Amino Acid Composition drives Peptide Aggregation: Predicting Aggregation for Improved Synthesis

Bálint Tamás[1†], Marvin Alberts[1,2,3†], Teodoro Laino[2,3], and Nina Hartrampf[1*]

[1] *University of Zürich, Department of Chemistry, Winterthurerstrasse 190, 8057 Zürich, Switzerland*

[2] *IBM Research Europe, Säumerstrasse 4, 8803 Rüschlikon, Switzerland*

[3] *National Center for Competence in Research-Catalysis (NCCR-Catalysis), Zürich, Switzerland*

[*] *nina.hartrampf@chem.uzh.ch*

Abstract

Peptide aggregation is a long-standing challenge in chemical peptide synthesis, limiting its efficiency and reliability. Although data-driven methods have enhanced our understanding of many sequence-based phenomena, no comprehensive approach addresses so-called "non-random difficult couplings" (generally linked to aggregation) during solid-phase peptide synthesis. Here, we leverage existing peptide synthesis datasets, supplemented with newly acquired experimental data, to build a predictive model that deciphers the role of individual amino acids in triggering aggregation. First, we identified and experimentally validated composition-dependent aggregation as a stronger predictor than sequence-based patterns. This insight enabled the development of a composition vector representation, allowing insights into the aggregation propensities of individual amino acids. Applying an ensemble of trained models, we predict the aggregation properties of peptides and recommend optimized synthesis conditions. By elucidating each individual amino acid's influence, this method holds the potential to accelerate synthesis optimization through existing data, offering a robust framework for understanding and controlling peptide aggregation.

[†] Equal Contribution, Author order interchangeable

1 Introduction

Peptides and proteins play diverse biological roles, functioning as hormones, enzymes, and signalling molecules, which are critical for maintaining physiological processes. Their versatility and specificity have made them valuable therapeutic agents, driving innovations in the pharmaceutical industry. [1] Understanding their structures has been a long-standing challenge in biochemistry. [2, 3] Despite key advances, human intuition alone has proven insufficient for a systematic understanding of the structure of proteins based on their primary sequence, leading to the widely known "Protein Folding Problem". [4] With decades of accumulated data, computational methods have emerged as essential tools to predict the structure of proteins. [5–7] This evolution in methodology culminated in the development of AlphaFold and RoseTTAFold, effectively solving the problem of accurately predicting a protein's structure from its sequence. [8, 9]

While these developments have greatly enhanced our understanding of peptide and protein folding under physiological conditions, folding properties during solid-phase peptide synthesis (SPPS) remain comparatively unexplored. During SPPS, aggregation of resin- and linker-bound peptides often induces peptide folding, which can hinder synthetic efficiency and render certain sequences inaccessible. Aggregation is thought to originate from the undesired formation of β-sheet structures on the solid support. [10–13] This causes both truncations and deletions of the peptide sequence, often making it challenging, if not impossible, to isolate the desired peptide. Notably, even additional coupling or deprotection cycles and a large excess of amino acid do not lead to full conversion post-aggregation. Aggregation depends on several factors, such as synthesis temperature, loading of the solid support, and–most importantly–the peptide sequence and its amino acid side chain protecting groups. It has been shown that aggregation often occurs within 5–15 amino acids from the anchoring point to the resin. [14, 15] Consequently, C-terminal amino acids exert the greatest influence on aggregation, with current literature suggesting that β-branched amino acids aggravate this effect. [14] Despite multiple attempts to understand the sequence-dependence of aggregation experimentally [16–18] and with advanced data analytics on UV data obtained from flow-SPPS [19, 20], a robust method to predict aggregation and to propose an alternative synthesis strategy remains elusive.

In this study, we use machine learning on deprotection peak data collected from the inline UV-Vis of an automated fast-flow peptide synthesiser (AFPS) [21]. This data directly correlates to the aggregation state of the peptide being synthesised on the resin (see Figure 1 B). We leverage the UV-Vis data to gain new insights into the factors contributing to peptide aggregation, including the influence of each individual amino acid. Through shuffling of peptide sequences, it was found that the composition of the peptide, rather than the specific sequence, largely determines the aggregation characteristics of a given peptide. We verify this claim through experimental results and ultimately demonstrate

