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

(19) World Intellectual Property Organization
International Bureau

(43) International Publication Date
12 December 2024 (12.12.2024)

WIPO | PCT

(10) International Publication Number
WO 2024/251783 A1

(51) International Patent Classification:
G16B 30/00 (2019.01) *G16B 40/20* (2019.01)

(21) International Application Number:
PCT/EP2024/065419

(22) International Filing Date:
05 June 2024 (05.06.2024)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:
23305893.2 05 June 2023 (05.06.2023) EP
24305323.8 01 March 2024 (01.03.2024) EP

(71) Applicant: **SANOFI** [FR/FR]; 46 avenue de la Grande Armée, 75017 Paris (FR).

(72) Inventors: **GUPTA, Abhinav**; 46 avenue de la Grande Armée, 75017 Paris (FR). **QIU, Yu**; 46 avenue de la Grande Armée, 75017 Paris (FR). **JAGER, Sven**; 46 avenue de la Grande Armée, 75017 Paris (FR). **CORROCHANO NAVARRO, Alejandro**; 46 avenue de la Grande Armée, 75017 Paris (FR). **BAILEY, Michael**; 46 avenue de la Grande Armée, 75017 Paris (FR). **AMMENGUAL RIGO, Jose Antonio**; 46 avenue de la Grande Armée, 75017 Paris (FR). **FURTMANN, Norbert**; 46 avenue de la Grande Armée, 75017 Paris (FR). **ROMMELAERE, Heidi**; 46 avenue de la Grande Armée, 75017 Paris (FR).

(74) Agent: **FISH & RICHARDSON P.C.**; Highlight Business Towers, Mies-van-der-Rohe-Str. 8, 80807 Munich (DE).

(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, CV, CZ, DE, DJ, DK, DM, DO, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IQ, IR, IS, IT, JM, JO, JP, KE, KG, KH, KN, KP, KR, KW, KZ, LA, LC, LK, LR, LS, LU, LY, MA, MD, MG, MK, MN, MU, 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, CV, GH, GM, KE, LR, LS, MW, MZ, NA, RW, SC, SD, SL, ST, SZ, TZ, UG, ZM, ZW), Eurasian (AM, AZ, BY, KG, KZ, RU, TJ, 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, ME, MK, MT, NL, NO, PL, PT, RO, RS, SE,

(54) Title: PREDICTING THERMAL STABILITIES OF IMMUNOGLOBULIN SINGLE VARIABLE DOMAINS USING MACHINE-LEARNING MODELS

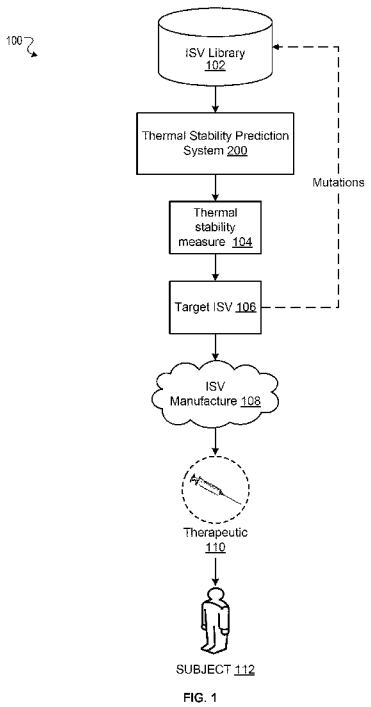


FIG. 1

(57) Abstract: Methods, computer systems, and apparatus, including computer programs encoded on computer storage media, for predicting the thermal stability of immunoglobulin single variable domains (ISVs). The system obtains data representing an amino acid sequence of the ISV, generates an input token vector by numerically encoding the amino acid sequence, generates an embedded feature vector by processing the input token vector using an embedding machine-learning model having a first set of model parameters, and processes an input including the embedded feature vector using a prediction machine-learning model to generate an output that predicts a thermal stability measure of the ISV.

WO 2024/251783 A1

[Continued on next page]

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))*

— *in black and white; the international application as filed contained color or greyscale and is available for download from PATENTSCOPE*

PREDICTING THERMAL STABILITIES OF IMMUNOGLOBULIN SINGLE VARIABLE DOMAINS USING MACHINE-LEARNING MODELS

CROSS-REFERENCE TO RELATED APPLICATION

[0001] This application claims priorities to European Patent Application No. EP23305893, filed on June 5, 2023, and European Patent Application No. EP24305323, filed on March 1, 2024, the disclosures of both of which are hereby incorporated by reference in its entirety.

TECHNICAL FIELD

[0002] This specification generally relates to predicting the thermal stability of immunoglobulin single variable domains (ISVs) using machine-learning models.

BACKGROUND

[0003] The term "immunoglobulin single variable domain" (ISV), interchangeably used with "single variable domain", defines immunoglobulin molecules wherein the antigen binding site is present on, and formed by, a single immunoglobulin domain. This sets immunoglobulin single variable domains apart from "conventional" immunoglobulins (e.g. monoclonal antibodies) or their fragments (such as Fab, Fab', $F(ab')_2$, scFv, di-scFv), wherein two immunoglobulin domains, in particular two variable domains, interact to form an antigen binding site. Typically, in conventional immunoglobulins, a heavy chain variable domain (V_H) and a light chain variable domain (V_L) interact to form an antigen binding site. In this case, the complementarity determining regions (CDRs) of both V_H and V_L will contribute to the antigen binding site, i.e. a total of 6 CDRs will be involved in antigen binding site formation.

[0004] In contrast, immunoglobulin single variable domains are capable of specifically binding to an epitope of the antigen without pairing with an additional immunoglobulin variable domain. The binding site of an immunoglobulin single variable domain is formed by a single V_H, a single V_{HH} or single V_L domain. Hence, the antigen binding site of an immunoglobulin single variable domain is formed by no more than three CDRs.

[0005] As such, the single variable domain may be a light chain variable domain sequence (e.g., a V_L-sequence) or a suitable fragment thereof; or a heavy chain variable domain sequence (e.g., a V_H-sequence or V_{HH} sequence) or a suitable fragment thereof; as long as it is capable of forming a single antigen binding unit (i.e., a functional antigen binding unit that essentially

consists of the single variable domain, such that the single antigen binding domain does not need to interact with another variable domain to form a functional antigen binding unit).

[0006] An immunoglobulin single variable domain (ISV) can for example be a heavy chain ISV, such as a V_H, V_{HH}, including a camelized V_H or humanized V_{HH}. In one embodiment, it is a V_{HH}, including a camelized V_H or humanized V_{HH}. Heavy chain ISVs can be derived from a conventional four-chain antibody or from a heavy chain antibody.

[0007] For example, the immunoglobulin single variable domain may be a (single) domain antibody (or an amino acid sequence that is suitable for use as a single domain antibody), a "dAb" or dAb (or an amino acid sequence that is suitable for use as a dAb) or an® ISV (as defined herein and including but not limited to a V_{HH}); other single variable domains, or any suitable fragment of any one thereof.

[0008] In particular, the immunoglobulin single variable domain may be a Nanobody® ISV (such as a V_{HH}, including a humanized V_{HH} or camelized V_H) or a suitable fragment thereof.

[0009] "V_{HH} domains", also known as V_{HH}s, V_{HH} antibody fragments and V_{HH} immunoglobulins, have originally been described as the antigen binding immunoglobulin variable domain of "heavy chain antibodies" (i.e., of "antibodies devoid of light chains"; Hamers-Casterman et al. 1993 (Nature 363: 446-448). The term "V_{HH} domain" has been chosen in order to distinguish these variable domains from the heavy chain variable domains that are present in conventional 4-chain antibodies (which are referred to herein as "V_H domains") and from the light chain variable domains that are present in conventional 4-chain antibodies (which are referred to herein as "V_L domains"). For a further description of V_{HH}'s, reference is made to the review article by Muyldermans 2001 (Reviews in Molecular Biotechnology 74: 277-302).

[0010] For the term "dAb's" and "domain antibody", reference is for example made to Ward et al. 1989 (Nature 341: 544), to Holt et al. 2003 (Trends Biotechnol. 21: 484); as well as to for example WO 2004/068820, WO 2006/030220, WO 2006/003388 and other published patent applications of Domantis Ltd. It should also be noted that, although less preferred in the context of the present invention because they are not of mammalian origin, single variable domains can be derived from certain species of shark (for example, the so-called "IgNAR domains", see for example WO 2005/18629).

[0011] Immunoglobulin sequences of different origin, comprising mouse, rat, rabbit, donkey, human and camelid immunoglobulin sequences can be used herein. Also, fully human, humanized or chimeric sequences can be used in the method described herein. For example,

camelid immunoglobulin sequences and humanized camelid immunoglobulin sequences, or camelized domain antibodies, e.g. camelized dAb as described by Ward et al. 1989 (Nature 341: 544), WO 1994/04678, and Davis and Riechmann (1994, Febs Lett., 339:285-290; and 1996, Prot. Eng., 9:531-537) can be used herein. Moreover, the ISVs are fused forming a multivalent and/or multispecific construct (for multivalent and multispecific polypeptides containing one or more V_{HH} domains and their preparation, reference is also made to Conrath et al. 2001 (J. Biol. Chem., Vol. 276, 10. 7346-7350) as well as to for example WO 1996/34103 and WO 1999/23221).

[0012] A "humanized V_{HH}" comprises an amino acid sequence that corresponds to the amino acid sequence of a naturally occurring V_{HH} domain, but that has been "humanized", i.e. by replacing one or more amino acid residues in the amino acid sequence of said naturally occurring V_{HH} sequence (and in particular in the framework sequences) by one or more of the amino acid residues that occur at the corresponding position(s) in a V_H domain from a conventional 4-chain antibody from a human being (e.g. indicated above). This can be performed in a manner known per se, which will be clear to the skilled person, for example on the basis of the prior art (e.g. WO 2008/020079). Again, it should be noted that such humanized V_{HH}s can be obtained in any suitable manner known per se and thus are not strictly limited to polypeptides that have been obtained using a polypeptide that comprises a naturally occurring VHH domain as a starting material.

