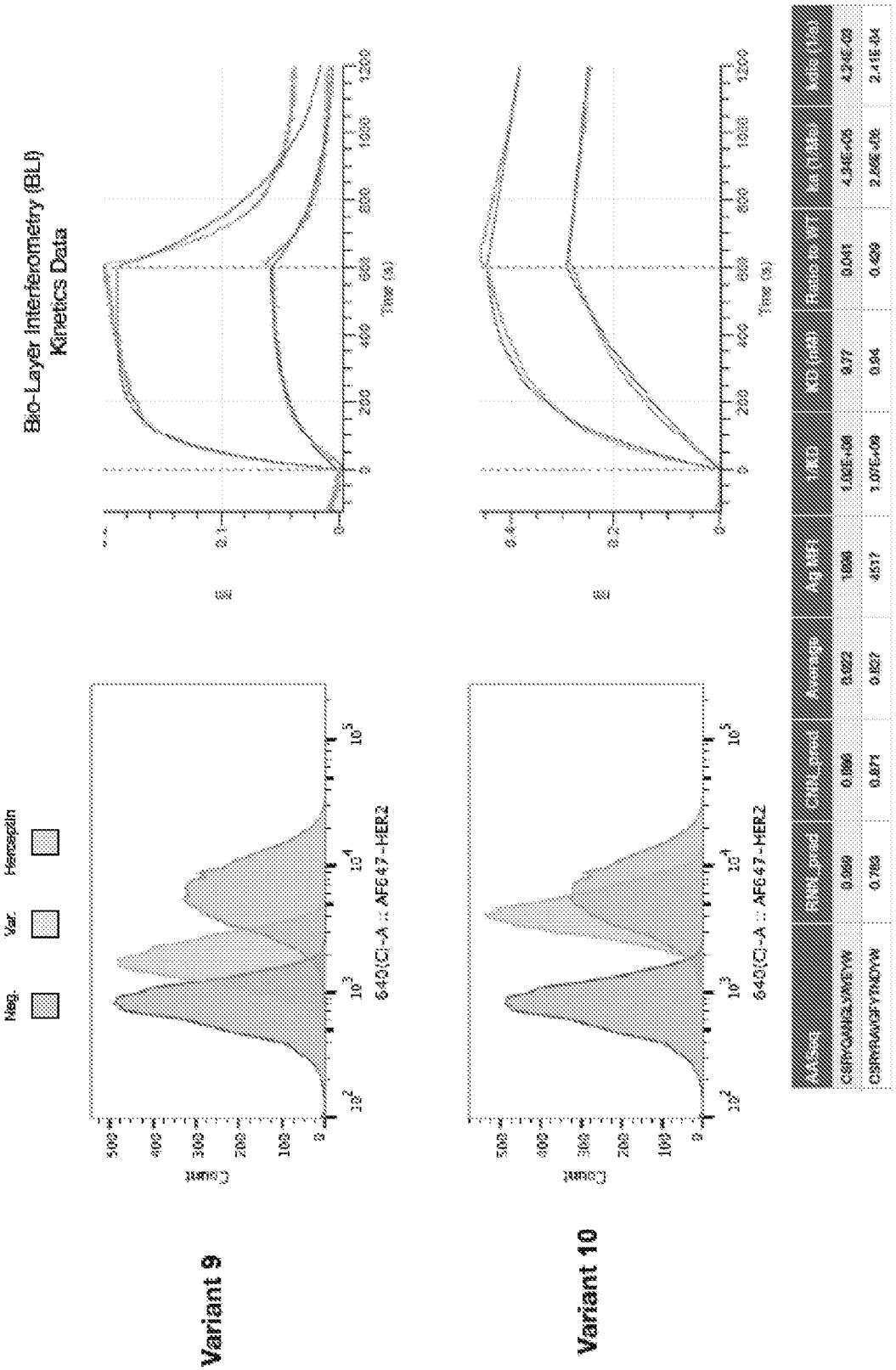

Bio-Layer Interferometry (BLI)
Kinetics Data
Variant 9
Variant 10
Neg.
Var.
Herceptin
Count
640(C)-A :: AF647-HER2
Time (s)
nm