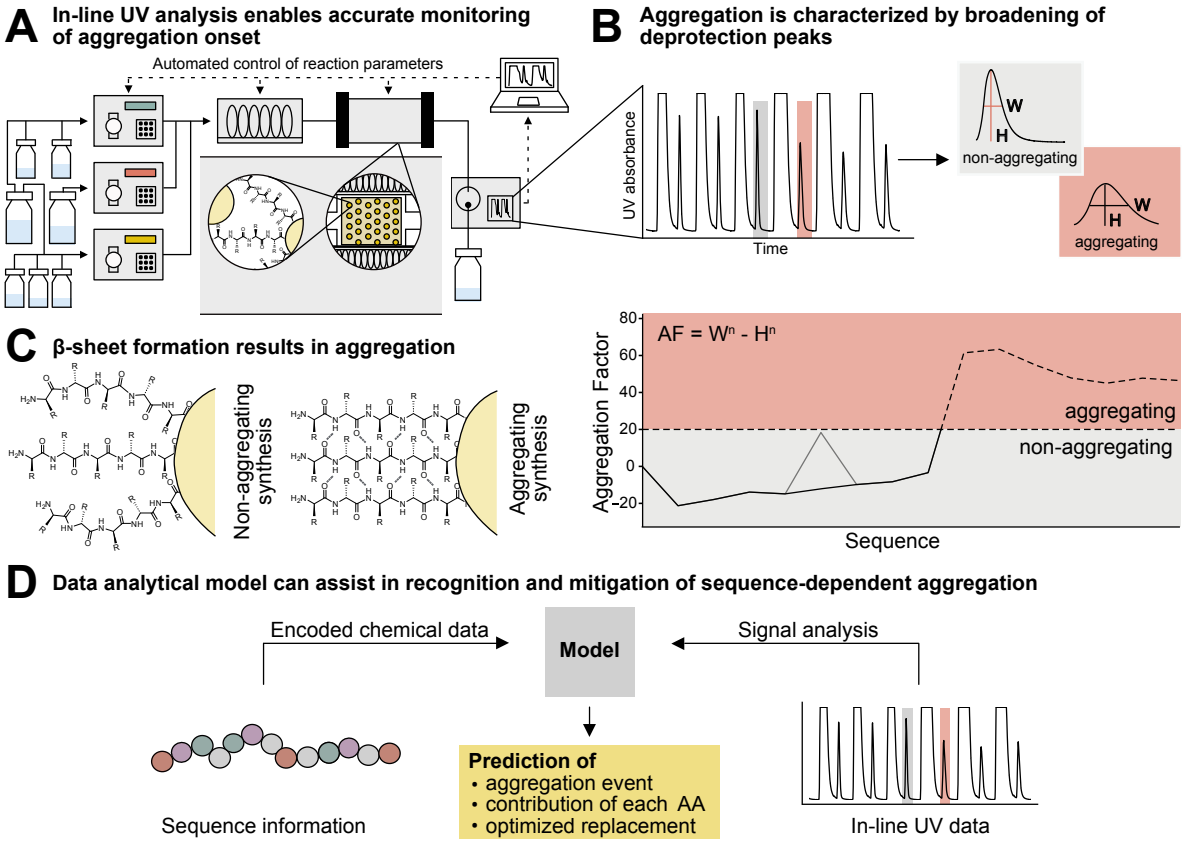

Figure 1: Analytical data collected with an in-line UV module enables data-driven methods for synthesis analysis. A) AFPS enables the precise monitoring of reaction kinetics, which corresponds to the aggregation of the sequences. B) Aggregation in the in-line UV traces is characterized as the broadening of the deprotection peak. Aggregation is quantified by an aggregation factor, calculated using the following formula: AF = Wn – Hn. Wn: half of the maximum height, normalized to the first peak, Wn: peak height normalized to the first peak. If AF $>$ 20, the sequence is considered aggregating. C) Aggregation is driven by β-sheet formation between the growing peptide chains. D) In-line UV data collected during synthesis was leveraged to predict the occurrence of aggregation and the contribution of individual amino acids.

how these findings can be used to avoid aggregation.

2 Results and Discussion

Prediction of aggregation during SPPS is model- and representation-independent.

Predicting peptide aggregation requires criteria to distinguish between aggregating and non-aggregating sequences. All data used in this study were collected on an AFPS platform equipped with an in-line UV-Vis detector monitoring coupling and deprotection peaks during synthesis (Figure 1A). Deprotection peaks, which result from 9-fluorenylmethyloxycarbonyl (Fmoc) removal, provide two crucial pieces of information:

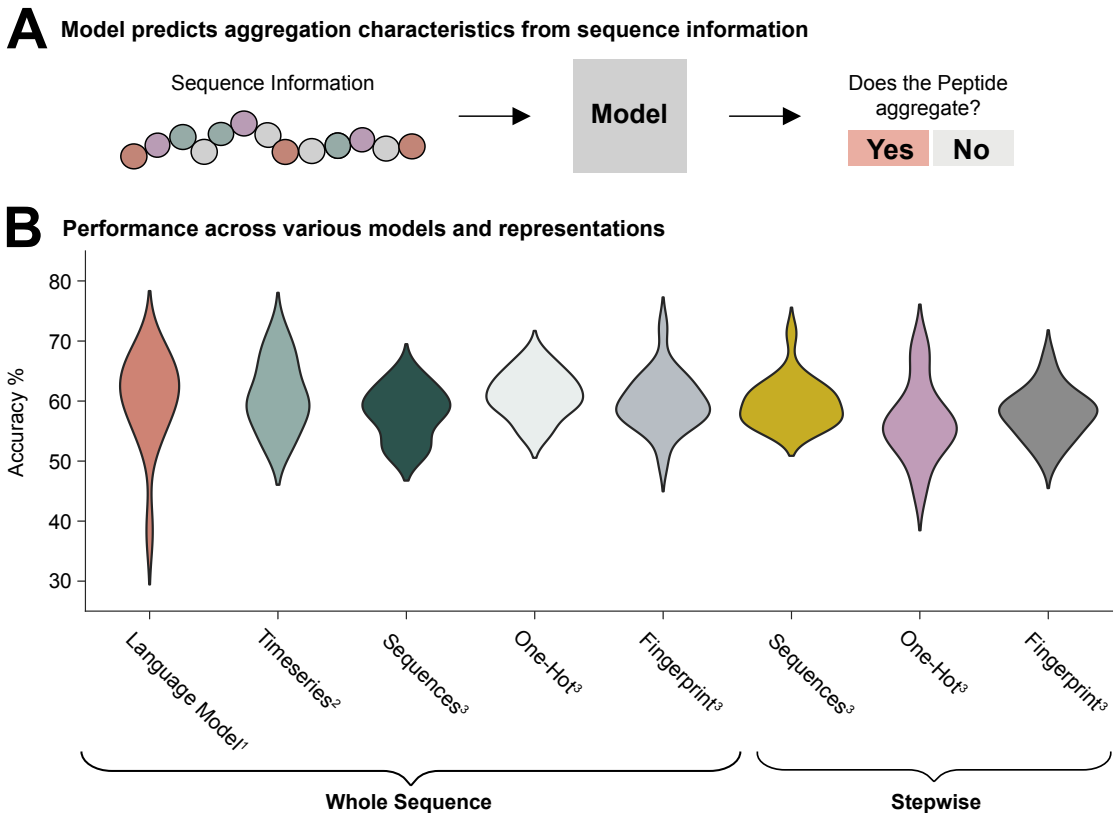

Figure 2: Prediction accuracy is independent of model or representation. A) A variety of different models ranging from language models to classical machine learning models were trained to predict whether a given peptide sequence aggregates or not. B) Consistent prediction accuracy scores are observed across all models and representations regardless of the model, chemical representation, or if the sequence is fed stepwise or as a whole sequence.
[1]: ESM 2.0, BERT. [2]: HIVE-COTE 2, WEASEL, TimeForest. [3]: XGBoost, Random Forest, KNN, Gaussian Processes.

their area indicates the coupling/deprotection efficiency, while their shape reflects the aggregation state. [16, 20, 22, 23] Following Mohapatra et al. [19], we defined aggregation as the deprotection peak broadening by more than 20% relative to the baseline. If any peak during synthesis exceeds this threshold, we classify the entire sequence as aggregating. In practice, this directly correlates to a decreased crude purity (Figure 1B).

Next, we used machine learning to predict the aggregation characteristics of a given peptide using two datasets: One published by Mohapatra et al. [19] and one internal dataset. Both were generated using similar AFPS platforms [21] and synthesis conditions, ensuring minimal statistical deviation between the two. After curating and merging the two datasets (see Methods 1), the combined dataset comprised 539 peptide sequences. Of the total sequences, 420 were sourced from the Mohapatra dataset, with 48.8% showing

aggregation, and an additional 119 sequences from our internal dataset, where 53.8% aggregated. This resulted in a nearly balanced combined dataset, with 49.9% of sequences exhibiting aggregation. As aggregation typically occurs 5–15 amino acids from the anchoring point to the resin, all peptides longer than 20 amino acids were truncated and those shorter than five amino acids were discarded (see Supporting Information Figure 1 for length distribution).