[0013] A "camelized V_H" comprises an amino acid sequence that corresponds to the amino acid sequence of a naturally occurring V_H domain, but that has been "camelized", i.e. by replacing one or more amino acid residues in the amino acid sequence of a naturally occurring V_H domain from a conventional 4-chain antibody by one or more of the amino acid residues that occur at the corresponding position(s) in a V_{HH} domain of a (camelid) heavy chain antibody. This can be performed in a manner known per se, which will be clear to the skilled person, for example on the basis of the description in the prior art (e.g. Davies and Riechman 1994, FEBS 339: 285; 1995, Biotechnol. 13: 475; 1996, Prot. Eng. 9: 531; and Riechman 1999, J. Immunol. Methods 231: 25). Such "camelizing" substitutions are inserted at amino acid positions that form and/or are present at the V_H-V_L interface, and/or at the so-called Camelidae hallmark residues, as defined herein (see for example WO 1994/04678 and Davies and Riechmann (1994 and 1996, supra). In one embodiment, the V_H sequence that is used as a starting material or starting point for generating or designing the camelized V_H is a V_H sequence from a mammal, such as the V_H sequence of a human being, such as a V_H3 sequence. However, it should be noted that such

camelized V_H can be obtained in any suitable manner known per se and thus are not strictly limited to polypeptides that have been obtained using a polypeptide that comprises a naturally occurring V_H domain as a starting material.

[0014] A machine-learning model is a computational model that learns patterns and relationships in data, and then uses that knowledge to make predictions or decisions on new data. Neural networks are machine learning models that employ one or more layers of nonlinear units to predict an output for a received input. The output of each hidden layer is used as input to the next layer in the network, i.e., the next hidden layer or the output layer. Each layer of the network generates an output from a received input in accordance with current values of a respective set of parameters.

SUMMARY

[0015] This disclosure describes methods, computer systems, and apparatus, including computer programs encoded on computer storage media, for predicting thermal stabilities of immunoglobulin single variable domains (ISVs).

[0016] In one aspect, this disclosure provides a prediction method for predicting the thermal stability of an ISV. The method can be implemented by a system including one or more computers. The system obtains data representing an amino acid sequence of the ISV, generates an input token vector by numerically encoding the amino acid sequence, and generates an embedded feature vector by processing the input token vector using an embedding machine-learning model having a first set of model parameters. The first set of model parameters have been updated using self-supervised learning of a first machine-learning model that includes the embedding machine-learning model and is configured to perform a sequence reconstruction task. The system processes an input including the embedded feature vector using a prediction machine-learning model to generate an output that predicts a thermal stability measure of the ISV. The prediction machine-learning model has a second set of model parameters that have been updated using supervised learning, based on a plurality of training examples, of a second machine-learning model including the prediction machine-learning model. Each respective training example includes (i) a respective training input specifying a representation of a respective ISV and (ii) a respective label specifying a thermal stability measure of the respective ISV.

[0017] In some implementations, the self-supervised learning includes training the first machine-learning model on a first dataset including sequence representations of a set of example ISVs. In some cases, the set of example ISVs include a plurality of heavy chain single variable domains (VHs). In some cases, the set of example ISVs include a plurality of camelized VHs. In some cases, the set of example ISVs include a plurality VHHs. In some cases, the set of example ISVs include a plurality of humanized VHHs.

[0018] In some implementations, the self-supervised learning includes training the first machine-learning model on one or more of: a reconstruction task, a token unmasking task, or a next token prediction task.

[0019] In some implementations, the thermal stability measure is a melting temperature.

[0020] In some implementations, the input to the prediction machine-learning model further includes data representing the amino acid sequence of the ISV.

[0021] In some implementations, the input further includes data characterizing a sequence length of ISV.

[0022] In some implementations, the input further includes data characterizing a three-dimensional (3D) structure of the ISV.

[0023] In some implementations, the input processed by the prediction machine-learning model further includes data characterizing a germline of the ISV. For example, the input can further include data characterizing mutations of the ISV from a corresponding wildtype ISV that the ISV is mutated from.

[0024] In some implementations, the input further includes a second embedded feature vector different from the embedded feature vector generated by the embedding machine-learning model.

[0025] In some implementations, generating the input token vector includes: mapping each amino acid of the amino acid sequence to a respective numerical value; and generating the input token vector by concatenating the numerical values.

[0026] In some implementations, the embedding machine-learning model includes a large language model (LLM). In some cases, the embedding machine-learning model includes: a variational autoencoder (VAE). In some cases, the embedding machine-learning model includes an autoregressive transformer. In some cases, the embedding machine-learning model includes a bidirectional transformer.

[0027] In some implementations, the prediction machine-learning model includes a regression model. In some cases, the regression model is a ridge regression model. In some cases, the regression model is a lasso regression model. In some cases, the regression model is implemented by one or more of: a neural network, a K-nearest neighbors model, a support vector machine, a decision trees model, or a random forest model.

[0028] In some implementations, the first set of model parameters are fixed after the self-supervised learning process and during the supervised learning process.

[0029] In some implementations, the first set of model parameters are further updated during the supervised learning process wherein the embedding machine-learning model and the prediction machine-learning model are jointly trained end-to-end.

[0030] In some implementations, the system further performs operations for selecting an ISV from a set of candidate ISVs. The operations include: predicting a respective thermal stability measure of each of the candidate ISVs using the method described above, and selecting the ISV from the set of candidate ISVs based on the predicted thermal stability measure.

[0031] In another aspect, this disclosure provides a training method for training a prediction model for predicting the thermal stabilities of ISVs. The method can be implemented by a system including one or more computers. The prediction model includes (i) an embedding machine-learning model configured to generate an embedded feature vector for a model input representing an amino acid sequence of the ISV and (ii) a prediction machine-learning model configured to process an input including the embedded feature vector to generate an output specifying one or more properties of the ISV. The system obtains a first dataset including a set of sequence representations of ISVs, performs self-supervised learning of a first machine-learning model including the embedding machine-learning model on a reconstruction task using the first data set, and obtains a second dataset including a plurality of training examples. Each respective training example includes (i) a respective training input specifying a representation of a respective ISV and (ii) a respective label specifying a respective thermal stability measure for the respective ISV. The system performs supervised learning of a second machine-learning model including the prediction machine-learning on the second dataset.

[0032] In some implementations, the system further finetunes the first machine-learning model on amino acid sequences of a set of humanized VHHs.

[0033] In some implementations, the first machine-learning model includes a large language model (LLM). In some cases, the first machine-learning model includes: a variational autoencoder (VAE), an autoregressive transformer, or a bidirectional transformer.

[0034] In some implementations, the prediction machine-learning model includes one or more of: a neural network, a K-nearest neighbors model, a support vector machine, a decision trees model, a random forest model, a ridge regression model, or a lasso regression model.

[0035] In some implementations, a first set of model parameters of the embedding machine-learning model are fixed after the self-supervised learning process and during the supervised learning process.

[0036] In some implementations, a first set of model parameters of the embedding machine-learning model are further updated during the supervised learning process wherein the embedding machine-learning model and the prediction machine-learning model are jointly trained end-to-end.

[0037] In some implementations, to perform the self-supervised learning of the first machine-learning model, the system initiates values of parameters of the first machine-learning model, and updates the values of the parameters of the first machine-learning model by minimizing a loss function defined for the sequence reconstruction task. In one example, the loss function is defined as

$$\mathcal{L}_{\text{MLM}} = -\mathbb{E}_{x\sim X}\mathbb{E}_M \sum \log p(x_i \mid x_M),$$

where $p(x_i \mid x_M)$ represents a probability of the first machine-learning model predicting that a token x_i is present at a particular masked position, given an unmasked portion x_M of an input sequence x.

[0038] This disclosure also provides a system including one or more computers and one or more storage devices storing instructions that, when executed by the one or more computers, cause the one or more computers to perform the methods described above.

[0039] This disclosure also provides one or more computer storage media storing instructions that when executed by one or more computers, cause the one or more computers to perform the methods described above.

[0040] The subject matter described in this disclosure can be implemented in particular embodiments so as to realize one or more advantages.

[0041] Thermal stability is an important property for immunoglobulin single variable domains (ISVs) due to its impact on their functionality and applicability in various biomedical and biotechnological applications. The stability of ISVs under different temperature conditions is crucial for ensuring their integrity, maintaining binding specificity, and prolonging their shelf life. In applications such as diagnostics, therapeutics, and biotechnology, where ISVs are often employed for targeting specific antigens, the ability of these molecules to withstand temperature variations is essential for storage, transport, and eventual use. Furthermore, thermal stability is particularly vital in manufacturing processes, such as bioprocessing and formulation, where maintaining the structural integrity of ISVs ensures consistent and reliable performance, ultimately contributing to the efficiency and success of downstream applications. Robust thermal stability not only enhances the practical utility of ISVs but also facilitates their broader adoption in diverse fields, emphasizing the importance of this parameter in optimizing the performance and reliability of these unique antibody fragments.

[0042] Thus, an important goal of ISV engineering is to design, create, and/or select ISVs with optimized thermal stability for specific applications. One approach of ISV engineering includes generating a large number of candidate ISVs, measuring the thermal stability of each ISV for the specific application, and selecting the ISV with optimal thermal stability. This process can be iteratively performed by diversifying the selected ISVs to generate the candidate ISVs for the next iteration.

[0043] Existing ISV engineering processes are associated with many challenges, such as multi-objective optimization (i.e., the need to optimize multiple thermal stability for a specific application) and a large search space. For example, for a 14 amino acid CDRH3, the sequence space is as large as 1.6×10^{18}. Experimental benchmarking such a search space becomes unattainable.