FIG. 23E

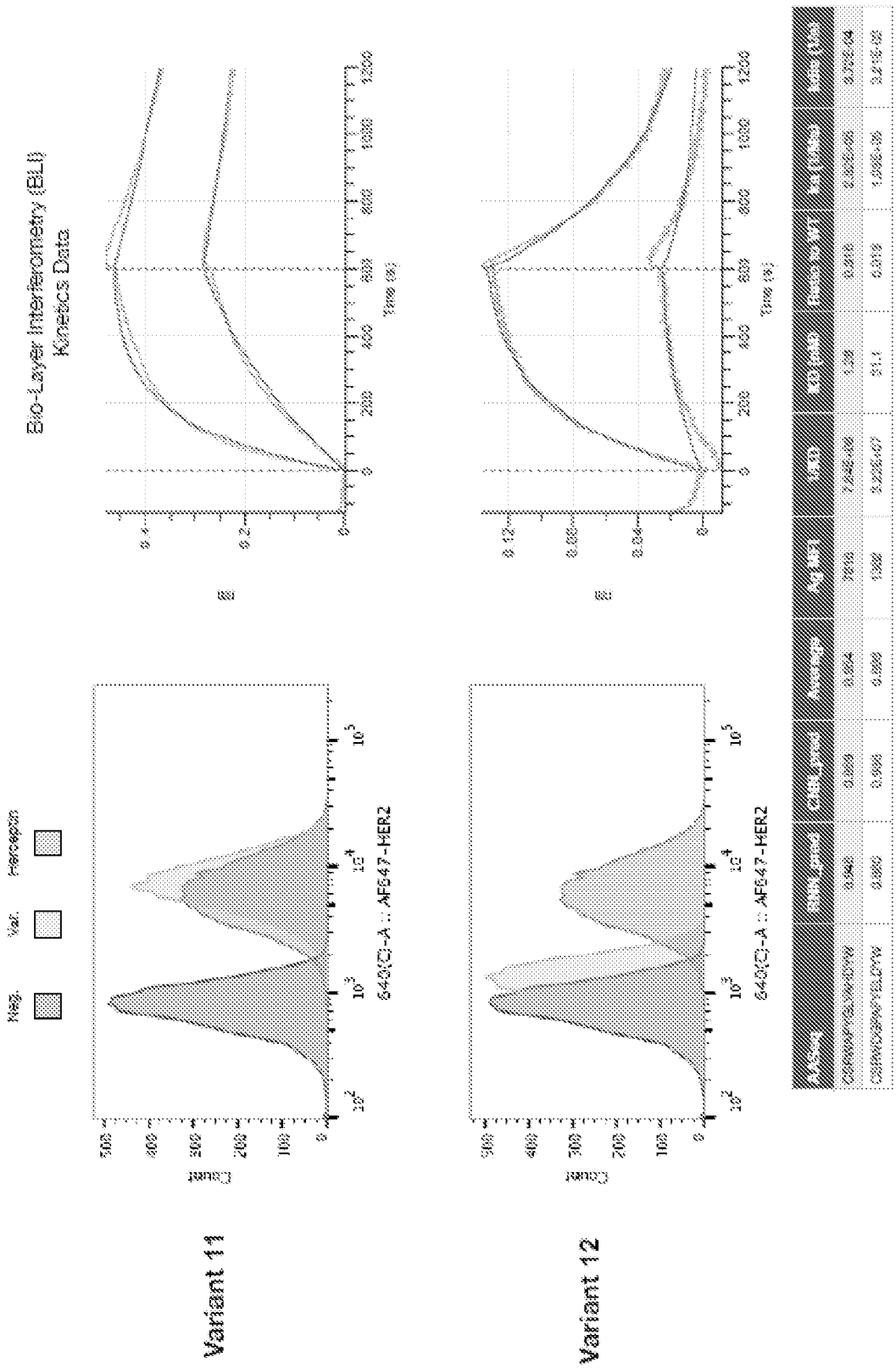
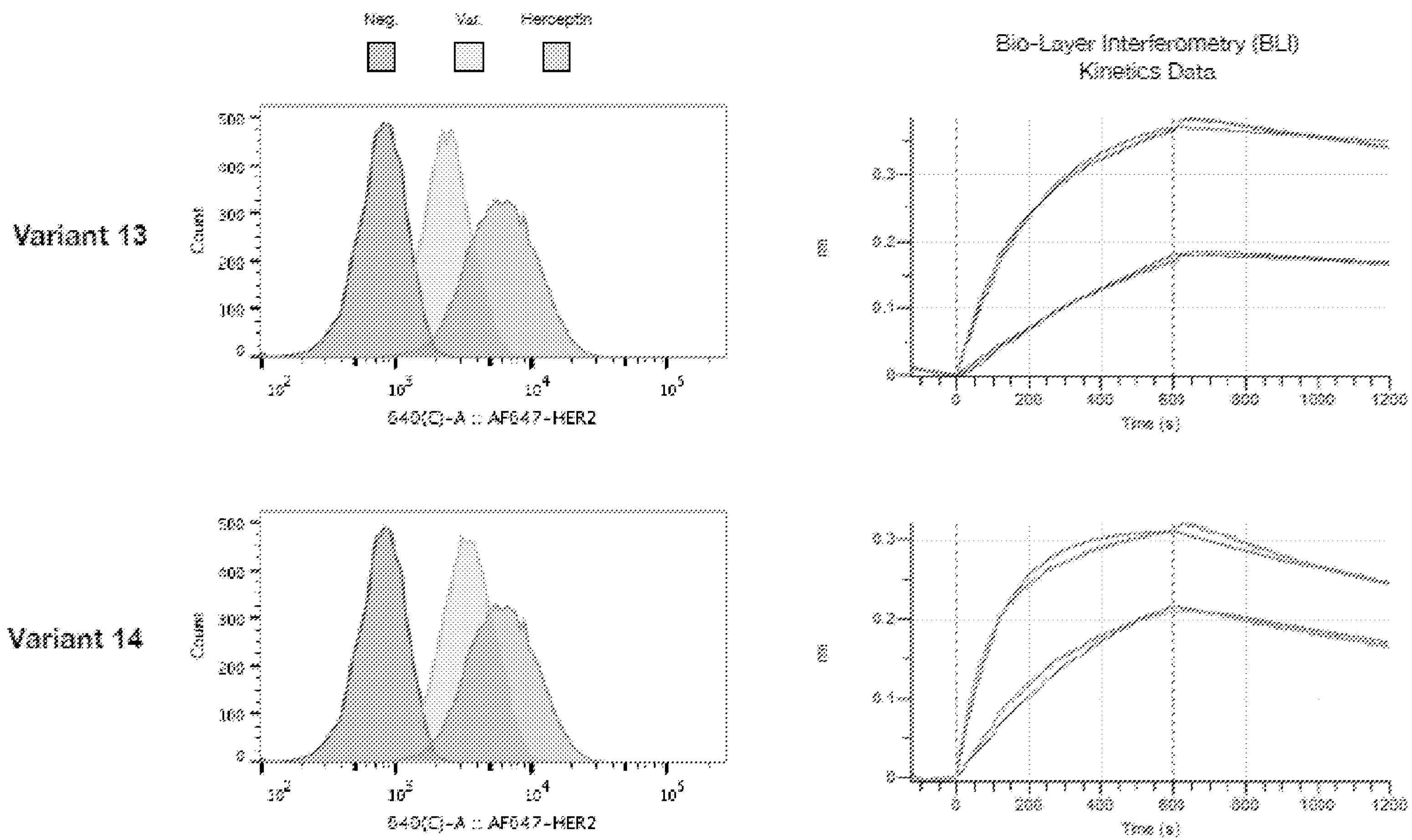

Bio-Layer Interferometry (BLI)
Kinetics Data
Neg.
Var.
Perception
Variant 11
Variant 12
Count
640(C)-A :: AF647-HER2
Time (s)

FIG. 23F



AASeq	RNN_pred	CNN_pred	Average	Ag MFI	1/KD	KD (nM)	Ratio to WT	ka (1/Ms)	kdis (1/s)
CSRWGHSFYEHDYW	0.844	0.899	0.867	2517	1.86E+09	0.54	0.748	1.79E+06	9.60E-02
CSRYGEYGMYVNKYW	0.883	0.870	0.879	3801	8.07E+08	1.24	0.925	8.23E+05	4.01E-04