While extensive research has been conducted on identifying suitable statistical models and molecular representations for proteins, considerably less attention has been devoted to peptides. To address this gap, we explored a wide range of models and representations for peptide synthesis data. In all cases, we framed the problem as a binary classification task: Does a given peptide sequence aggregate or not? Our data was collected during synthesis, allowing for two distinct prediction approaches: either predicting the aggregation characteristics of the final synthesized peptide directly or leveraging the step-by-step nature of the synthesis process. During synthesis, the peptide is elongated amino acid by amino acid, with information on whether the peptide has aggregated available at each synthesis step. We explored both approaches for the predictions (Figure 2): "Whole Sequence" corresponds to predictions based on the final peptide sequence and "Stepwise" emphasises the step-by-step nature of the syntheses. For the step-by-step approach, all peptides are labelled as non-aggregating for the first few couplings. Once an aggregation event, i.e. broadening of the deprotection peak, occurs, all subsequent peptide couplings are labelled as aggregating.

To evaluate both approaches, we experimented with a range of models and representations. One highly successful approach for proteins treats the amino acid sequence as text and leverages language models to predict protein properties. [24, 25] Inspired by this approach, we fine-tuned a specialized protein language model (ESM2.0 [26]) as well as a generalist language model (BERT [27]) to classify whether a peptide aggregates or not. In addition to fine-tuning pretrained models, we also trained a BERT model from scratch (Figure 2, Language Models).

Another common data type in machine learning are time series. A time series consists of a sequence of data points collected at regular time intervals. Following this definition, the stepwise synthesis of a peptide can be considered a time series with each addition of an amino acid corresponding to one time step. We trained three state-of-the-art time series classification models on this problem, representing each amino acid with a numerical token and padding to accommodate varying sequence lengths (see Figure 2, Timeseries).

In addition to these models, we also explored the performance of classical machine learning models (e.g. Random Forest [28], XGBoost [29]) on three different representations. These representations consist of a numerical token matching the approach for timeseries models, a one-hot encoding approach, and a fingerprint-based method inspired

by Mohapatra et. al (see Methods 5.1 for more detail). [19] All models were trained with five-fold cross-validation and the performance of each model was assessed using the accuracy. Surprisingly, we observed similar performance across all representations, models, or hyperparameter configurations.

To further guide the model, we focused on labelling the most relevant segment of the sequence, leveraging the step-by-step nature of the synthesis process. We hypothesized that the sequence preceding the aggregation point, i.e. the amino acid coupling at which aggregation occurs, is the most informative to distinguish between aggregating and non-aggregating sequences. In contrast, the remaining peptide sequence beyond the aggregation point contains little to no meaningful information. Therefore, we systematically investigated how many amino acids before and after the aggregation point are ideal to label as aggregating: We evaluated ranges up to ten amino acids before and after the point of aggregation (see Supporting Information Figure 2). The model's performance remained consistent regardless of the modified hyperparameters, models, or representation used, provided the sequences were sufficiently long to form secondary structures (¿6 amino acids). [30] This suggests that aggregation may be determined by factors other than peptide sequence or the models were unable to effectively capture the aggregation signal from the data.

Amino acid composition, rather than the sequence itself, influences aggregation.

The consistent results across different models and representations prompted us to question the quality and consistency of our dataset. As a validation experiment, the models were trained on a shuffled version of the peptide sequence. Assuming aggregation is highly sequence-dependent, inconsistent performance with shuffled data would indicate that the models fail to capture a sequence-specific aggregation signal.

We trained XGBoost models using whole sequence representation on both the original and a randomly shuffled dataset. No significant difference in accuracy was observed ($0.580 \pm 0.035\%$ for original sequences vs $0.579 \pm 0.036\%$ for the shuffled sequences). This result was consistent across all tested representations and models (see Supporting Information Section 3). These findings challenge the widely accepted view of aggregation as a phenomenon that is highly dependent on peptide sequence. [15] To investigate this further, a simplified encoding method was developed, representing each sequence as a 20-dimensional vector corresponding to the normalized composition of amino acids. Using this minimal representation, the accuracy remains comparable ($0.610 \pm 0.038\%$), reinforcing the notion that amino acid composition might outweigh sequence order in influencing aggregation (Figure 3A).