[0044] The techniques described in this specification use deep learning to computationally predict ISV thermal stability based on the amino acid sequence of the ISV. In particular, the described techniques use self-supervised learning to pre-train a language model for generating effective embeddings for ISV sequences, followed by supervised learning using labeled data for predicting the thermal stability (e.g., the melting temperature) of the ISV. In particular, the language model is generally pre-trained on ISV sequences, rather than being trained on general protein sequences. This selection of training data is important for ensuring that the embeddings generated by the language model are meaningful for the downstream task of predicting ISV

properties. The self-supervised pre-training process makes it possible to generate high-performance embeddings when labeled data is limited for downstream tasks.

[0045] Based on the predicted ISV thermal stability, the described system or another system can select optimal ISV sequences for specific applications. For example, the system can generate an output that indicates whether a particular ISV is suitable for a particular application, an output that specifies the optimal ISV sequence for the particular application, or an output that indicates where mutations can be made in the sequence. The system can transmit the output to a fabrication apparatus operative to implement the instruction to produce the ISV. Overall, by training high-performance prediction models based on limited experimental data and using the trained model to predict ISV thermal stability, the described techniques can greatly improve the efficacy and efficiency of ISV engineering.

[0046] Predicting protein thermal stability remains a significant challenge despite recent advancements in structural biology, such as the breakthroughs achieved with AlphaFold2 which have significantly increased the number of solved protein structures. Existing approaches have shown limited success in various contexts. For instance, challenges like Kaggle's Novozymes Enzyme Stability Prediction competition highlighted the difficulty of the task, where the highest achievable R-squared values only reached 0.2-0.55. This underscores the inherent complexity of predicting protein stability and the need for innovative solutions.

[0047] The techniques described in this specification leverage a machine learning-based analytical framework that involves several components. Firstly, in some implementations, the embedding model is pre-trained on a carefully curated dataset encompassing a diverse range of ISVs, including, for example, VHs, VHHs, humanized VHHs, and camelized VHs. This departs from the conventional approach for training embedding models for predicting protein properties, where training on a larger, more general dataset like all proteins is often considered advantageous. The choice to limit the training data to ISV sequences is based on the recognition that ISVs share distinct properties and thermostability patterns compared to other proteins. The focused pre-training ensures that the model captures the features specific to ISVs while excluding irrelevant signals from other proteins, which is crucial for accurate thermal stability predictions for ISVs. Secondly, in some implementations, the framework employs a bidirectional encoder representation from transformers (BERT) model as the architecture for the embedding model. The use of BERT, a powerful transformer-based model known for its success in natural language processing, combined with the curated ISV training dataset, provides an effective approach for extracting high-quality features in ISV sequences. Furthermore, a downstream

regression model is trained to predict the melting temperature (Tm) of ISVs, providing a quantitative measure of their thermal stability. These components operate synergistically, producing a model that achieves prediction accuracy significantly surpassing the current state of the art.

[0048] The details of one or more embodiments of the subject matter described in this disclosure are set forth in the accompanying drawings and the description below. Other features, aspects, and advantages of the subject matter will become apparent from the description, the drawings, and the claims.

BRIEF DESCRIPTION OF THE DRAWINGS

[0049] FIG. 1 shows an example environment for screening an immunoglobulin single variable domain (ISV) library using a thermal stability prediction system.

[0050] FIG. 2 shows an example of a thermal stability prediction system.

[0051] FIG. 3 is a flow diagram illustrating an example process for predicting the thermal stability of an ISV.

[0052] FIG. 4 is a flow diagram illustrating an example process for training a prediction model for predicting the thermal stability of ISVs.

[0053] FIG. 5 is a block diagram of an example computer system.

[0054] FIG. 6 shows example results of predicting thermal stability measures of ISV sequences.

[0055] Like reference numbers and designations in the various drawings indicate like elements.

DETAILED DESCRIPTION

[0056] FIG. 1 shows an example environment 100 for screening an immunoglobulin single variable domain (ISV) library 102 using a thermal stability prediction system 200. The ISV library 102 defines a set of ISVs, where each ISV is represented by a respective sequence of amino acids. The ISV library 102 can include any appropriate number of ISVs, e.g., 1 hundred, 1 thousand, 10 thousand, or 1 million ISVs.

[0057] The ISV library 102 can be generated in any of a variety of possible ways. For instance, some or all of the ISVs in the ISV library 102 can be variants of one or more “original” or

"wildtype" ISVs. "Original" or "wildtype" can refer to an ISV which is the starting point of generating one or more ISV variants. More specifically, each ISV in the ISV library 102 can be generated by modifying the identity of a respective amino acid at one or more positions in the amino acid sequence of an original ISV. Positions in the amino acid sequence of the original ISV can be selected for mutation in any appropriate way, e.g., through random selection or through selection in accordance with a predefined rule. The identities of new amino acids substituted into positions in the amino acid sequence of the original ISV can be selected in any appropriate way, e.g., randomly selected from a probability distribution over a set of possible amino acids. The respective amino acid sequence of each ISV in the ISV library can differ from the amino acid sequence of the original ISV in any appropriate number of positions, e.g., 1 position, 3 positions, or 10 positions.

[0058] The thermal stability prediction system 200 is configured to process a sequence of an ISV (e.g., from the ISV library 102) to generate a thermal stability measure 104 for the ISV that characterizes a predicted thermal stability of the corresponding ISV. The thermal stability measure 104 can be represented by, for example, a melting temperature (Tm) of the ISV. The melting temperature (Tm) of an ISV can be defined as the temperature at which the ISV undergoes a transition from a folded to an unfolded state.

[0059] The thermal stability prediction system 200 can screen the ISV library 102 to identify ISVs having desirable thermal stability measures. More specifically, the thermal stability prediction system 200 can predict a respective thermal stability measure 104 for each ISV in the ISV library 102. The thermal stability prediction system 200 can designate a proper subset of the ISVs in the ISV library 102 as being "target" ISVs 106 based at least in part on the predicted thermal stability measures 104.

[0060] The thermal stability prediction system 200 can select a proper subset of the ISVs in the ISV library to be designated as target ISVs in any of variety of possible ways. For instance, the thermal stability prediction system 200 can designate any ISV having a thermal stability measure 104 that satisfies a predefined threshold as being a target ISV. As another example, the thermal stability prediction system 200 can designate a predefined number of ISVs having the highest thermal stability measures 104 as being target ISVs.

[0061] The thermal stability prediction system 200 can designate any appropriate number of ISVs from the ISV library 102 as being target ISVs 106, e.g., 10 ISVs, 100 ISVs, or 1000 ISVs. In some cases, the thermal stability prediction system 200 designates only a small fraction of the

total number of ISVs in the ISV library (e.g., <1%, <0.1%, or less than <0.01% of the total number of ISVs in the ISV library) as being target ISVs.

[0062] In some cases, the above screening process can be performed iteratively. That is, the target ISVs 106 can be used as the "original" ISVs to generate additional variants through mutations to be included in the ISV library 102 for the next iteration. After the iterations have been completed (e.g., when a predefined number of iterations have been performed or when the target ISVs have thermal stability measures satisfying one or more predefined conditions), the target ISVs 106 can then be manufactured 108, i.e., physically generated, using appropriate manufacturing techniques.

[0063] The generated ISVs can be used in any of a variety of applications, e.g., the generated ISVs can be applied as a therapeutic 110 to a subject 112 to achieve a therapeutic effect in the subject. In particular, the generated ISVs can target specific disease-related proteins or cells, such as cells related to cancer, inflammatory disorders, or infectious diseases. For example, the ISVs can be used to interfere with disease pathways by binding to and blocking the activity of specific proteins. This can be particularly useful in conditions where abnormal protein signaling contributes to the pathogenesis of the disease. In another example, the ISVs can be conjugated to therapeutic agents or payloads to create targeted drug delivery systems. This approach allows for the specific delivery of drugs to disease sites, reducing off-target effects and improving the therapeutic index. In another example, the ISVs can modulate the immune system by targeting immune cells or regulating immune responses. They may be designed to enhance or suppress immune functions, depending on the therapeutic goal. In another example, the ISVs can be incorporated into antibody-drug conjugates, where the ISV serves as the antigen-binding domain. This allows for targeted delivery of cytotoxic drugs to cancer cells, enhancing the specificity of treatment. In another example, the ISVs can be designed to bind to and neutralize pathogens, preventing them from infecting host cells.

[0064] FIG. 2 shows an example of a thermal stability prediction system 200. The system 200 is an example of a system implemented as computer programs on one or more computers in one or more locations, in which the systems, components, and techniques described below can be implemented.

[0065] The thermal stability prediction system 200 uses machine-learning models to predict the thermal stability measure 240 of an ISV based on input data 210 specifying the amino acid sequence of the ISV. In general, the machine-learning models include an embedding machine-

learning model 220 having a first set of model parameters 222 and a prediction machine-learning model 230 having a second set of model parameters 232.

[0066] The system 200 includes a sequence tokenizer 225 configured to generate an input token vector as an input to the embedding machine-learning model 220. The sequence tokenizer 225 generates the input token vector by numerically encoding the ISV sequence, i.e., the amino acid sequence of the ISV.

[0067] The embedding machine-learning model 220 is configured to process the input token vector to generate an embedded feature vector 225. The embedded feature vector 225 is a numerical representation of input data that captures the essential information required for one or more tasks. In particular, the embedded feature vector 225 can be a high-dimensional vector of real numbers that captures features of the model input specifying the ISV.

[0068] In some implementations, the embedding machine-learning model 220 is a neural network. The embedding neural network can adopt any appropriate architecture. In particular, the embedding neural network 220 can include at least a portion (e.g., the embedding portion) of a state-of-the-art large language model (LLM).

[0069] For example, in some implementations, the embedding neural network 220 can include the encoder network of a variational autoencoder (VAE). Implementation examples of a VAE are described in "Auto-encoding variational Bayes," Kingma et al., arXiv: 1312.6114, 2013.

[0070] In some implementations, the embedding neural network 220 can include the embedding layers of an autoregressive transformer, e.g., a generative pre-trained transformer (GPT). Implementation examples of the GPT are described in "Language Models are Few-Shot Learners," Brown et al., Advances in Neural Information Processing Systems 33: 1877–1901, 2020.