FIG. 23G

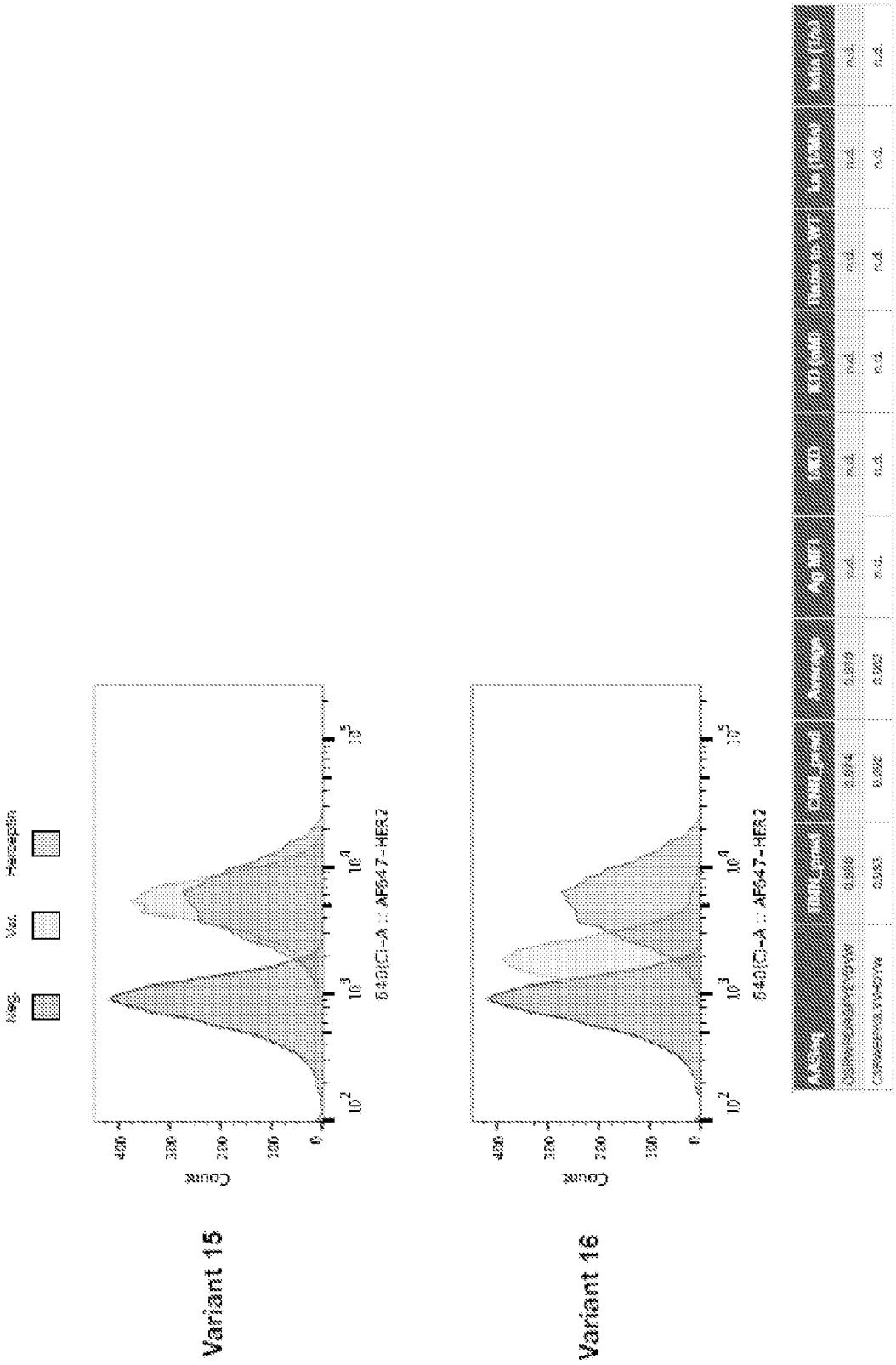
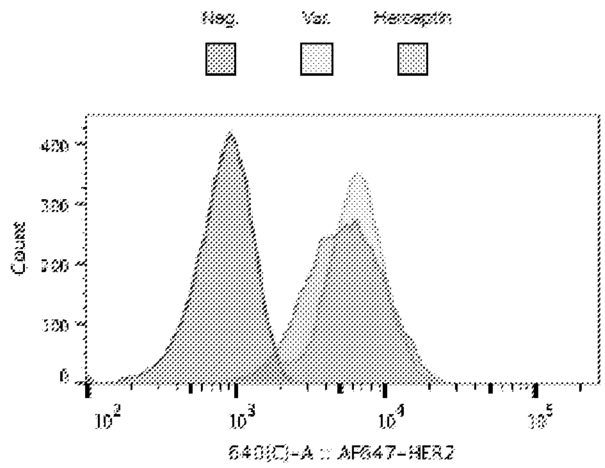
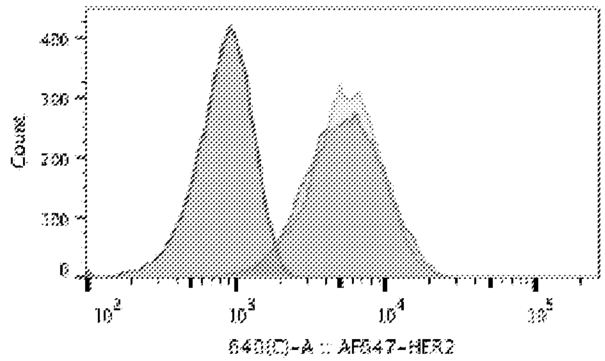


FIG. 23H

Variant 17



Variant 18

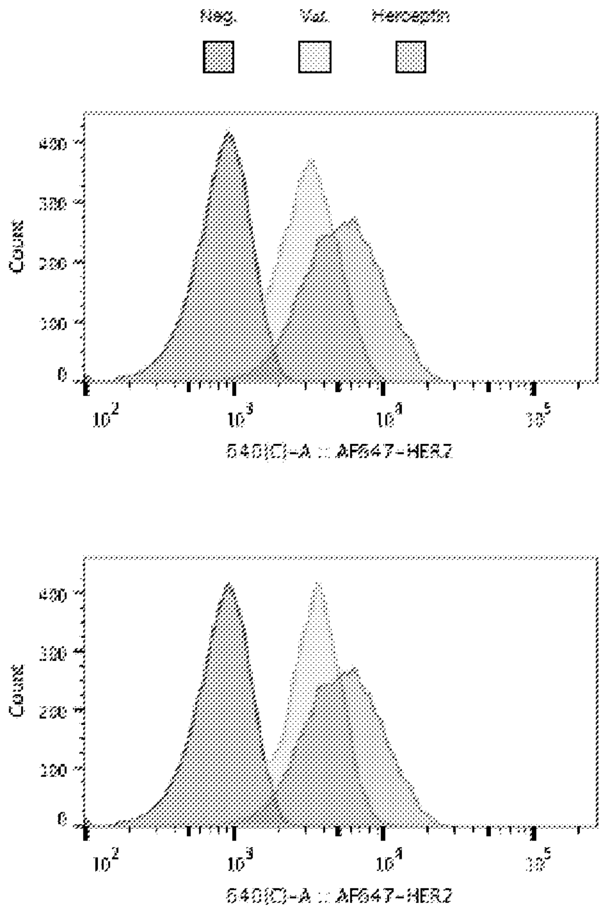


AASeq	RNN_pred	CNN_pred	Average	Ag MFI	1/KD	KD (nM)	Ratio to WT	ka (1/Ms)	kdis (1/s)
CSRYASAGMYTHQYW	0.927	0.883	0.905	n.d.	n.d.	n.d.	n.d.	n.d.	n.d.
CSRYGDAGMYALKYW	0.973	0.995	0.984	n.d.	n.d.	n.d.	n.d.	n.d.	n.d.