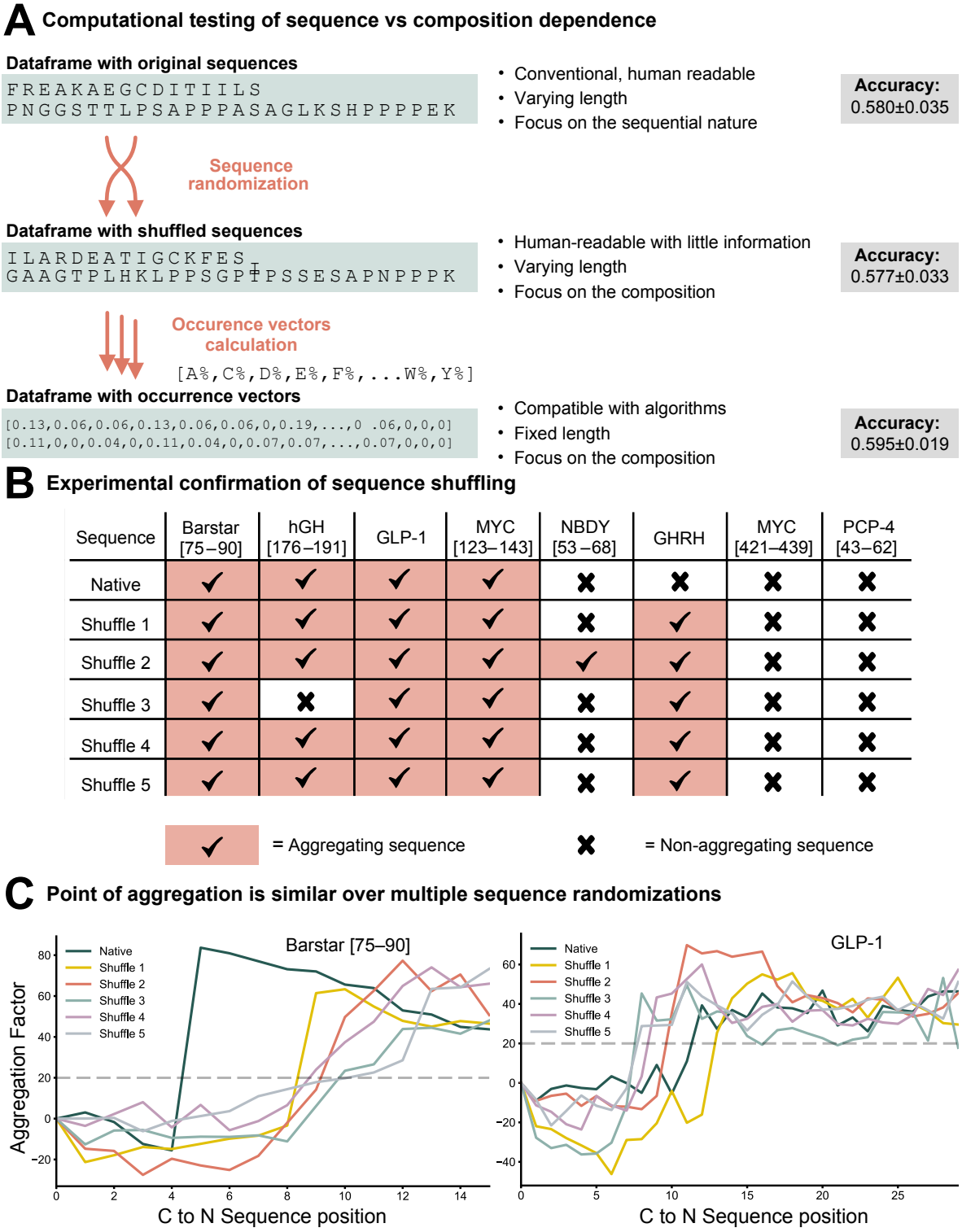

Figure 3: Computational and experimental investigation of sequence shuffling on aggregation behaviour. A) Training models on randomly shuffled sequences or only with a composition vector of the amino acids present in the peptide does not lead to a decrease in accuracy compared to training on the original sequences. B) To verify the computational results, four aggregating and four non-aggregating sequences were synthesized with five reproducible shuffles each. C) For aggregating test peptides, the point of aggregation remains consistent across the shuffled peptide sequences. Native UV-Vis traces for Barstar and GLP-1 were adapted from Tamás *et al.* [20] and Bürgisser *et al.* [13].

To experimentally test whether composition, rather than sequence, determines aggregation, we selected eight literature-known test peptides and synthesized five randomly shuffled variants of each peptide (Figure 3B). Barstar[75–90] [20], hGH[176–191]Y176F [13] (abbreviated as hGH), GLP-1 [13], and MYC[123–243] [13] were selected as aggregating sequences, and NBDY[53–68] [20], GHRH [21], MYC[421–439], and PCP-4[43–62] as non-aggregating sequences. The shuffled sequences were generated through a reproducible randomization process to avoid selection bias. The peptides, ranging from 16 to 28 amino acids in length, were experimentally evaluated for aggregation behaviour during AFPS. In alignment with the in silico results, 19 out of 20 of the shuffled aggregating peptides retained their aggregation characteristics, while 14 out of 20 of the shuffled non-aggregating sequences also maintained their non-aggregating character (Figure 3). The majority of peptides preserve their aggregation characteristics, regardless of amino acid order, as long as the overall composition remains unchanged. In addition, the aggregation point also remains similar for the shuffled sequences (Figure 3C). This suggests that factors beyond the sequence, i.e. amino acid composition, play a prominent role in determining peptide aggregation rather than sequence information alone.

Interpretation of individual amino acid contribution to aggregation

To understand the impact of each individual amino acid on aggregation, we leveraged Shapley Additive Explanations (SHAP) values [31]. SHAP enables the quantification of the contribution of each amino acid to the aggregation propensity of a peptide sequence. In these experiments, the amino acid composition vector was used as the representation, establishing a direct link between the composition of amino acids and the model prediction.

The analysis revealed distinct patterns in how different amino acids influence aggregation (Figure 4). Amino acids such as Ser(*t*-Bu), Ile, Val, and Thr (*t*-Bu) were found to increase the likelihood of aggregation when present in higher proportions. Conversely, the presence of Phe, Asp(*t*-Bu), Tyr(*t*-Bu), and Arg(Pbf) tended to reduce aggregation. The remaining amino acids appeared to contribute neutrally, without a strong positive or negative effect (see Supporting Information Section 5). While our analysis revealed peptide composition to be predominantly driving aggregation, other factors influence aggregation as well. To this end, we investigated the effect of dipeptide motifs on aggregation, with Gly–Ser and Leu–Leu contributing most to aggregation (see Supporting Information Section 6).

The aggregation-promoting amino acids generally have aliphatic, non-polar side chains, which seem to facilitate intermolecular interactions and packing between peptide strands. In contrast, amino acids that inhibit aggregation often have aromatic or polar side groups, which may increase spacing and disrupt aggregation-prone structures.