[0071] In some implementations, the embedding neural network 220 can include a bidirectional transformer, e.g., a bidirectional encoder representations from transformers (BERT) model. Implementation examples of the BERT are described in "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding," Devlin et al., arXiv:1810.04805, 2018.

[0072] Using a language model to generate embeddings for an ISV sequence provides several advantages for predicting ISV thermal stability in the downstream task. Due to evolutionary pressures, ISV sequences are not random. For example, an ISV can include evolutionarily conserved regions and portions of the ISV sequence can be reused. Further, correlations and interactions can exist between pairs of positions in an ISV. Language models can learn complex

patterns in ISV sequences, and can be used to identify previously unknown patterns and correlations in ISV sequences. As will be described in more detail below, the language models for generating the ISV embeddings are trained specifically using ISV sequence data. This is important for obtaining a high-performance model for ISV property prediction, that is, for obtaining a model having high prediction accuracy and avoiding model bias or overfitting.

[0073] The system 200 further includes a prediction machine-learning model 230 configured to process an input including the embedded feature vector 225 to generate an output 240 that predicts a thermal stability measure 240 of the ISV. The predicted thermal stability can be a melting temperature (Tm) of the ISV.

[0074] In general, the prediction machine-learning model 230 can be a regression model that processes an input to output a value for the thermal stability measure 240, e.g., the melting temperature (Tm). The input to the prediction machine-learning model 230 includes the embedded feature vector 225 generated by the embedding model 220. In some cases, the input to the prediction machine-learning model 230 can include additional data to characterize the ISV sequence. For example, in some implementations, the input can further include data representing the amino acid sequence of the ISV, e.g., a one-hot encoding of the ISV sequence. In some implementations, the input can further include data characterizing the ISV sequence length. In some implementations, the input can further include data (e.g., molecular properties) derived from a three-dimensional (3D) structure of the ISV. The 3D structure of the ISV can be obtained experimentally or using a prediction model. In some implementations, the ISV is a variant of a wildtype ISV, and the input can further include data characterizing a germline of the ISV, i.e., the identity of the wildtype ISV that the variant ISV was mutated from. In some implementations, the input further comprises data characterizing the mutations of the variant ISV. In some implementations, the input further includes a second embedded feature vector different from the embedded feature vector 225. The second embedded feature vector can be generated, for example, by processing the ISV sequence using a general protein language model.

[0075] The prediction machine-learning model 230 can be implemented with any suitable machine-learning techniques, and can include one or more of: a neural network, a K-nearest neighbors model, a support vector machine, a decision trees model, a random forest model, a ridge regression model, or a lasso regression model.

[0076] In some implementations, the system 200 or another system includes a self-supervised learning engine 250 configured to update the model parameters 222 of the embedding machine-learning model using self-supervised learning, based on a set of ISV sequence representations

255. The goal of the self-supervised learning is to learn meaningful embeddings of ISV sequences without needing to use labeled data. The self-supervised learning engine 250 learns the embeddings using unlabeled ISV sequence data, that is, data specifying or representing each of a set of ISV sequences without ISV property labels. That is, the self-supervised learning engine 250 can leverage the large number of known ISV sequences to learn the embeddings without needing to obtain a large amount of experimental benchmark data for the thermal stability of the known ISVs. In general, the dataset 225 includes a large number of ISV sequences, e.g., hundreds of thousands of ISV sequences, millions of ISV sequences, tens of millions of ISV sequences, or hundreds of millions of ISV sequences. The ISV sequences included in the dataset 225 can include sequences of a variety of ISVs and ISV variants, e.g., VHH variants, VH variants, and humanized VHH variants.

[0077] The choice in limiting the training data to ISV sequences departs from conventional approaches for training embedding models for predicting protein properties, where training on a larger, more general dataset like all proteins is often considered advantageous. This choice is based on the recognition that ISVs share distinct properties and thermostability patterns compared to other proteins. The ISV data focused pre-training ensures that the model captures the features specific to ISVs while excluding irrelevant signals from other proteins, which is crucial for the downstream task of accurately predicting thermal stabilities for ISVs.

[0078] In order to effectively learn the embeddings from unlabeled data, the self-supervised learning engine 250 can be configured to train a first machine-learning model to perform a reconstruction task, that is, a task for generating embeddings for an input ISV sequence representation, and reconstructing the input ISV sequence representation from the embeddings. The first machine-learning model includes the embedding machine-learning model 220 as a sub-network for generating the embeddings. In some cases, the self-supervised learning engine 250 can train the first machine-learning model to perform a token unmasking task and/or a next token prediction task. The unmasking task is a task of predicting masked tokens from the unmasked tokens in an input token sequence. The next token prediction task is a task of predicting the next token in a token sequence, based on the preceding tokens.

[0079] Training the embedding machine-learning model 220 using self-supervised learning techniques, e.g., to perform tasks such as reconstruction, token unmasking, and/or next token prediction, can cause the embedding machine-learning model 220 to learn to generate embeddings encoded with rich information characterizing the ISV sequences.

[0080] As discussed above, the embedding machine-learning model 220 can be a neural network having an appropriate architecture. In general, the neural network is a deep neural network (DNN) having a plurality of hidden layers. Each hidden layer is assocaitated with a activation function, e.g., a ReLU, a Sigmoid, a Tanh, a leaky ReLU, a Gaussian Error Linear Unit (GELU), or a Softmax activation function. Also as discussed above, the training dataset 225 is selected to include a large number of ISV sequences. In a particular illustrative example, the neural network can include a Transformer encoder having encoder layers and a pooling layer with a dimension of 768 and a feed-forward layer with a dimension of 3072. The Transformer encoder can include 12 hidden layers. Each attention layer of the Transformer encoder can have 12 attention heads.

[0081] After the architecture of the neural network have been appropriately selected, the values of the parameters of the first machine-learning model can be initiated (e.g., randomly). The self-supervised learning engine 250 can update the parameters of the first machine-learning model (including the model parameters 222 of the embedding machine-learning model 220) by minimizing a loss function (e.g., the reconstruction error between the input ISV sequence and the reconstructed ISV sequence in the reconstruction task) computed using the training data.

[0082] In one example, the loss function includes a masked language modeling loss for the unmasking task of predicting masked tokens from the unmasked tokens in input token sequences x's of the ISV sequences in the training data. The masked language modeling loss can be defined as

$$\mathcal{L}_{\text{MLM}} = -\mathbb{E}_{x\sim X}\mathbb{E}_{M}\sum \log p(x_i \mid x_M),$$

where $p(x_i \mid x_M)$ represents the probability of the model predicting that the token x_i is present at a particular masked position, given the unmasked portion x_M of the input ISV sequence x.

[0083] The self-supervised learning engine 250 can update the model parameters using any appropriate backpropagation-based machine learning technique, e.g., using the Adam or AdaGrad optimizers.

[0084] The system 200 or another system can further include a supervised learning engine 260 configured to update the model parameters 232 of the prediction model 230 based on a labeled dataset 265. The labeled dataset 265 includes a plurality of labeled training examples. Each training example includes (i) a training input specifying a representation of a respective ISV and (ii) a label specifying the thermal stability measure of the respective ISV. The ISV labels can

obtained be based on experimental measurements of the thermal stability of the corresponding ISVs. For example, techniques such as circular dichroism (CD) spectroscopy, differential scanning fluorimetry (DSF), nano-DSF, or differential scanning calorimetry (DSC) can be used to measure the melting temperature (Tm) of the respective ISV. In general, the labeled dataset 265 includes a much fewer number of training sequences compared to the unlabeled dataset 255. In an illustrative example, while the unlabeled dataset 255 includes millions, tens of millions, or hundreds of millions of ISV sequences, the labeled dataset 265 may include thousands, tens of thousands, or hundreds of thousands of labeled training examples.

[0085] The supervised learning engine 260 is configured to perform supervised learning of a second machine-learning model including the prediction machine-learning model 230 on the labeled dataset 265. That is, the supervised learning engine 260 is configured to update the parameters of the second machine-learning model (including the model parameters 232 of the prediction machine-learning model 230) based on the labeled dataset 265.

[0086] In some implementations, the second machine-learning model further includes the embedding machine-learning model 220. That is, the model parameters 222 of the embedding machine-learning model 220 are further fine-tuned end-to-end with the prediction machine-learning model 230 via supervised learning based on the labeled dataset 265.

[0087] In some other implementations, the model parameters 222 of the embedding machine-learning model 220 are fixed during the supervised learning when the model parameters 232 of the prediction machine-learning model 230 are being updated.

[0088] The supervised learning engine 260 can update the parameters of the second machine-learning model (including model parameters 232 of the prediction machine-learning model 230, and optionally including the model parameters 222 of the embedding machine-learning model 220) by minimizing a prediction error between the predicted ISV thermal stability measure (e.g., Tm) and the thermal stability measure specified in the labels. The supervised learning engine 260 can update the model parameters using any appropriate backpropagation-based machine learning technique, e.g., using the Adam or AdaGrad optimizers.

[0089] Based on the predicted ISV thermal stability measure 240, the system 200 can select ISV sequences with desirable thermal stabilities from candidate ISV sequences for a specific application. For example, the system 200 can generate an output that indicates whether a particular ISV is suitable for a particular application, or an output that specifies the optimal ISV sequence for the particular application. The system can transmit the output to a fabrication apparatus operative to implement the instruction to produce the ISV.

[0090] FIG. 3 is a flow diagram illustrating an example process 300 for predicting the thermal stability of an ISV. For convenience, the process 300 will be described as being performed by a system of one or more computers located in one or more locations. For example, a thermal stability prediction system, e.g., the thermal stability prediction system 200 of FIG. 2, appropriately programmed in accordance with this disclosure, can perform the process 300.