FIG. 23I

Variant 19

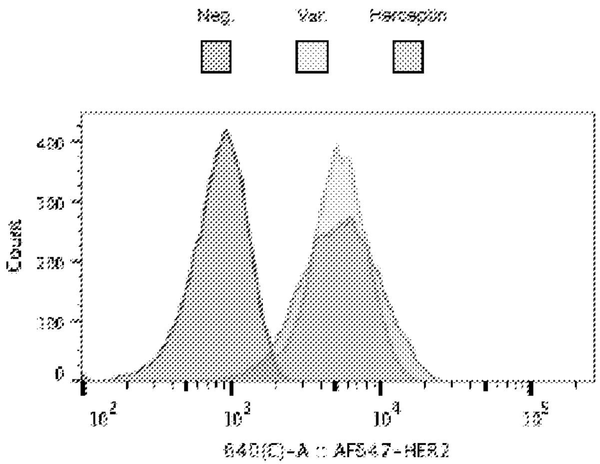
Variant 20



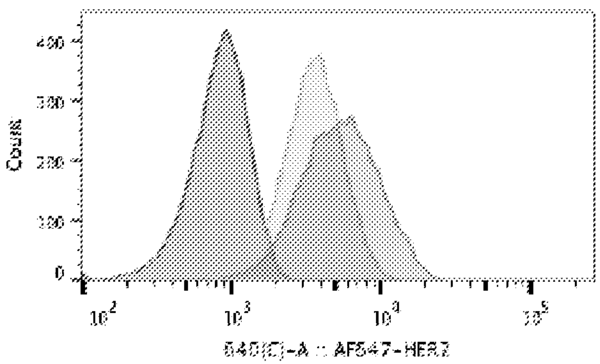
AASeq	RBM_pred	CNN_pred	Average	Ag MFI	1/KD	KD (nM)	Ratio to WT	ka (1/Ms)	kdis (1/s)
CSRWGLGGMYTHDYW	0.919	0.841	0.880	n.d.	n.d.	n.d.	n.d.	n.d.	n.d.
CSRWNSSGLYAYEYW	0.864	0.961	0.913	n.d.	n.d.	n.d.	n.d.	n.d.	n.d.

FIG. 23J

Variant 21



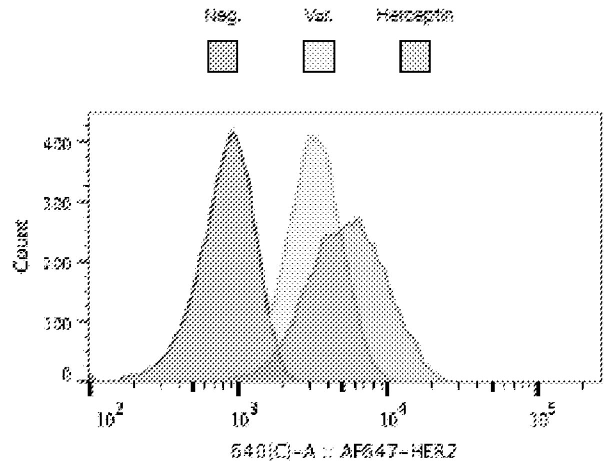
Variant 22



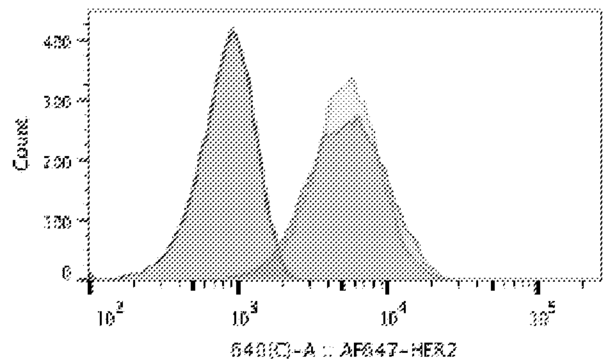
AASeq	RNN_pred	CNN_pred	Average	Ag MFI	1/KD	KD (pM)	Ratio to WT	ka (1/Ms)	kdis (1/s)
CSRWQPGGFYAMDYW	0.958	0.989	0.974	n.d.	n.d.	n.d.	n.d.	n.d.	n.d.
CSRYGARGFYQNDYW	0.898	0.783	0.840	n.d.	n.d.	n.d.	n.d.	n.d.	n.d.