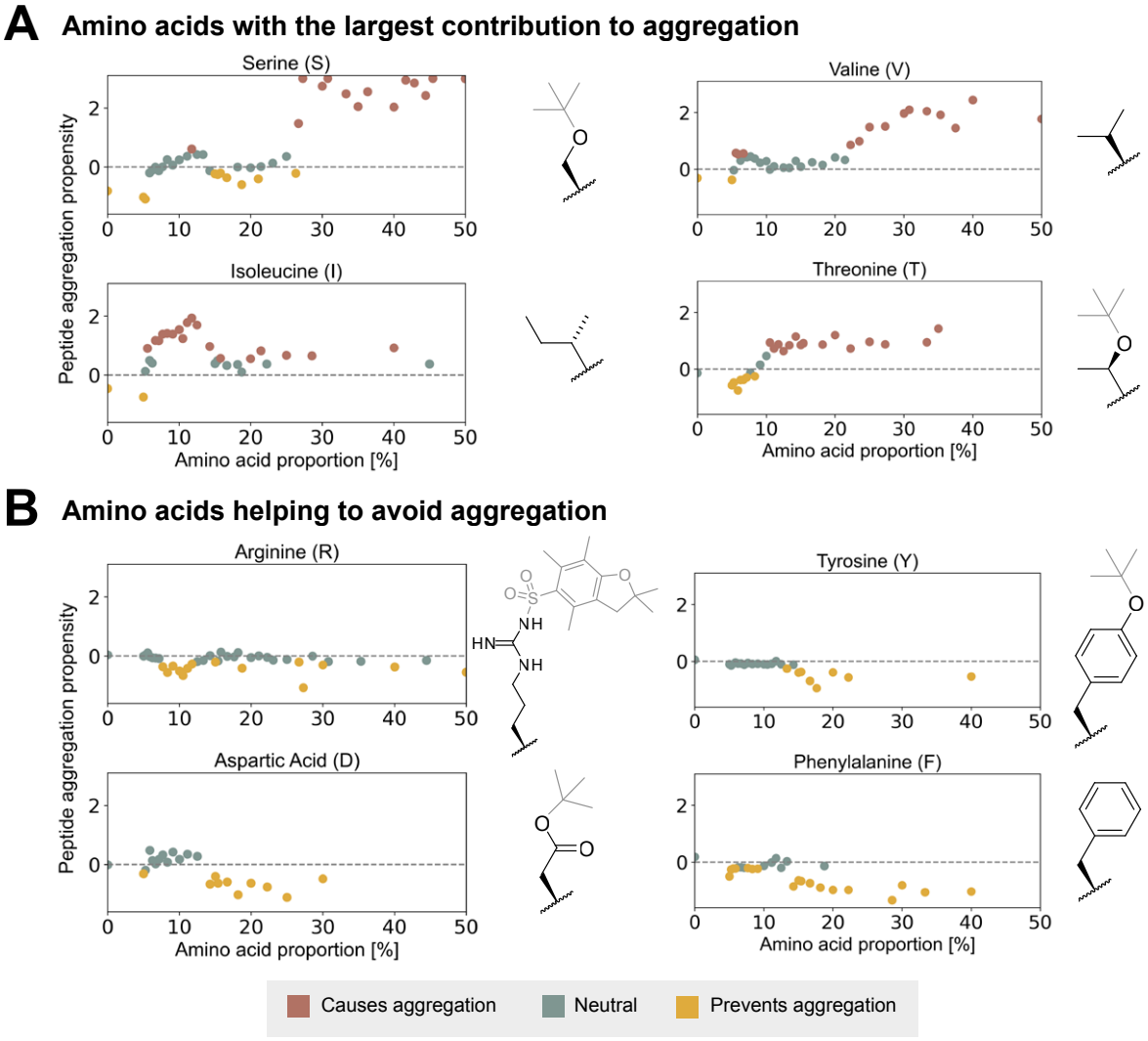

Figure 4: Analysis of amino acids influencing the model's decision-making the most. The X-axis represents the amino acid proportion in the sequences, with the Y-axis corresponding to the importance the model assigns to each data point. A positive value is associated with a higher likelihood of the model predicting aggregation and a negative one with a lower aggregation chance. A) Amino acids that contribute the most to aggregation: serine, valine, isoleucine, and threonine. B) Amino acids that contribute the least to aggregation: arginine, tyrosine, aspartic acid, and phenylalanine.

Trained models suggest conditions for improved solid-phase peptide synthesis

The optimization of peptide synthesis can be a tedious process: As sequence-dependent events such as aggregation are difficult to predict, the usual workflow requires repetitive synthesis with the trial-and-error use of known aggregation-reducing tools. Our trained model not only enables the prediction of the aggregation propensity of a given peptide but also provides insights into how aggregation could be mitigated through strategic modifications. By understanding the contributions of specific amino acids, we can predict the