[0091] At 310, the system obtains an ISV sequence, i.e., a token sequence representing the amino acid sequence of the ISV. The ISV sequence can be a sequence of a candidate ISV that is intended for a particular application. The candidate ISV can be a variant mutated from an "original" or "wildtype" ISV.

[0092] At 320, the system generates an input token vector by numerically encoding the token sequence. For example, the system can generate a vector by concatenating numerical values assigned to different amino acids to form the input token vector.

[0093] At 330, the system generates an embedded feature vector by processing the input token vector using an embedding machine-learning model. The embedding machine-learning model has a first set of model parameters. The first set of model parameters have been updated using self-supervised learning of a first machine-learning model that includes the embedding machine-learning model and configured to perform a sequence reconstruction task. In one particular example, the embedding machine-learning model is a large language model (LLM) based on a bidirectional transformer.

[0094] At 340, the system processes an input including the embedded feature vector using a prediction machine-learning model to generate an output that predicts a thermal stability measure of the input ISV. In some cases, the input can further include additional data that supplements the embedded feature vector generated by the embedding machine-learning model. For example, the additional data can include one or more of: an encoding of the input ISV sequence, a length of the ISV sequence, data characterizing the 3D structure of the ISV, data specifying a germline of the ISV, and/or data characterizing the mutations of the input ISV from the original ISV. In some implementations, the input further includes a second embedded feature vector that has been generated by a general protein language model.

[0095] The prediction machine-learning model has a second set of model parameters. The second set of model parameters have been updated using supervised learning, based on a plurality of training examples, of a second machine-learning model including the prediction machine-learning model.

[0096] FIG. 4 is a flow diagram illustrating an example process 400 for training a prediction model for predicting thermal stability measures of ISVs. For convenience, the process 400 will be described as being performed by a system of one or more computers located in one or more locations. For example, a thermal stability prediction system, e.g., the thermal stability prediction system 200 of FIG. 2, appropriately programmed in accordance with this disclosure, can perform the process 400.

[0097] In general, the prediction model includes (i) an embedding machine-learning model configured to generate an embedded feature vector for a model input representing an amino acid sequence of the ISV and (ii) a prediction machine-learning model configured to process the embedded feature vector to generate an output specifying a thermal stability measure of the ISV.

[0098] At 410, the system obtains a first dataset including a set of sequence representations of ISVs. For example, the sequence representation can be a token vector that numerically encodes the sample sequence of an ISV for which the amino acid sequence is known.

[0099] At 420, the system performs self-supervised learning of a first machine-learning model including the embedding machine-learning model on a reconstruction task using the first data set. The first machine-learning model can be a large language model and can include a variational autoencoder (VAE), an autoregressive transformer, or a bidirectional transformer.

[0100] At 430, the system obtains a second dataset including a plurality of training examples. Each training example includes (i) a respective training input specifying a representation of a respective ISV and (ii) a respective label specifying a thermal stability measure of the respective ISV.

[0101] At 440, the system performs supervised learning of a second machine-learning model including the prediction machine-learning model based on the second dataset. The second machine-learning model can include one or more of: a neural network, a K-nearest neighbors model, a support vector machine, a decision trees model, a random forest model, or a ridge regression model.

[0102] FIG. 5 is a block diagram of an example computer system 500 that can be used to perform operations described above. The system 500 includes a processor 510, a memory 520, a storage device 530, and an input/output device 540. Each of the components 510, 520, 530, and 540 can be interconnected, for example, using a system bus 550. The processor 510 is capable of processing instructions for execution within the system 500. In one implementation, the processor 510 is a single-threaded processor. In another implementation, the processor 510 is a

multi-threaded processor. The processor 510 is capable of processing instructions stored in the memory 520 or on the storage device 530.

[0103] The memory 520 stores information within the system 500. In one implementation, the memory 520 is a computer-readable medium. In one implementation, the memory 520 is a volatile memory unit. In another implementation, the memory 520 is a non-volatile memory unit.

[0104] The storage device 530 is capable of providing mass storage for the system 500. In one implementation, the storage device 530 is a computer-readable medium. In various different implementations, the storage device 530 can include, for example, a hard disk device, an optical disk device, a storage device that is shared over a network by multiple computing devices (for example, a cloud storage device), or some other large capacity storage device.

[0105] The input/output device 540 provides input/output operations for the system 500. In one implementation, the input/output device 540 can include one or more network interface devices, for example, an Ethernet card, a serial communication device, for example, a RS-232 port, and/or a wireless interface device, for example, a 502.11 card. In another implementation, the input/output device can include driver devices configured to receive data and send output data to other input/output devices, for example, keyboard, printer and display devices 560. Other implementations, however, can also be used, such as mobile computing devices, mobile communication devices, set-top box television client devices, etc.

[0106] Although an example processing system has been described in FIG. 5, implementations of the subject matter and the functional operations described in this disclosure can be implemented in other types of digital electronic circuitry, or in computer software, firmware, or hardware, including the structures disclosed in this disclosure and their structural equivalents, or in combinations of one or more of them.

[0107] FIG. 6 shows melting temperatures (Tm's) predicted using the techniques described above for two sets of ISV sequences compared with measured Tm's. In particular, panel (a) shows predicted Tm's vs. measured Tm's for 102 wildtype ISVs. The correlation coefficient r is calculated as 0.6 for the wildtype ISVs. Panel (b) shows predicted Tm vs. measured Tm for 454 wild ISV variants. The correlation coefficient r is calculated as 0.74 for the wildtype ISVs. FIG. 6 demonstrates the efficacy of using the machine-learning model described in this specification for predicting the thermal stability of ISV sequences.

[0108] This disclosure uses the term "configured" in connection with systems and computer program components. For a system of one or more computers to be configured to perform particular operations or actions means that the system has installed on it software, firmware, hardware, or a combination of them that in operation cause the system to perform the operations or actions. For one or more computer programs to be configured to perform particular operations or actions means that the one or more programs include instructions that, when executed by data processing apparatus, cause the apparatus to perform the operations or actions. Embodiments of the subject matter and the functional operations described in this disclosure can be implemented in digital electronic circuitry, in tangibly-embodied computer software or firmware, in computer hardware, including the structures disclosed in this disclosure and their structural equivalents, or in combinations of one or more of them. Embodiments of the subject matter described in this disclosure can be implemented as one or more computer programs, i.e., one or more modules of computer program instructions encoded on a tangible non transitory storage medium for execution by, or to control the operation of, data processing apparatus. The computer storage medium can be a machine-readable storage device, a machine-readable storage substrate, a random or serial access memory device, or a combination of one or more of them. Alternatively or in addition, the program instructions can be encoded on an artificially generated propagated signal, e.g., a machine-generated electrical, optical, or electromagnetic signal, that is generated to encode information for transmission to suitable receiver apparatus for execution by a data processing apparatus.

[0109] The term "data processing apparatus" refers to data processing hardware and encompasses all kinds of apparatus, devices, and machines for processing data, including by way of example a programmable processor, a computer, or multiple processors or computers. The apparatus can also be, or further include, special purpose logic circuitry, e.g., an FPGA (field programmable gate array) or an ASIC (application specific integrated circuit). The apparatus can optionally include, in addition to hardware, code that creates an execution environment for computer programs, e.g., code that constitutes processor firmware, a protocol stack, a database management system, an operating system, or a combination of one or more of them.

[0110] A computer program, which may also be referred to or described as a program, software, a software application, an app, a module, a software module, a script, or code, can be written in any form of programming language, including compiled or interpreted languages, or declarative or procedural languages; and it can be deployed in any form, including as a stand-alone program or as a module, component, subroutine, or other unit suitable for use in a

computing environment. A program may, but need not, correspond to a file in a file system. A program can be stored in a portion of a file that holds other programs or data, e.g., one or more scripts stored in a markup language document, in a single file dedicated to the program in question, or in multiple coordinated files, e.g., files that store one or more modules, sub programs, or portions of code. A computer program can be deployed to be executed on one computer or on multiple computers that are located at one site or distributed across multiple sites and interconnected by a data communication network.

[0111] In this disclosure, the term "database" is used broadly to refer to any collection of data: the data does not need to be structured in any particular way, or structured at all, and it can be stored on storage devices in one or more locations. Thus, for example, the index database can include multiple collections of data, each of which may be organized and accessed differently.

[0112] Similarly, in this disclosure the term "engine" is used broadly to refer to a software-based system, subsystem, or process that is programmed to perform one or more specific functions. Generally, an engine will be implemented as one or more software modules or components, installed on one or more computers in one or more locations. In some cases, one or more computers will be dedicated to a particular engine; in other cases, multiple engines can be installed and running on the same computer or computers.

[0113] The processes and logic flows described in this disclosure can be performed by one or more programmable computers executing one or more computer programs to perform functions by operating on input data and generating output. The processes and logic flows can also be performed by special purpose logic circuitry, e.g., an FPGA or an ASIC, or by a combination of special purpose logic circuitry and one or more programmed computers.

[0114] Computers suitable for the execution of a computer program can be based on general or special purpose microprocessors or both, or any other kind of central processing unit. Generally, a central processing unit will receive instructions and data from a read only memory or a random access memory or both. The essential elements of a computer are a central processing unit for performing or executing instructions and one or more memory devices for storing instructions and data. The central processing unit and the memory can be supplemented by, or incorporated in, special purpose logic circuitry. Generally, a computer will also include, or be operatively coupled to receive data from or transfer data to, or both, one or more mass storage devices for storing data, e.g., magnetic, magneto optical disks, or optical disks. However, a computer need not have such devices. Moreover, a computer can be embedded in another device, e.g., a mobile telephone, a personal digital assistant (PDA), a mobile audio or video player, a game console, a

Global Positioning System (GPS) receiver, or a portable storage device, e.g., a universal serial bus (USB) flash drive, to name just a few.

[0115] Computer readable media suitable for storing computer program instructions and data include all forms of non volatile memory, media and memory devices, including by way of example semiconductor memory devices, e.g., EPROM, EEPROM, and flash memory devices; magnetic disks, e.g., internal hard disks or removable disks; magneto optical disks; and CD ROM and DVD-ROM disks.