FIG. 23K

Variant 23

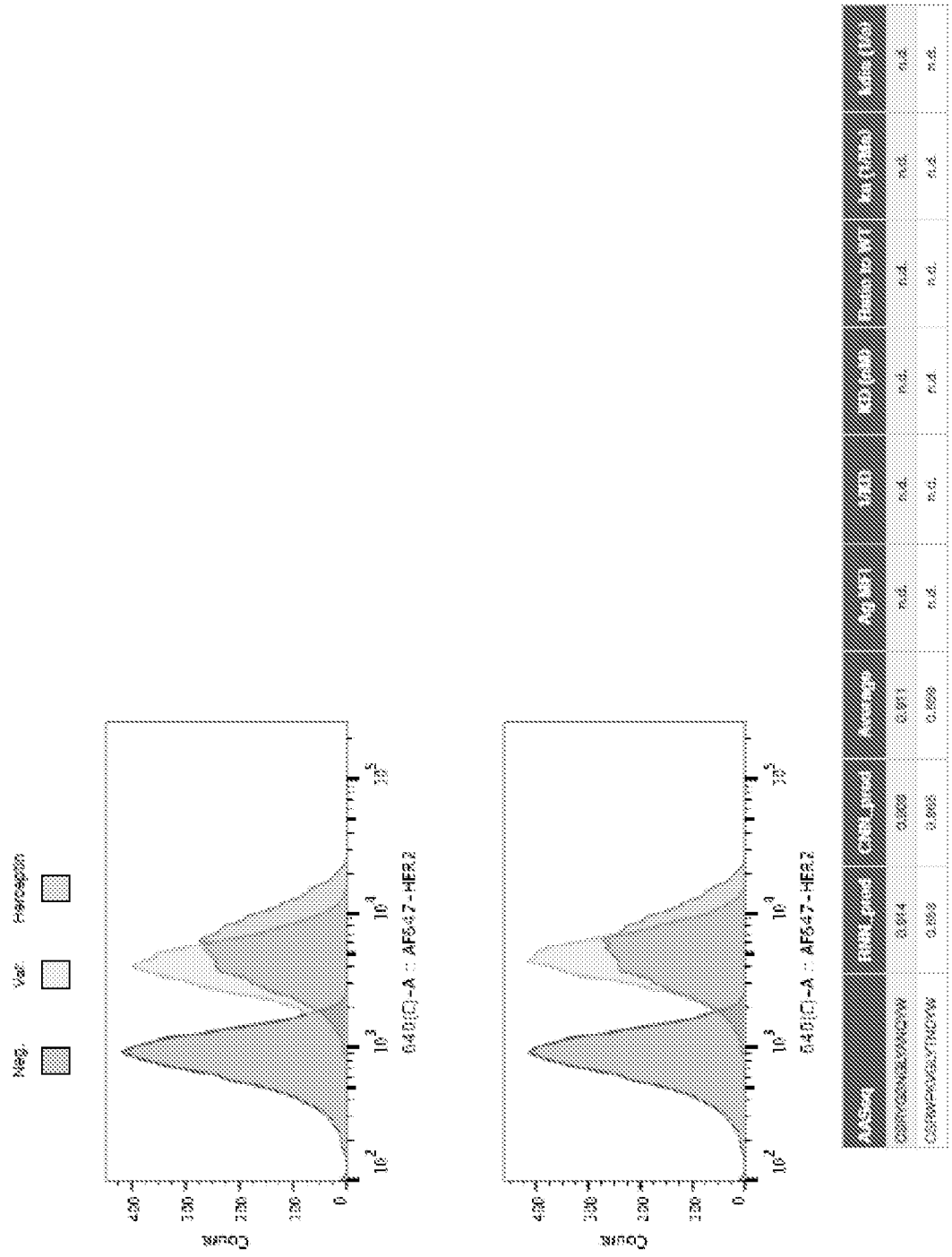


Variant 24



AASeq	RNN_pred	CNN_pred	Average	Ag MFI	1/KD	KD (nM)	Ratio to WT	ka (1/Ms)	kdis (1/s)
CSRYAGPSMYTNDYW	0.870	0.930	0.900	n.d.	n.d.	n.d.	n.d.	n.d.	n.d.
CSRWNFHGLYVNDYW	0.939	0.974	0.956	n.d.	n.d.	n.d.	n.d.	n.d.	n.d.

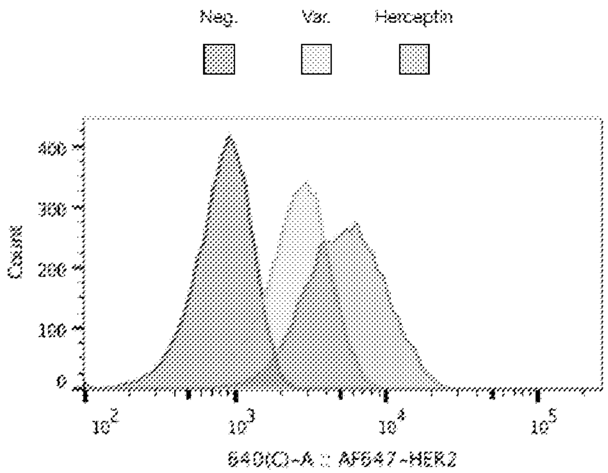
FIG. 23L



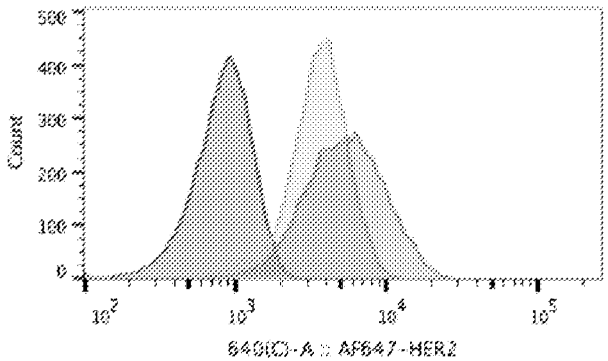
Assay	Background	Background	Average	Assay	Assay	Assay	Assay	Assay
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]

FIG. 23M

Variant 27



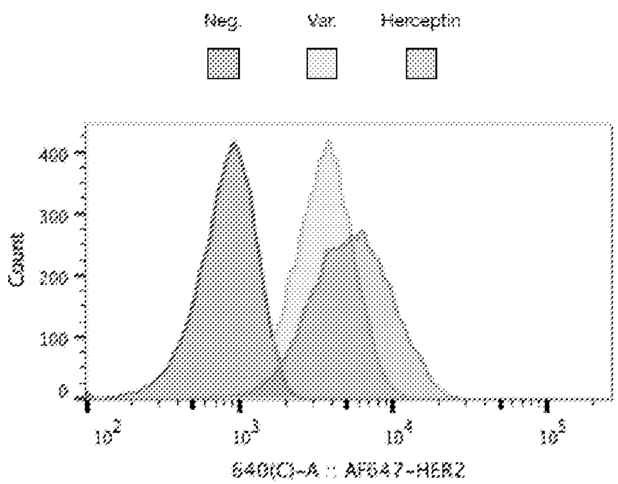
Variant 28



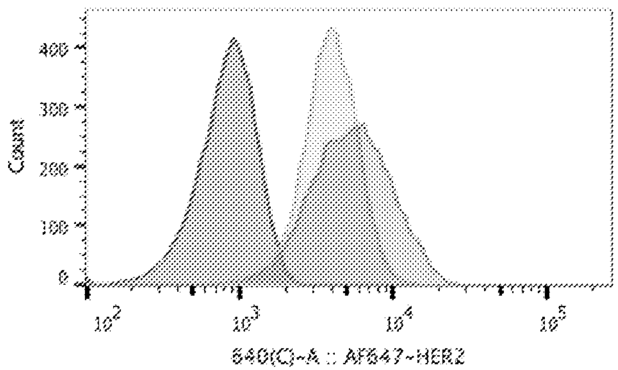
AASeq	RNN_pred	CNN_pred	Average	Ag MFI	1/KD	KD (nM)	Ratio to WT	ka (1/Ms)	kdis (1/s)
CSRWGVSFYEMDYW	0.871	0.873	0.872	n.d.	n.d.	n.d.	n.d.	n.d.	n.d.
CSRYSMPGMYTNAYW	0.848	0.938	0.893	n.d.	n.d.	n.d.	n.d.	n.d.	n.d.

FIG. 23N

Variant 29



Variant 30



AASeq	RNN_pred	CNN_pred	Average	Ag MFI	1/KD	KD (nM)	Ratio to WT	ka (1/Ms)	kdis (1/s)
CSRWAEAGMYEFDYW	0.880	0.915	0.898	n.d.	n.d.	n.d.	n.d.	n.d.	n.d.
CSRWPMCGMYTHDYW	0.935	0.955	0.945	n.d.	n.d.	n.d.	n.d.	n.d.	n.d.

FIG. 23O

Target/ Antigen	Working conc.	Incubation volume	Fluorophore	Product ID
Human IgG (H+L)	1.5 ug/mL	100 µl	AlexaFluor® 488	109-545-088 (Jackson ImmunoReseatch)
ErbB2 (HER2) Round 1	2.9 µg/ml (40 nM)	100 µl	AlexaFluor® 647	SRP6405-50UG (Sigma-Aldrich)
ErbB2 (HER2) Round 2	1.4 µg/ml (20 nM)	100 µl	AlexaFluor® 647	SRP6405-50UG (Sigma-Aldrich)
ErbB2 (HER2) Round 3	1.4 µg/ml (20 nM)	100 µl	AlexaFluor® 488	SRP6405-50UG (Sigma-Aldrich)

FIG. 24A

Target/ Antigen	Working conc.	Incubation volume	Fluorophore	Product ID
Human IgG (H+L)	1.5 ug/mL	100 µl	AlexaFluor® 488	109-545-088 (Jackson ImmunoResearch)
ErbB2 (HER2) Round 1	2.9 µg/ml (40 nM)	100 µl	AlexaFluor® 647	SRP6405-50UG (Sigma-Aldrich)
ErbB2 (HER2) Round 2a	1.4 µg/ml (20 nM)	100 µl	AlexaFluor® 488	SRP6405-50UG (Sigma-Aldrich)
ErbB2 (HER2) Round 2b	1.4 µg/ml (20 nM)	100 µl	AlexaFluor® 647	SRP6405-50UG (Sigma-Aldrich)