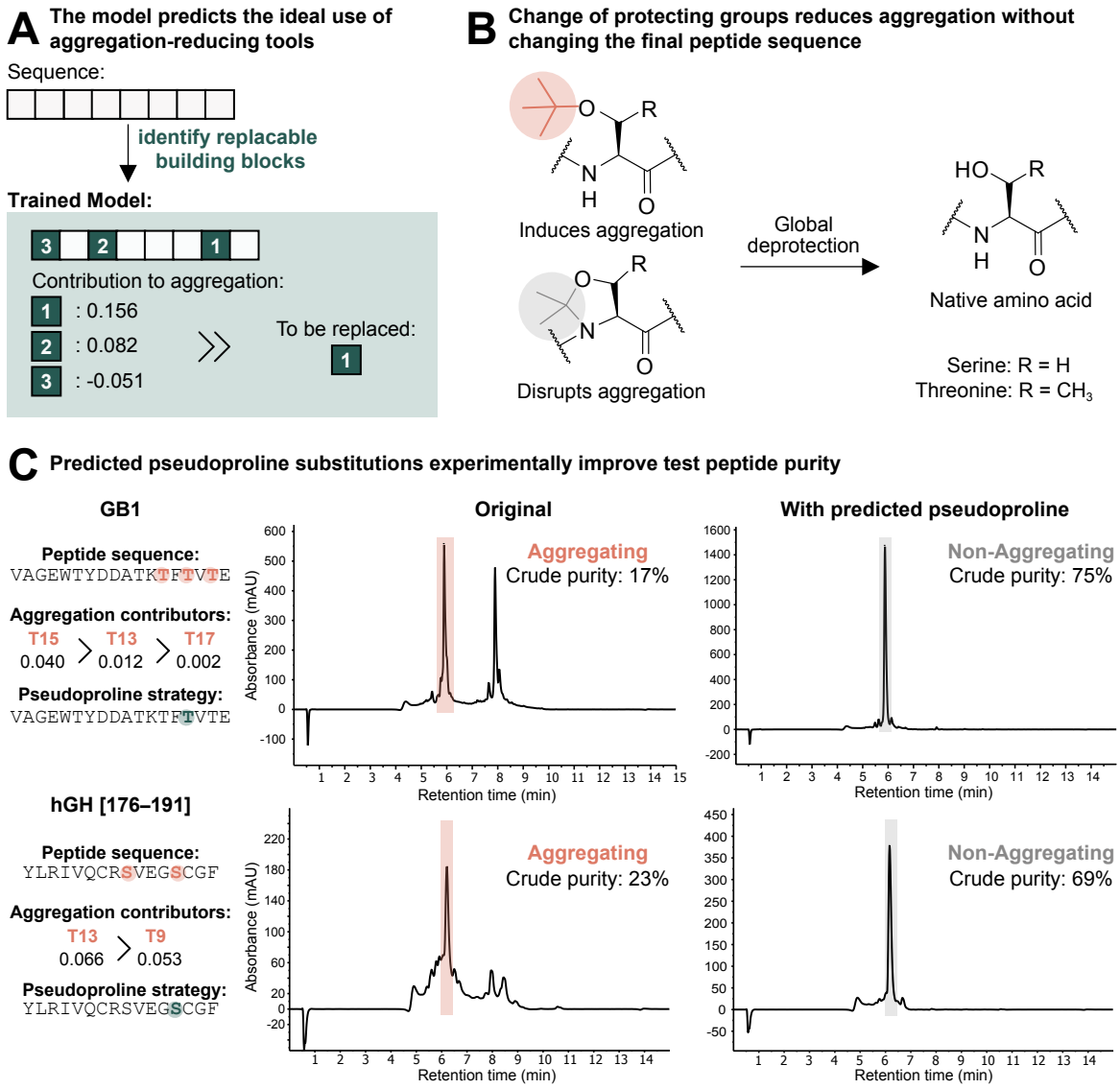

Figure 5: Leveraging the model for rational use of aggregation reduction tools to suggest improved synthesis conditions. A) With the user input sequence and replaceable amino acids, the trained model ensemble predicts and scores the aggregation property of the sequence and predicts the contribution of the present amino acids in the early fragment of the peptide (position 2–12). This enables more effective introduction of aggregation-suppressing moieties. B) Serine and threonine, two *t*-Bu-protected amino acids with significant predicted contribution to aggregation, can also be introduced as pseudoprolines. The latter are established aggregation-reducing tools, which upon global deprotection yield the native amino acid. C) The potential of the model was tested in two known aggregating sequences: GB1 and hGH. The serines and threonines with the largest contribution were selected and replaced, resulting in a significant purity increase of 58% for the GB1 fragment and 46% for the hGH fragment.

most effective use of aggregation-reducing tools, such as different backbone and side chain protecting groups. The algorithm we developed works as follows (Figure 5A): 100 models were trained on varying splits of the data, forming an ensemble to avoid bias stemming from the relatively small size of the dataset. The user inputs the peptide sequence and the amino acids with available aggregation-reducing substitutions. The models then predict

whether the given sequence is likely to aggregate. If the sequence is predicted to be aggregating, the models analyse the key positions (2–12) to identify amino acids that could be substituted with their aggregation-reducing counterparts. These potential substitutions are then ranked in order of their relative contribution to aggregation, allowing the user to prioritize the most impactful changes. By substituting the highest-contributing residues, the synthesis process can be optimized to avoid aggregation issues.

To test this capability, we selected two aggregating sequences, hGH and GB1, and pseudoproline-protected amino acid building blocks as a widely used tool to mitigate aggregation [32]. The use of pseudoprolines is advantageous as they serve as an aggregation-disrupting equivalent of the two protected amino acids with the highest contribution, Ser(t-Bu) and Thr(t-Bu) (Figure 5B). For hGH, 74% of the models predicted aggregation, whereas for GB1 this increased to 90%. Next, the contribution of Ser(t-Bu) and Thr(t-Bu) in the 2–12 amino acids from the resin (C-terminus) was assessed. In both cases structural motifs contributing to aggregation were identified: Ser(t-Bu) in position 13 for hGH and Thr(t-Bu) in position 15 for GB1. We synthesized both optimised sequences on the AFPS and, in both cases, we could confirm a reduction of aggregation via in-line UV signal and MS-MS. The incorporation of pseudoproline resulted in a crude purity increase from 23% to 69% for hGH and 17% to 75% for GB1. In summary, the developed algorithm can use the trained model to predict the aggregation property and suggest efficient incorporation of aggregation-reducing tools to increase synthetic efficiency.

3 Conclusions

In this study, machine learning was used as a discovery tool, uncovering a surprisingly strong composition-dependence of peptide aggregation. This finding was validated experimentally by synthesizing forty sequences (eight sequences, each shuffled five times). In the process, we developed a simple composition vector as a new peptide representation to investigate the aggregation character during SPPS. By leveraging the interpretability of this representation, we found that bulkier and more polar side chains or protecting groups have a tendency to reduce aggregation, while characteristically aliphatic side chains increase the likelihood of aggregation. In addition, we demonstrated the practical value of these findings by pinpointing the key amino acids contributing to aggregation in a given target peptide. By strategically introducing pseudoprolines at these positions, we observed a reduction in aggregation and an increase in the purity of two test sequences by 58% and 46%, respectively.

These findings question the understanding of aggregation as a mainly sequence-dependent event originating from intermolecular hydrogen bonding between backbones, resulting in β-sheet structures. [10, 11, 13, 15, 30] For biological systems, it has been established that

amino acids with aliphatic side chains, such as valine or leucine, tend to be large contributors to β-sheet formation and aggregation. [33,34] Aromatic side chains also seem to have a major impact on the aggregation of native peptides and proteins under physiological conditions. [34] Our findings revealed that during SPPS amino acids with aliphatic side chains, such as valine or isoleucine, predominantly contribute to aggregation. Similarly, protecting groups that mimic these structures, such as *t*-Bu-protected serine or threonine, exhibit similar behavior during SPPS. In contrast to native peptides, amino acids with aromatic side chains or protecting groups, such as phenylalanine or tyrosine, tend to reduce aggregation occurrence. Furthermore, aggregation is widely considered sequence-dependent, yet our results indicate that during SPPS, amino acid composition is more influential. This discovery led to the development of the composition vector, a simplified representation of peptides allowing us to predict the onset of aggregation, while also recommending mitigation strategies.

Our machine learning driven approach revealed previously undetected patterns in peptide aggregation. The strong correlation between peptide composition and aggregation emerged only through the use of computational analysis, highlighting how machine learning can discover complex relationships in chemical systems. This work demonstrates that machine learning's value in chemistry extends beyond its common applications in property prediction and molecular generation: It serves as a powerful discovery tool that can challenge established paradigms and uncover hidden patterns in molecular data.

4 Methods

4.1 Computational Methods

4.1.1 Dataset Curation

The data used in this study consists of the UV-traces gathered during the SPPS of various peptides. We used the dataset published by Mohapatra *et. al.* containing 769 unique syntheses in addition to an internal dataset of 167 unique syntheses. Both datasets were combined, and all syntheses containing non-canonical amino acids, steps not performed on an AFPS (e.g. batch synthesis of a pre-chain), and synthesis of peptides with fewer than five amino acids were removed. As aggregation was reported to primarily occur between amino acids 5 and 15, only the synthesis steps of the first 20 amino acids were considered. [15] In addition, we filtered all duplicated sequences from the dataset. This reduced the size of the combined dataset to 539 unique syntheses. We defined aggregation as a broadening of the deprotection peak in excess of 20% compared to the first deprotection peak. During the synthesis, the addition of histidine and cysteine requires changes in the temperature of the reactor causing a broadening of the deprotection peak.

Following Tamas et. al. [20] we ignored these peaks and interpolated with the previous and subsequent peaks for all histidine and cysteine additions.