[0116] To provide for interaction with a user, embodiments of the subject matter described in this disclosure can be implemented on a computer having a display device, e.g., a CRT (cathode ray tube) or LCD (liquid crystal display) monitor, for displaying information to the user and a keyboard and a pointing device, e.g., a mouse or a trackball, by which the user can provide input to the computer. Other kinds of devices can be used to provide for interaction with a user as well; for example, feedback provided to the user can be any form of sensory feedback, e.g., visual feedback, auditory feedback, or tactile feedback; and input from the user can be received in any form, including acoustic, speech, or tactile input. In addition, a computer can interact with a user by sending documents to and receiving documents from a device that is used by the user; for example, by sending web pages to a web browser on a user's device in response to requests received from the web browser. Also, a computer can interact with a user by sending text messages or other forms of message to a personal device, e.g., a smartphone that is running a messaging application, and receiving responsive messages from the user in return.

[0117] Data processing apparatus for implementing machine learning models can also include, for example, special-purpose hardware accelerator units for processing common and compute-intensive parts of machine learning training or production, i.e., inference, workloads.

[0118] Machine learning models can be implemented and deployed using a machine learning framework, .e.g., the PyTorch, Scikit-learn, Keras, or TensorFlow framework.

[0119] Embodiments of the subject matter described in this disclosure can be implemented in a computing system that includes a back end component, e.g., as a data server, or that includes a middleware component, e.g., an application server, or that includes a front end component, e.g., a client computer having a graphical user interface, a web browser, or an app through which a user can interact with an implementation of the subject matter described in this disclosure, or any combination of one or more such back end, middleware, or front end components. The components of the system can be interconnected by any form or medium of digital data

communication, e.g., a communication network. Examples of communication networks include a local area network (LAN) and a wide area network (WAN), e.g., the Internet.

[0120] The computing system can include clients and servers. A client and server are generally remote from each other and typically interact through a communication network. The relationship of client and server arises by virtue of computer programs running on the respective computers and having a client-server relationship to each other. In some embodiments, a server transmits data, e.g., an HTML page, to a user device, e.g., for purposes of displaying data to and receiving user input from a user interacting with the device, which acts as a client. Data generated at the user device, e.g., a result of the user interaction, can be received at the server from the device.

[0121] While this disclosure contains many specific implementation details, these should not be construed as limitations on the scope of any invention or on the scope of what may be claimed, but rather as descriptions of features that may be specific to particular embodiments of particular inventions. Certain features that are described in this disclosure in the context of separate embodiments can also be implemented in combination in a single embodiment. Conversely, various features that are described in the context of a single embodiment can also be implemented in multiple embodiments separately or in any suitable subcombination. Moreover, although features may be described above as acting in certain combinations and even initially be claimed as such, one or more features from a claimed combination can in some cases be excised from the combination, and the claimed combination may be directed to a subcombination or variation of a subcombination.

[0122] Similarly, while operations are depicted in the drawings and recited in the claims in a particular order, this should not be understood as requiring that such operations be performed in the particular order shown or in sequential order, or that all illustrated operations be performed, to achieve desirable results. In certain circumstances, multitasking and parallel processing may be advantageous. Moreover, the separation of various system modules and components in the embodiments described above should not be understood as requiring such separation in all embodiments, and it should be understood that the described program components and systems can generally be integrated together in a single software product or packaged into multiple software products.

[0123] Particular embodiments of the subject matter have been described. Other embodiments are within the scope of the following claims. For example, the actions recited in the claims can be performed in a different order and still achieve desirable results. As one example, the

processes depicted in the accompanying figures do not necessarily require the particular order shown, or sequential order, to achieve desirable results. In some cases, multitasking and parallel processing may be advantageous.

5

CLAIMS

1. A computer-implemented method for predicting a thermal stability of an immunoglobulin single variable domain (ISV), the method comprising:

obtaining data representing an amino acid sequence of the ISV;

generating an input token vector by numerically encoding the amino acid sequence;

generating an embedded feature vector by processing the input token vector using an embedding machine-learning model having a first set of model parameters, wherein the first set of model parameters have been updated using self-supervised learning of a first machine-learning model that comprises the embedding machine-learning model and is configured to perform a sequence reconstruction task; and

processing an input comprising the embedded feature vector using a prediction machine-learning model to generate an output that predicts a thermal stability measure of the ISV, wherein the prediction machine-learning model has a second set of model parameters that have been updated using supervised learning, based on a plurality of training examples, of a second machine-learning model comprising the prediction machine-learning model, each respective training example comprising (i) a respective training input specifying a representation of a respective ISV and (ii) a respective label specifying a thermal stability measure of the respective ISV.

2. The method of claim 1, wherein the self-supervised learning comprises training the first machine-learning model on a first dataset comprising sequence representations of a set of example ISVs.

3. The method of claim 2, wherein the set of example ISVs include a plurality of heavy chain single variable domains (VHs).

4. The method of any of the preceding claims, wherein the thermal stability measure is a melting temperature.

5. The method of any of the preceding claims, wherein the input to the prediction machine-learning model further comprises data representing the amino acid sequence of the ISV.

6. The method of any of the preceding claims, wherein the input further comprises data characterizing a sequence length of ISV.

7. The method of any of the preceding claims, wherein the input further comprises data characterizing a three-dimensional (3D) structure of the ISV.

8. The method of any of the preceding claims, wherein the input processed by the prediction machine-learning model further comprises data characterizing a germline of the ISV.

9. The method of claim 12, wherein the input further comprises data characterizing mutations of the ISV from a corresponding wildtype ISV that the ISV is mutated from.

10. The method of any of the preceding claims, wherein the embedding machine-learning model comprises a bidirectional transformer.

11. The method of any preceding claim, wherein performing the self-supervised learning of the first machine-learning model comprises:

initiating values of parameters of the first machine-learning model; and

updating the values of the parameters of the first machine-learning model by minimizing a loss function defined for the sequence reconstruction task.

12. The method of claim 11, wherein the loss function is defined as

$$\mathcal{L}_{\text{MLM}} = -\mathbb{E}_{x\sim X}\mathbb{E}_{M}\sum \log p(x_i \mid x_M),$$

where $p(x_i \mid x_M)$ represents a probability of the first machine-learning model predicting that a token x_i is present at a particular masked position, given an unmasked portion x_M of an input sequence x.

13. A method for selecting an ISV from a set of candidate ISVs, the method comprising:

predicting a respective thermal stability measure of each of the candidate ISVs using the method according to the method of any one of the preceding claims; and

selecting the ISV from the set of candidate ISVs based on the predicted thermal stability measure.

14. A system comprising:

one or more computers; and

one or more storage devices storing instructions that when executed by the one or more computers, cause the one or more computers to perform operations of the respective method of any one of claims 1-13.

15. One or more computer-readable storage media storing instructions that, when executed by one or more computers, cause the one or more computers to perform operations of the respective method of any one of claims 1-13.

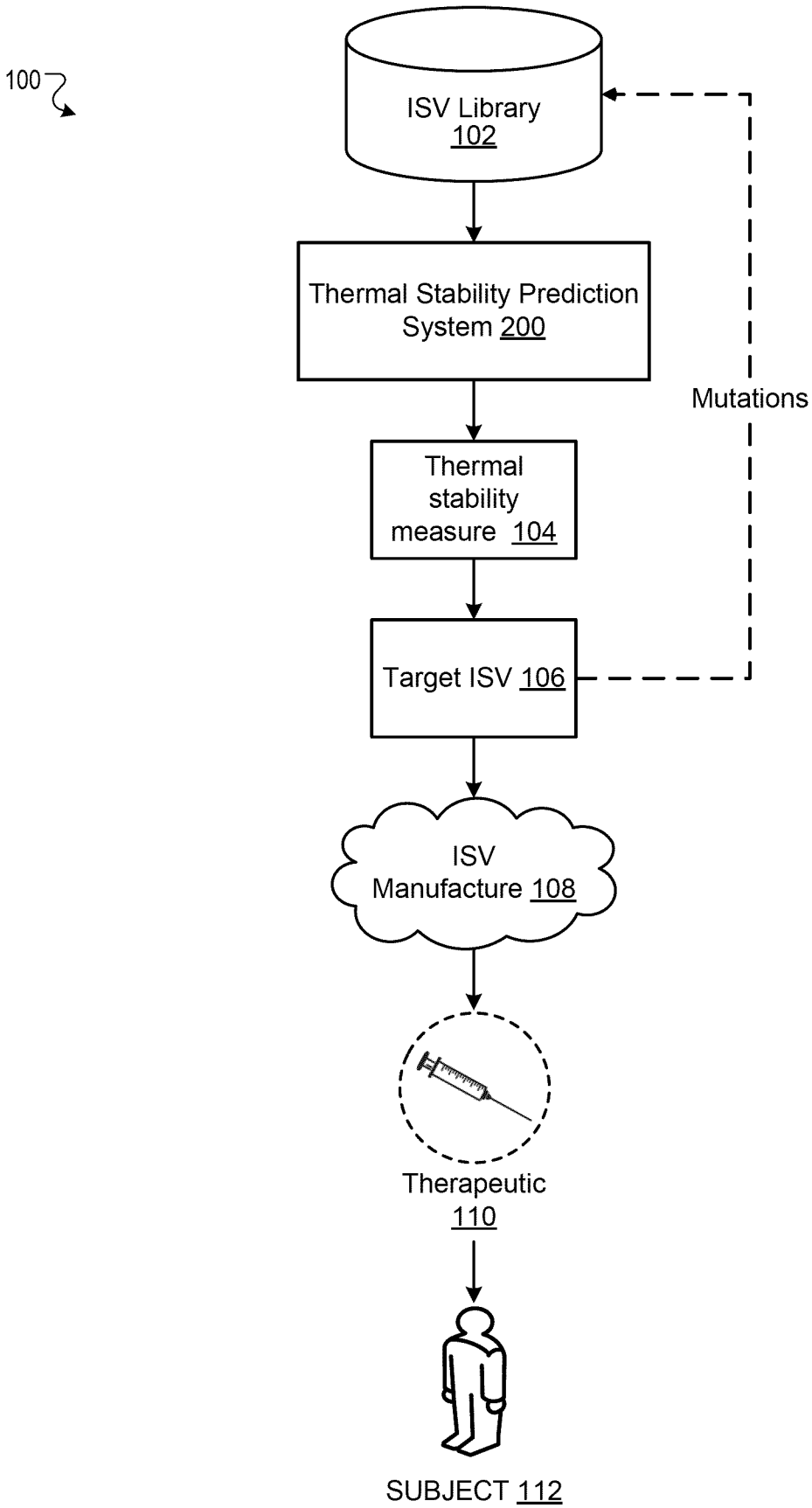

100
ISV Library
102
Thermal Stability Prediction
System 200
Mutations
Thermal
stability
measure 104
Target ISV 106
ISV
Manufacture 108
Therapeutic
110
SUBJECT 112