FIG. 24B

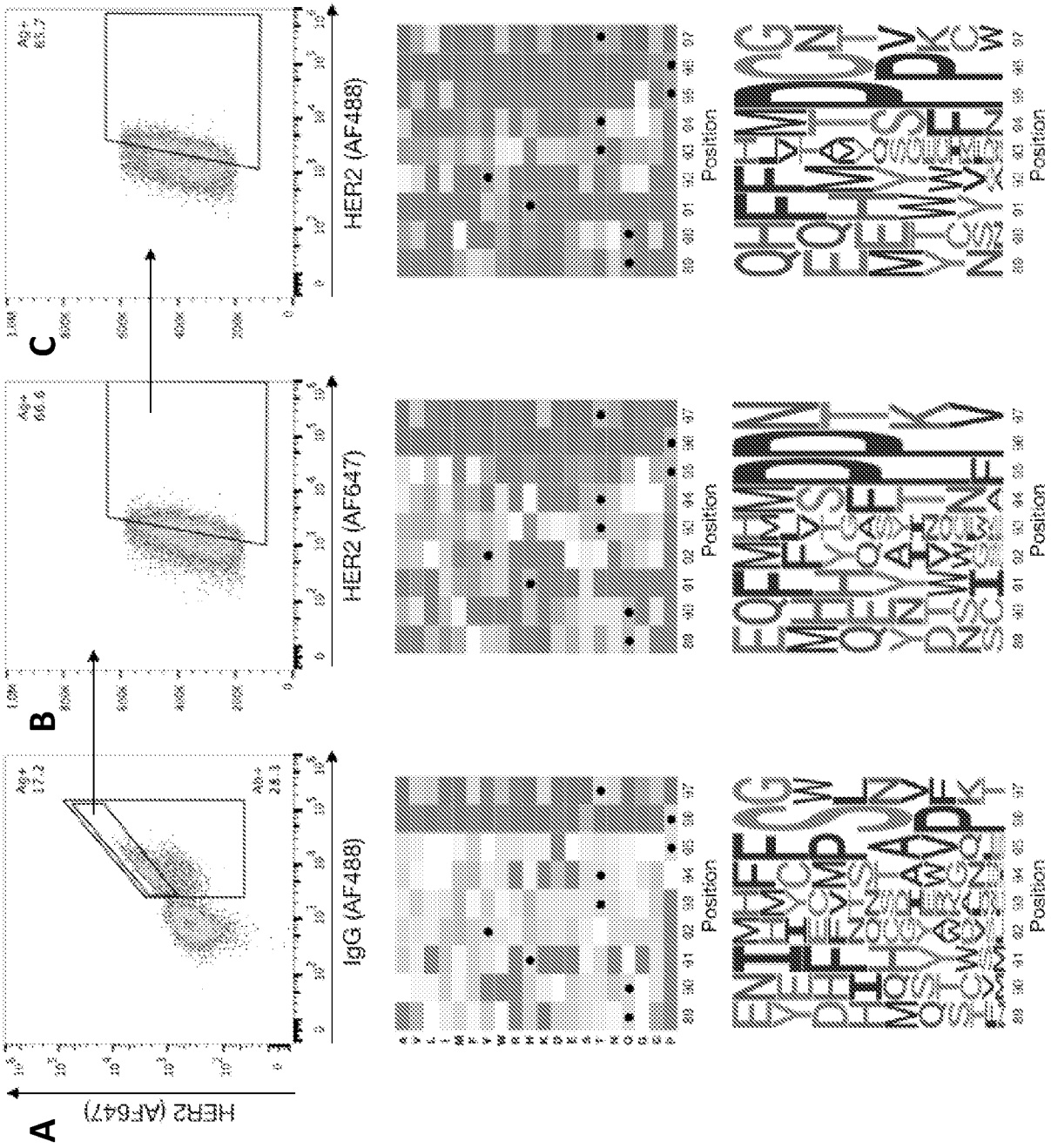


FIG. 25

Sample	Description	Raw Read Count (post-merge)	Aligned Reads	Unique CDRL3s
DMS-L3-Ab	Library of antibody expressing hybridomas following transfection with NNK-tiled mutagenesis ssODN donors	2,243,308	1,352,963	1,529
DMS-L3-Ag1	Library of antigen specific antibodies expressed in hybridomas following transfection with NNK-tiled mutagenesis ssODN donors	2,952,330	1,805,240	491
DMS-L3-Ag2	Library of antigen specific antibodies in hybridomas following two rounds of enrichment for antigen binding variants	547,930	327,415	311
DMS-L3-Ag3	Library of antigen specific antibodies in hybridomas following three rounds of enrichment for antigen binding variants	2,494,602	1,513,569	246