4.1.2 Data Processing

We used the following processing strategies for the peptide sequences:

Step-by-Step: Since SPPS builds the peptide sequence one amino acid at a time and aggregation information is available for each synthesis step, the problem can be framed as predicting whether a peptide sequence has aggregated at a given synthesis step. In theory, this approach has multiple advantages. It exposes the model to a considerably larger amount of training data (a total of 7.000 synthesis steps in the dataset) and enables the practitioner to not only predict whether a peptide will aggregate, but also pinpoint where aggregation occurs. In total, this approach yielded 7.000 training samples.

Whole Peptide: In this approach, we only considered the full peptide sequence and labeled it as aggregating or not aggregating. This yields 500 training samples.

4.1.3 Peptide Representation

Text: In this approach we leveraged pretrained Transformer models to predict whether a peptide aggregates or not. The peptide sequence is used as is and fed into the tokenizer of the Transformer model. ESM and BERT models were used.

Sequence: This representation converts a peptide into a vector by mapping each amino acid to a value between 1 and 20. We padded each sequence to the maximum sequence length (in this case 20) and fed this vector into the models.

One Hot Encoding: This approach works similarly to sequence representation. Instead of mapping each amino acid to a numerical value, we one-hot encoded each amino acid and concatenated the vectors. In addition, we pad the resultant vector to match the maximum sequence length.

Fingerprint: This approach is inspired by Mohapatra et. al. Here we used a Morgan Fingerprint [35] with a radius of three and a bit size of 128 to represent each amino acid. We concatenated the fingerprint for each amino acid and padded the vector with zero to a uniform length regardless of the sequence size.

Composition Vector: For a given peptide we constructed a normalized vector where each index corresponds to a specific amino acid. This vector is built as follows: Assign a fixed index to each of the 20 standard amino acids, creating a 20-dimensional vector followed by counting the number of occurrences of each amino acid and populating the corresponding vector indices. This vector is normalized by dividing by the total number of amino acids, ensuring that the vector represents the proportional composition of the peptide independent of its length.

4.1.4 Models

All models were trained with five-fold cross-validation and a fixed seed.

Fine-tuning ESM 2.0 and BERT: For ESM 2.0 and BERT, the implementations provided on Huggingface were used. The problem is phrased as a sequence classification task for a given peptide sequence. The entire model is fine-tuned. We used a standard Huggingface trainer with a learning rate of 2.5e-5, a batch size of 16 and a weight decay of 0.01. Adam is used as an optimiser with β_1 of 0.9 and β_2 of 0.99. We trained each model for 15 epochs and evaluated the model with the best validation loss. For ESM 2.0 we evaluated the sizes varying from 8M, 35M, 350M to 650M whereas for BERT we evaluated the base and large checkpoints. For ESM 2.0 we only used pretrained models whereas for BERT we both fine-tuned a pretrained model and trained a model for each size from scratch.

All time series models were used as implemented in the SKTIME library [36] using the default parameters.

HIVE COTE V2: We used the implementation as provided by SKTIME with 500 estimators and a time limit of 10 minutes. [37]

WEASEL: Weasel is used with Anova and bi-grams using "information-gain" as the binning strategy. [38]

Time Forest: The time series forest classifier is used with a minimum interval of three and 200 estimators. [39]

XGBoost: We used the implementation in the XGBoost library [29] with the default settings.

Scikit-learn Models:

All scikit-learn models are used with the default hyperparameters. We evaluated the Random Forest-, Gaussian Processes-, and KNN-Classifier. [40]

4.1.5 Explainability

We used the Shap library [31] to explain the predictions of the models. Specifically, we leveraged the TreeExplainer and we trained a total of 50 models on random splits of the data to avoid noise in the explanations.

4.2 Experimental

4.2.1 Reagents and solvents

Fmoc- and side chain-protected L-amino acids (Fmoc-Ala-OH, Fmoc-Arg(Pbf)-OH, Fmoc-Asn(Trt)-OH, Fmoc-Asp(O*t*-Bu)-OH, Fmoc-Cys(Trt)-OH, Fmoc-Gln(Trt)-OH, Fmoc-Glu(O*t*-Bu)-OH, Fmoc-Gly-OH, Fmoc-His(Trt)-OH, Fmoc-Ile-OH, Fmoc-Leu-OH, Fmoc-Lys(Boc)-OH, Fmoc-Met-OH, Fmoc-Phe-OH, Fmoc-Pro-OH, Fmoc-Ser(tBu)-OH, Fmoc-Thr(*t*-Bu)-

OH, Fmoc-Trp(Boc)-OH, Fmoc-Tyr(*t*-Bu)-OH, Fmoc-Val-OH) and *N*'-tetramethyluronium hexafluorophosphate (HATU) were purchased from Bachem; O-(7-azabenzotriazol-1-yl)-*N*,*N*,*N*' and (7-azabenzotriazol-1-yloxy)tripyrrolidinophosphonium hexafluorophosphate (PyAOP) were purchased from Advanced ChemTech; *N*,*N*-diisopropylethylamine (*i*-Pr_2NEt, DIPEA, 99.5%) was purchased from Sigma-Aldrich; trifluoroacetic acid (TFA, for HPLC, ≥99.0%), triisopropylsilane (TIPS, 98%) and 3,6-dioxa-1,8-octane-dithiol (DODT, 95%) were purchased from Sigma-Aldrich. *N*,*N*-Dimethylformamide (DMF) was purchased from the from VWR International GmbH; dichloromethane (DCM, ≥99.8%) was purchased from Fisher Scientific Ltd.; diethyl ether was purchased from Honeywell Riedel-de Haën; acetonitrile (MeCN, for HPLC gradient grade, ≥99.9%) was purchased from Sigma-Aldrich. NovaPEG Rink Amide resin (0.41 or 0.20 mmol/g loading) was purchased from the Novabiochem-line from Sigma-Aldrich Canada Ltd. Piperidine (>99%, for synthesis) was purchased from Carl Roth GmbH. Formic acid (reagent grade, >95%) and AldraAmine trapping agent added to DMF were purchased from Sigma-Aldrich Canada Ltd.

4.2.2 Automated flow-based peptide synthesis (AFPS)

Peptides were synthesized on an automated flow system built in the Hartrampf lab, which is similar to the published AFPS system. [21] Capitalized letters refer to L-amino acids. For all synthesis (referred to as standard AFPS protocol) the following settings were used for peptide synthesis: flow rate = 20 mL/min for coupling and deprotection steps, temperature = 90 °C (loop) for all canonical amino acids, except histidine and cysteine which were coupled at room temperature and 90 °C (reactor). The standard synthetic cycle involves a first step of prewashing the resin at 90 °C for 60 s at 40 mL/min. During the coupling step, three HPLC pumps are used: a 50 mL/min pump head pumps the activating agent, a second 50 mL/min pump head pumps the amino acid, and a 5.0 mL/min pump head pumps *i*-Pr_2NEt (neat). The 50 mL/min pump head pumps delivered 0.398679 mL of liquid per pump stroke, the 5.0 mL/min pump head pumps 3.9239 × 10-2 mL of liquid per pump stroke.