FIG. 1

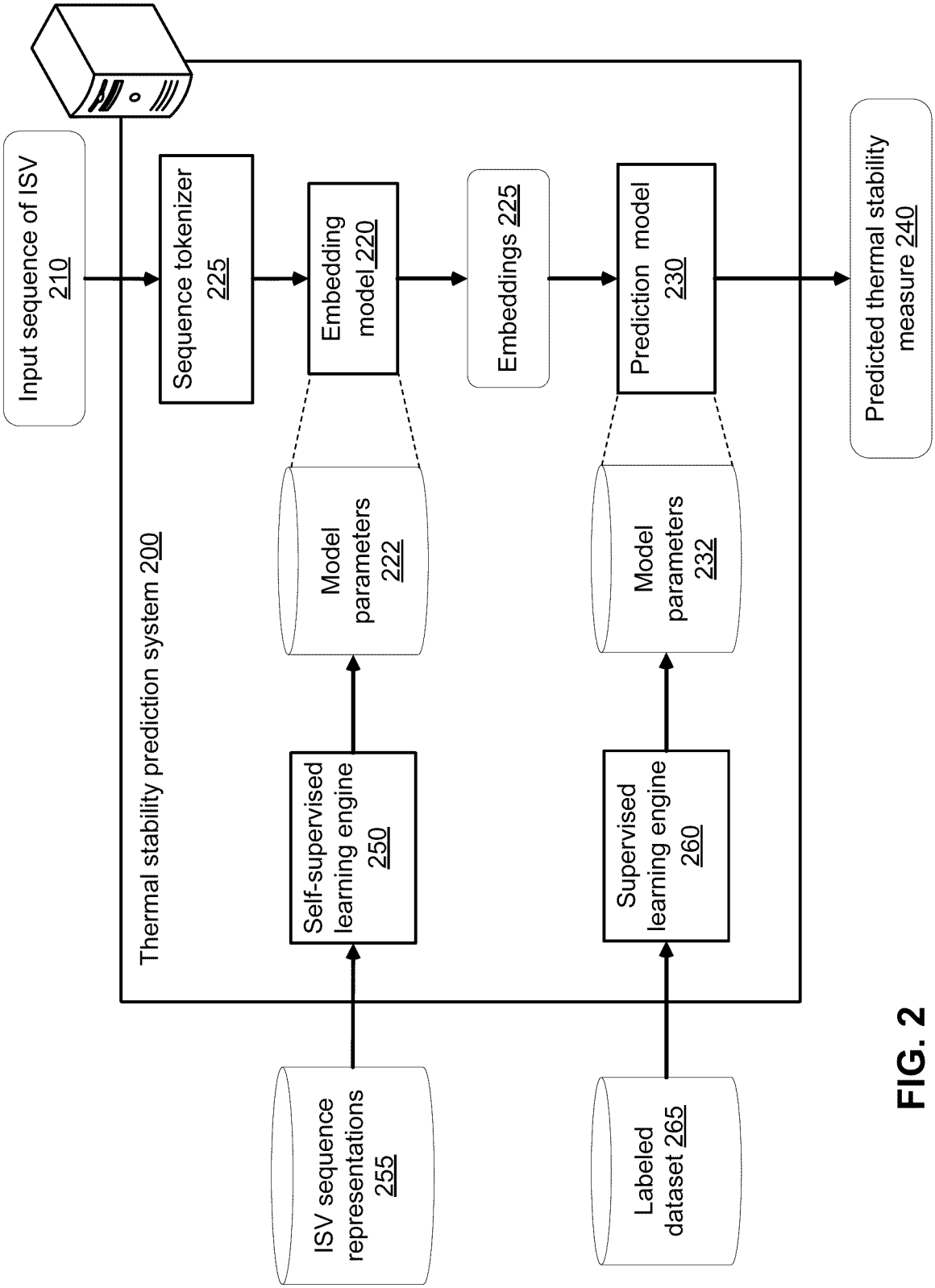

Input sequence of ISV 210
Sequence tokenizer 225
Embedding model 220
Embeddings 225
Prediction model 230
Predicted thermal stability measure 240
Thermal stability prediction system 200
Model parameters 222
Model parameters 232
Self-supervised learning engine 250
Supervised learning engine 260
ISV sequence representations 255
Labeled dataset 265

FIG. 2

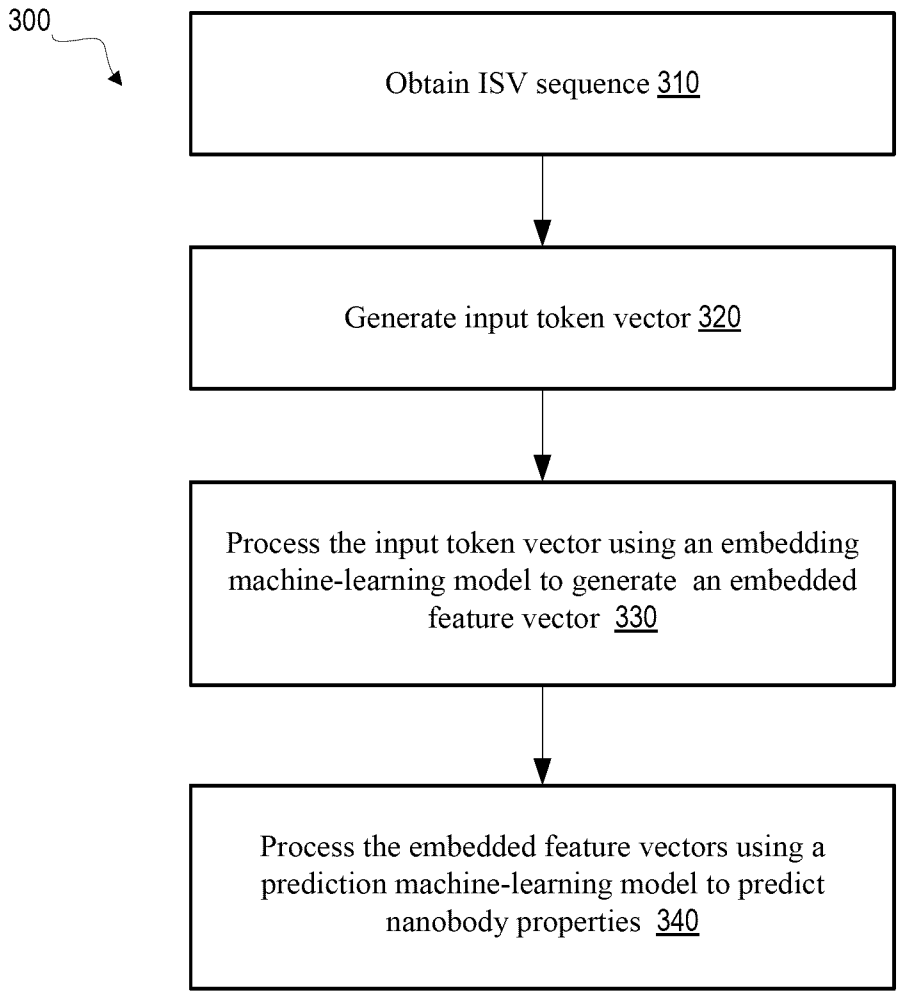

300
Obtain ISV sequence 310
Generate input token vector 320
Process the input token vector using an embedding machine-learning model to generate an embedded feature vector 330
Process the embedded feature vectors using a prediction machine-learning model to predict nanobody properties 340