FIG. 26

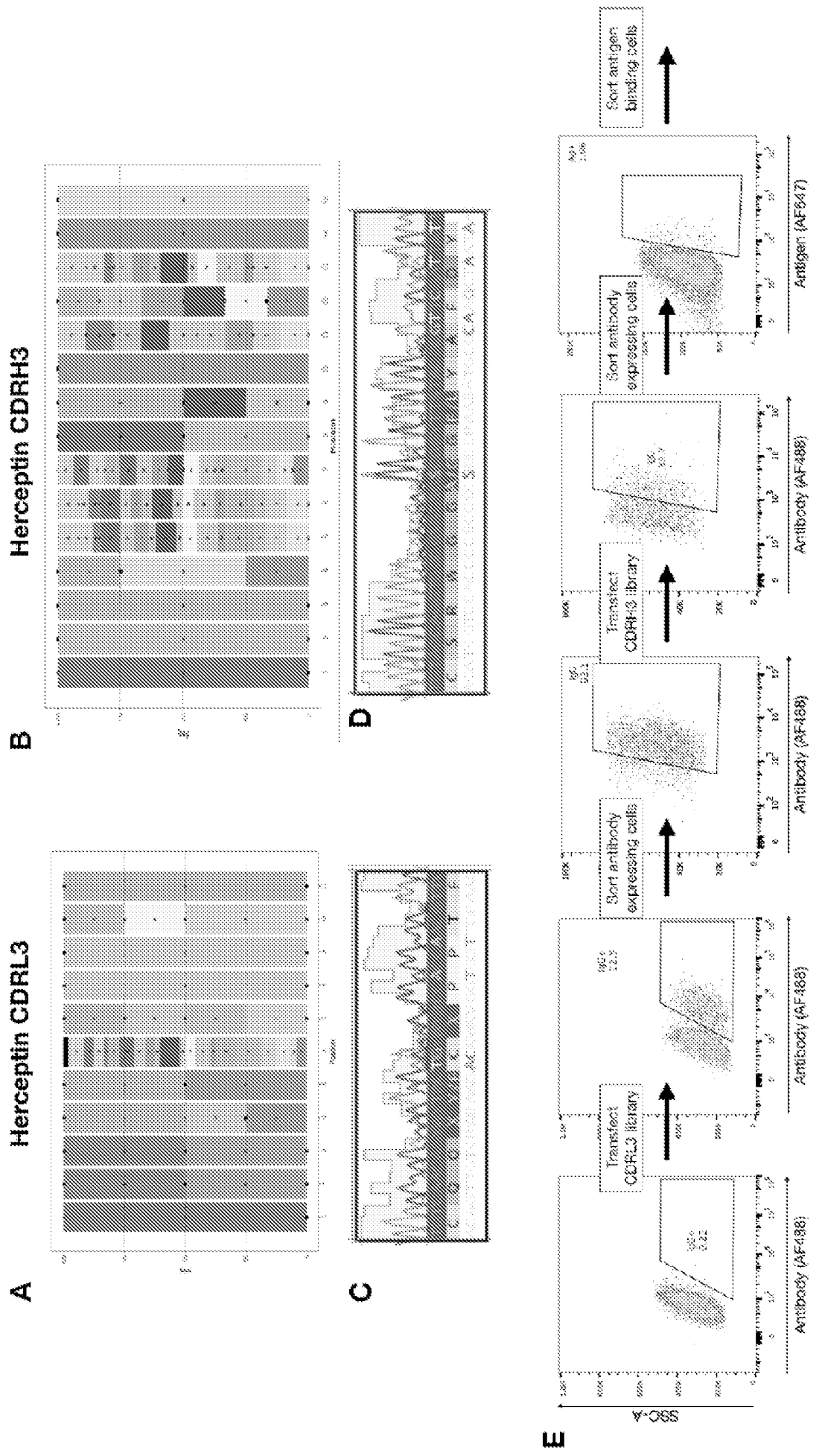

A
Herceptin CDRL3
B
Herceptin CDRH3
C
D
E
SSC-A
Antibody (AF488)
Antigen (AF647)
Transfect CDRL3 library
Transfect CDRH3 library
Sort antibody expressing cells
Sort antigen binding cells

FIG. 27

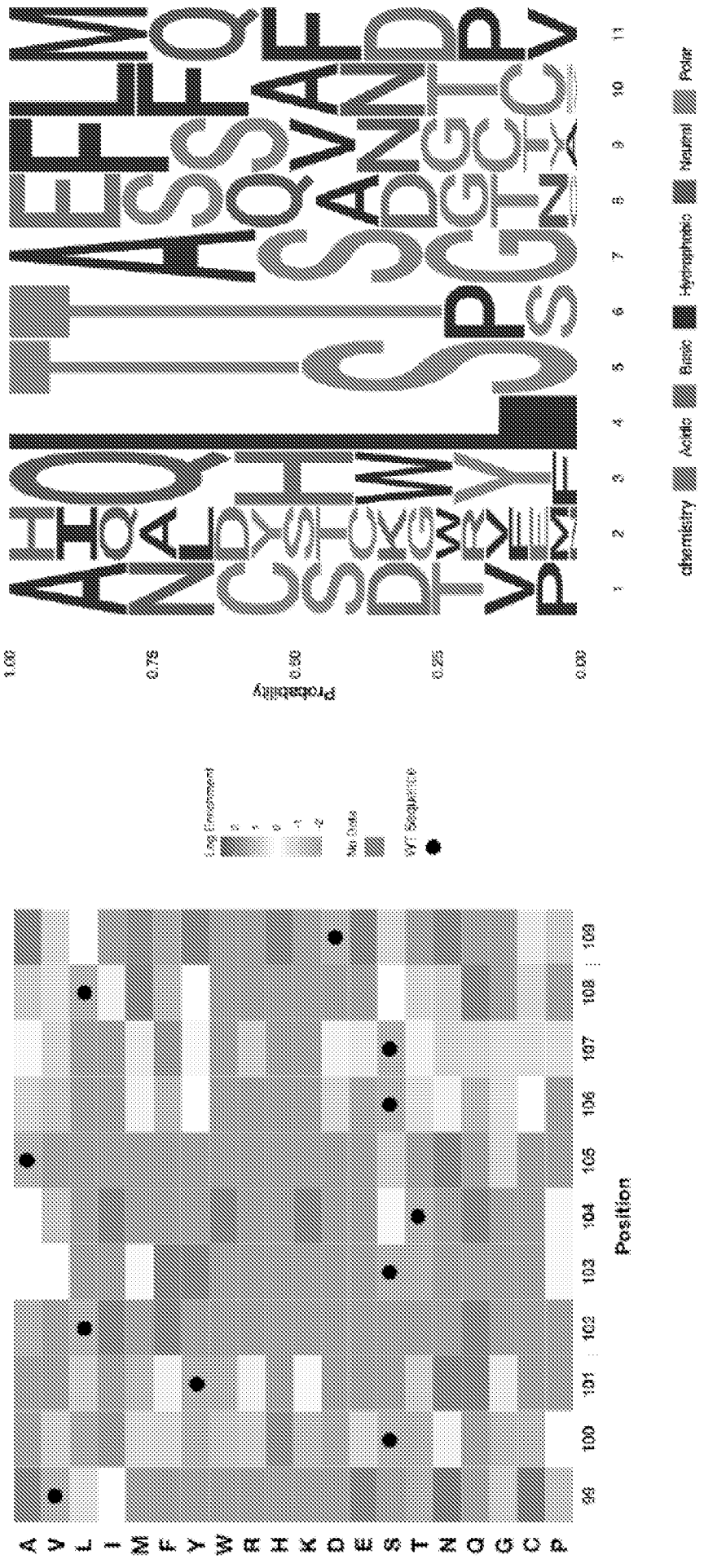

Probability
1.00
0.75
0.50
0.25
0.00
chemistry
Acidic
Basic
Hydrophobic
Neutral
Polar
Log Enrichment
Mut Data
WT Sequence
Position
A V L I M F Y W R H K D E S T N Q G C P
98
99
100
101
102
103
104
105
106
107
108
109

FIG. 28

Sample	Description	Raw Read count (post-merge)	Aligned Reads	Unique CDRH3s
DMS-H3-Ab	Library of antibody expressing hybridomas following transfection with NNK-tiled mutagenesis ssODN donors	118,040	84,995	1,102
DMS-H3-Ag1	Library of antigen specific antibodies expressed in hybridomas following transfection with NNK-tiled mutagenesis ssODN donors	74,154	50,638	415
DMS-H3-Ag2	Library of antigen specific antibodies in hybridomas following two rounds of enrichment for antigen binding variants	127,705	82,245	446

FIG. 29

INTERNATIONAL SEARCH REPORT

International application No
PCT/IB2020/053370

A. CLASSIFICATION OF SUBJECT MATTER
INV. G16B20/20 G16B20/30
ADD.

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)
G16B

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)
EPO-Internal, WPI Data, EMBASE, BIOSIS

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	WO 2018/132752 A1 (MASSACHUSETTS INST TECHNOLOGY [US]; ZENG HAOYANG [US]) 19 July 2018 (2018-07-19)	1-88
Y	Whole document, in particular claims; pages 35-40	4,57-62

Y	D. KURODA ET AL: "Computer-aided antibody design", PROTEIN ENGINEERING DESIGN AND SELECTION, vol. 25, no. 10, 1 October 2012 (2012-10-01), pages 507-522, XP055056463, ISSN: 1741-0126, DOI: 10.1093/protein/gzs024	57-62
A	whole document, in particular abstract, introduction and perspectives concluding remarks	1-56, 63-88

	-/--	

[X] Further documents are listed in the continuation of Box C.

[X] See patent family annex.

* Special categories of cited documents :

"A" document defining the general state of the art which is not considered to be of particular relevance

"E" earlier application or patent but published on or after the international filing date

"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)

"O" document referring to an oral disclosure, use, exhibition or other means

"P" document published prior to the international filing date but later than the priority date claimed

"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone

"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art

"&" document member of the same patent family

Date of the actual completion of the international search	Date of mailing of the international search report
23 July 2020	03/08/2020

Name and mailing address of the ISA/	Authorized officer
European Patent Office, P.B. 5818 Patentlaan 2 NL - 2280 HV Rijswijk Tel. (+31-70) 340-2040, Fax: (+31-70) 340-3016	Vanmontfort, D

INTERNATIONAL SEARCH REPORT

International application No.
PCT/IB2020/053370

Box No. II Observations where certain claims were found unsearchable (Continuation of item 2 of first sheet)

This international search report has not been established in respect of certain claims under Article 17(2)(a) for the following reasons:

1. ☐ Claims Nos.:
because they relate to subject matter not required to be searched by this Authority, namely:

2. ☒ Claims Nos.: 89-123
because they relate to parts of the international application that do not comply with the prescribed requirements to such an extent that no meaningful international search can be carried out, specifically:
see FURTHER INFORMATION sheet PCT/ISA/210

3. ☐ Claims Nos.:
because they are dependent claims and are not drafted in accordance with the second and third sentences of Rule 6.4(a).

Box No. III Observations where unity of invention is lacking (Continuation of item 3 of first sheet)

This International Searching Authority found multiple inventions in this international application, as follows:

1. ☐ As all required additional search fees were timely paid by the applicant, this international search report covers all searchable claims.

2. ☐ As all searchable claims could be searched without effort justifying an additional fees, this Authority did not invite payment of additional fees.

3. ☐ As only some of the required additional search fees were timely paid by the applicant, this international search report covers only those claims for which fees were paid, specifically claims Nos.:

4. ☐ No required additional search fees were timely paid by the applicant. Consequently, this international search report is restricted to the invention first mentioned in the claims; it is covered by claims Nos.:

Remark on Protest

☐ The additional search fees were accompanied by the applicant's protest and, where applicable, the payment of a protest fee.

☐ The additional search fees were accompanied by the applicant's protest but the applicable protest fee was not paid within the time limit specified in the invitation.

☐ No protest accompanied the payment of additional search fees.

INTERNATIONAL SEARCH REPORT

International application No
PCT/IB2020/053370

C(Continuation). DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	WO 2014/180490 A1 (BIONTECH AG [DE] ET AL.) 13 November 2014 (2014-11-13) Whole document, in particular claims and Figure 1	1-88

Y	EDGAR LIBERIS ET AL: "Parapred: Antibody Paratope Prediction using Convolutional and Recurrent Neural Networks", BIOINFORMATICS., 8 September 2017 (2017-09-08), XP055469248, GB ISSN: 1367-4803, DOI: 10.1093/bioinformatics/bty305	4
A	the whole document	1-3,5-88

INTERNATIONAL SEARCH REPORT

Information on patent family members

International application No
PCT/IB2020/053370

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
WO 2018132752 A1	19-07-2018	EP 3568782 A1	20-11-2019
		US 2019065677 A1	28-02-2019
		WO 2018132752 A1	19-07-2018
WO 2014180490 A1	13-11-2014	AU 2014264943 A1	26-11-2015
		AU 2019275637 A1	02-01-2020
		CA 2911945 A1	13-11-2014
		CN 105451759 A	30-03-2016
		EP 2994159 A1	16-03-2016
		HK 1215169 A1	19-08-2016
		IL 242281 A	30-04-2020
		JP 6710634 B2	17-06-2020
		JP 2016521128 A	21-07-2016
		JP 2020103297 A	09-07-2020
		KR 20160030101 A	16-03-2016
		RU 2015153007 A	16-06-2017
		SG 11201508816R A	27-11-2015
		US 2016125129 A1	05-05-2016
		WO 2014180490 A1	13-11-2014
		WO 2014180569 A1	13-11-2014
		ZA 201508048 B	31-05-2017

Continuation of Box II.2

Claims Nos.: 89-123

Present claims 89-111 relate to a protein or peptide, wherein the amino acid sequence of the protein or peptide is identified/generated by the production/screening method of any one of claims 1 to 87, or the system of claim 88.

The protein and peptide is structurally undefined, but is defined by its function on the basis of binding affinity properties and optionally on the basis of physicochemical properties, on which the in silico screening method of claims 1-87 is based. These claims are thus considered to represent "reach-through formulations", that are not accepted under Articles 5 and 6 PCT. As a matter of fact, the subject matter of said claims covers limitless and untried downstream developments in relation to yet to be demonstrated functions/activities. The claims amount to no more than an invitation to set up further research programmes for which no guidance is forthcoming and, therefore, it is an undue burden to put the claimed subject matter into practice, i.e. to identify all the relevant compounds having said desired property without indication of any structural limitation. A further lack of clarity (Article 6 PCT) arises because the public can not ascertain whether or not a particular compound falls within the scope of such a claim. Consequently, the examination can never with any certainty, ascertain whether or not such claims are distinguished over the state-of-the-art.

Claim 104, which refers to "a protein or peptide comprising an amino acid sequence depicted in FIGURE 15A to 15D or in FIGURE 23A to 2" is so unclear that no meaningful search is possible. Firstly, the wording of said claim is not restricted to the full-length amino acid sequence as listed in the Figures but also includes any fragment of said sequences. Secondly, it is not clear which SEQ IDs of the sequence listings correspond to the ones listed in said Figures.

The same applies to a cell comprising said protein or peptide (claims 112-117) and to the use of said protein or peptide (claims 112-123).

The applicant's attention is drawn to the fact that claims relating to inventions in respect of which no international search report has been established need not be the subject of an international preliminary examination (Rule 66.1(e) PCT). The applicant is advised that the EPO policy when acting as an International Preliminary Examining Authority is normally not to carry out a preliminary examination on matter which has not been searched. This is the case irrespective of whether or not the claims are amended following receipt of the search report or during any Chapter II procedure. If the application proceeds into the regional phase before the EPO, the applicant is reminded that a search may be carried out during examination before the EPO (see EPO Guidelines C-IV, 7.2), should the problems which led to the Article 17(2) declaration be overcome.