FIG. 3

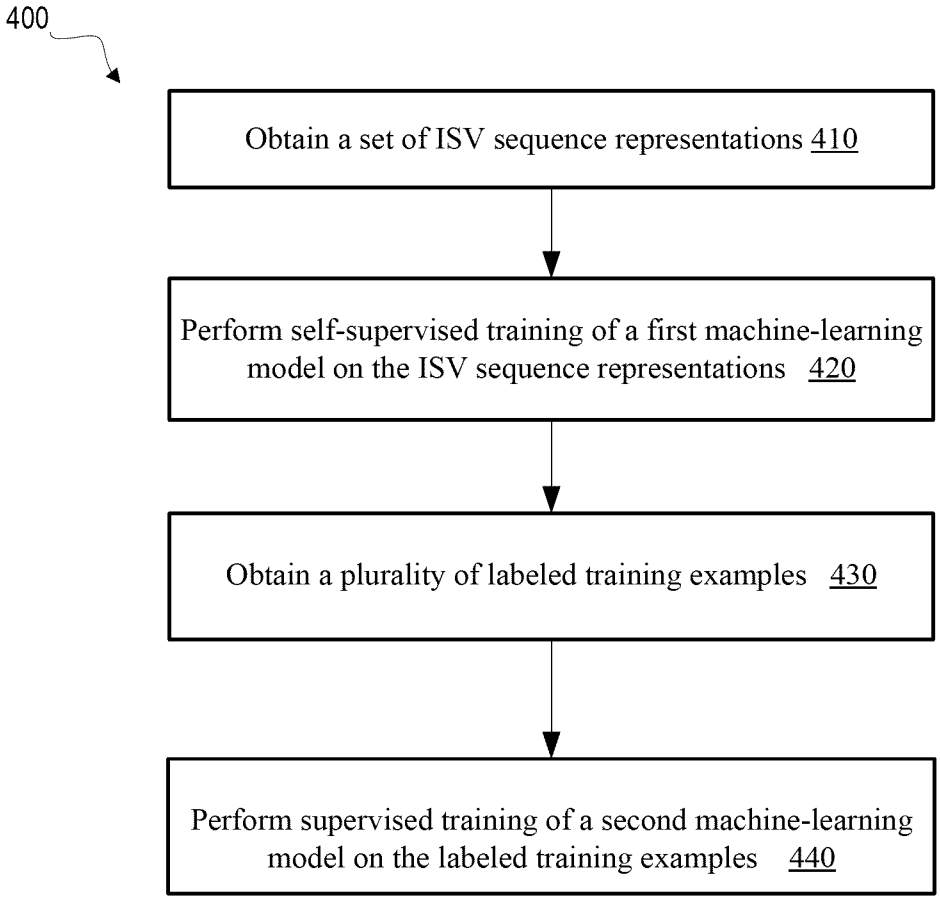

400
Obtain a set of ISV sequence representations 410
Perform self-supervised training of a first machine-learning model on the ISV sequence representations 420
Obtain a plurality of labeled training examples 430
Perform supervised training of a second machine-learning model on the labeled training examples 440

FIG. 4

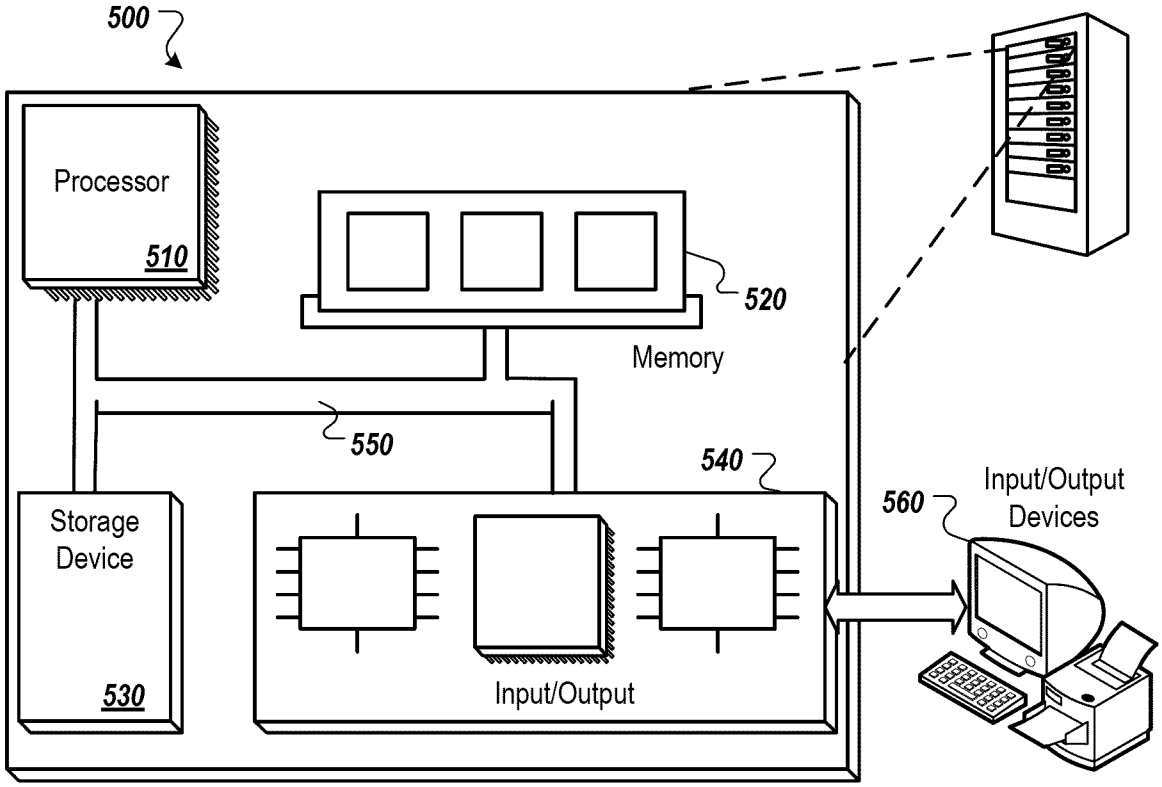
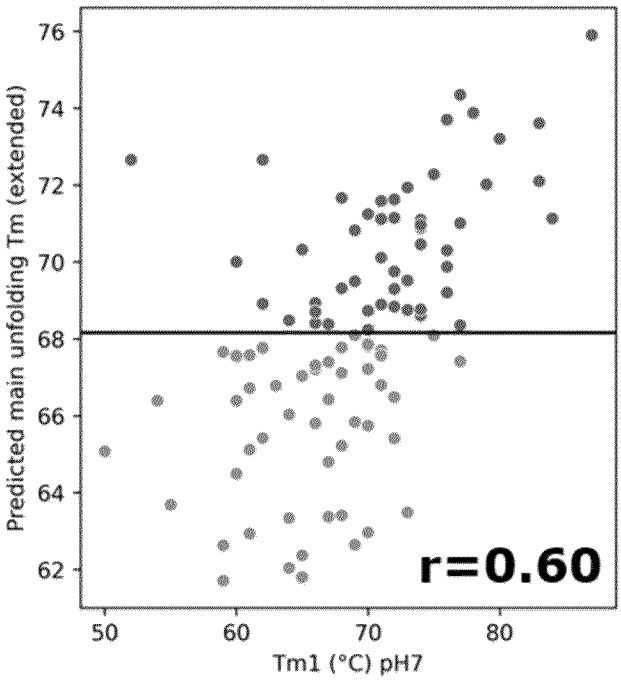

500
Processor
510
520
Memory
550
540
Storage Device
530
Input/Output
560
Input/Output Devices

FIG. 5

(a)


Predicted main unfolding Tm (extended)
Tm1 (°C) pH7
r=0.60

(b)

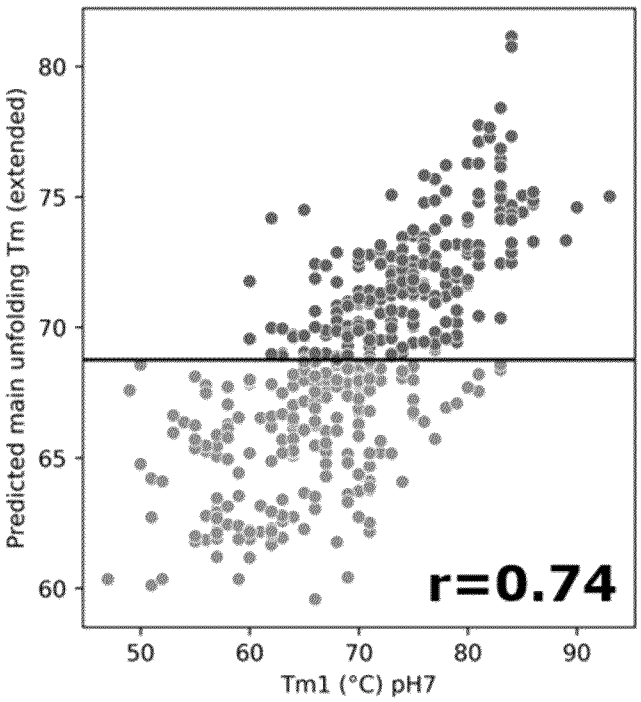

Predicted main unfolding Tm (extended)
Tm1 (°C) pH7
r=0.74

FIG. 6

INTERNATIONAL SEARCH REPORT

International application No
PCT/EP2024/065419

A. CLASSIFICATION OF SUBJECT MATTER
INV. G16B30/00 G16B40/20
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

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	HARMALKAR AMEYA ET AL: "Toward generalizable prediction of antibody thermostability using machine learning on sequence and structure features", MABS, [Online] vol. 15, no. 1, 22 January 2023 (2023-01-22), XP093186199, US ISSN: 1942-0862, DOI: 10.1080/19420862.2022.2163584 Retrieved from the Internet: URL:https://www.ncbi.nlm.nih.gov/pmc/articles/PMC9872953/pdf/KMAB_15_2163584.pdf> [retrieved on 2024-07-24] Abstract; Materials and methods (sections "nanoDSF Tm method", "Dataset distribution", "Pretrained language models", "Zero-shot evaluation", and "Finetuned evaluation"); -/--	1-15

[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
25 July 2024	06/08/2024

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	Hendrikse, Natalie

1

Form PCT/ISA/210 (second sheet) (April 2005)

C(Continuation). DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
	figures 1B, 1C, 2B, 3A, 4	
A	Pudziuvelyte Ieva ET AL: "TemStaPro: protein thermostability prediction using sequence representations from protein language models", bioRxiv, 23 April 2023 (2023-04-23), XP093189144, DOI: 10.1101/2023.03.27.534365 Retrieved from the Internet: URL:https://www.biorxiv.org/content/10.1101/2023.03.27.534365v2.full.pdf [retrieved on 2024-07-24] the whole document	1-15
X,P	US 2023/368861 A1 (HARMALKAR AMEYA [US] ET AL) 16 November 2023 (2023-11-16) paragraphs [0016], [0062], [0064], [0086], [0092], [0108] - [0113], [0169], [0209], [0239], [0246], [0259], [0267] - [0296]; figures 2A,7,8,14,15B	1-15

INTERNATIONAL SEARCH REPORT

Information on patent family members

International application No
PCT/EP2024/065419

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
US 2023368861 A1	16-11-2023	US 2023368861 A1	16-11-2023
		WO 2023220038 A1	16-11-2023