All peptides were prepared by AFPS on NovaPEG Rink Amide resin (0.41 mmol/g) and standard Fmoc/*t*-Bu protected amino acids (0.40 M in DMF) were coupled using HATU (0.38 M in DMF) or PyAOP (0.38 M in DMF) with DIPEA (neat, 3.0 mL/min) at a total flow rate of 20 mL/min. For amino acids D, E, F, G, I, K, L, M, P, S, W, and Y, a total volume of 6.4 mL of the "coupling solution" (i.e. amino acid (0.20 M), HATU or PyAOP (0.19 M), and DIPEA in DMF) was applied for each coupling. For amino acids A, C, H, N, Q, R, S, T, and V, a total of 10.4 mL of "coupling solution" was applied for each coupling. All amino acids except C and H were preheated at 90 °C during the activation step with HATU or PyAOP, whereas C and H were preactivated with PyAOP at room temperature. Removal of the Nα-Fmoc group was achieved using

20% piperidine with 1% formic acid in DMF at a flow rate of 20 mL/min and a total volume of 6.4 mL at 90 °C. Between each coupling and deprotection step, the resin was washed with DMF (32 mL) at 90 °C with a flow rate of 40 mL/min. After completion of the peptide sequence, the resins were manually washed with DCM (3 × 5 mL) and dried under reduced pressure.

5 Data and Models availability

The code for generating the data and training the models is freely available on GitHub: https://github.com/rxn4chemistry/AI4Aggregation and the data is available on Zenodo: https://zenodo.org/records/14824562

6 Competing Interests

All authors declare no competing interests.

7 Acknowledgments

Financial support for this project was provided by the Swiss National Science Foundation (N.H.; project grant no. 200021_200865) and the University of Zurich. B.T. is additionally supported by a Candoc grant from the University of Zurich. This publication was created as part of NCCR Catalysis (M.A. and T.L.; grant number 180544), a National Centre of Competence in Research funded by the Swiss National Science Foundation. We would like to thank Dr. A. Jeandin and C. E. Grigglestone for helpful discussions during the preparation of this manuscript.

References

[1] Lei Wang, Nanxi Wang, Wenping Zhang, Xurui Cheng, Zhibin Yan, Gang Shao, Xi Wang, Rui Wang, and Caiyun Fu. Therapeutic peptides: Current applications and future directions. *Signal Transduction and Targeted Therapy*, 7(1):1–27, 2022.

[2] M. F. Perutz, M. G. Rossmann, A. F. Cullis, H. Muirhead, G. Will, and A. C. North. Structure of haemoglobin: A three-dimensional Fourier synthesis at 5.5 Å. resolution, obtained by X-ray analysis. *Nature*, 185(4711):416–422, 1960.

[3] Letícia M. F. Bertoline, Angélica N. Lima, Jose E. Krieger, and Samantha K. Teixeira. Before and after AlphaFold2: An overview of protein structure prediction. *Frontiers in Bioinformatics*, 3, 2023.

[4] Ken A. Dill, S. Banu Ozkan, M. Scott Shell, and Thomas R. Weikl. The Protein Folding Problem. *Annual Review of Biophysics*, 37:289–316, 2008.

[5] David E. Shaw, Paul Maragakis, Kresten Lindorff-Larsen, Stefano Piana, Ron O. Dror, Michael P. Eastwood, Joseph A. Bank, John M. Jumper, John K. Salmon, Yibing Shan, and Willy Wriggers. Atomic-Level Characterization of the Structural Dynamics of Proteins. *Science*, 330(6002):341–346, 2010.

[6] Marcin J. Skwark, Daniele Raimondi, Mirco Michel, and Arne Elofsson. Improved Contact Predictions Using the Recognition of Protein Like Contact Patterns. *PLOS Computational Biology*, 10(11):e1003889, 2014.

[7] Kim T. Simons, Rich Bonneau, Ingo Ruczinski, and David Baker. Ab initio protein structure prediction of CASP III targets using ROSETTA. *Proteins: Structure, Function, and Bioinformatics*, 37(S3):171–176, 1999.

[8] Carol A. Rohl, Charlie E. M. Strauss, Kira M. S. Misura, and David Baker. Protein structure prediction using Rosetta. *Methods in Enzymology*, 383:66–93, 2004.

[9] John Jumper, Richard Evans, Alexander Pritzel, Tim Green, Michael Figurnov, Olaf Ronneberger, Kathryn Tunyasuvunakool, Russ Bates, Augustin Žídek, Anna Potapenko, Alex Bridgland, Clemens Meyer, Simon A. A. Kohl, Andrew J. Ballard, Andrew Cowie, Bernardino Romera-Paredes, Stanislav Nikolov, Rishub Jain, Jonas Adler, Trevor Back, Stig Petersen, David Reiman, Ellen Clancy, Michal Zielinski, Martin Steinegger, Michalina Pacholska, Tamas Berghammer, Sebastian Bodenstein, David Silver, Oriol Vinyals, Andrew W. Senior, Koray Kavukcuoglu, Pushmeet Kohli, and Demis Hassabis. Highly accurate protein structure prediction with AlphaFold. *Nature*, 596(7873):583–589, 2021.

[10] T. Miyazawa and E. R. Blout. The Infrared Spectra of Polypeptides in Various Conformations: Amide I and II Bands. *Journal of the American Chemical Society*, 83(3):712–719, 1961.

[11] Mitsuaki Narita, Toshihiko Ogura, Kazuhiro Sato, and Shinya Honda. Design of the Synthetic Route for Peptides and Proteins Based on the Solubility Prediction Method. I. Synthesis and Solubility Properties of Human Proinsulin C-Peptide Fragments. *Bulletin of the Chemical Society of Japan*, 59(8):2433–2438, 1986.

[12] Marta Paradís-Bas, Judit Tulla-Puche, and Fernando Albericio. The road to the synthesis of "difficult peptides". *Chemical Society Reviews*, 45(3):631–654, 2016.

[13] Héloïse Bürgisser, Elyse T. Williams, Aliénor Jeandin, Robin Lescure, Adhvitha Premanand, Songlin Wang, and Nina Hartrampf. A Versatile "Synthesis Tag" (SynTag)

for the Chemical Synthesis of Aggregating Peptides and Proteins. *Journal of the American Chemical Society*, 146(50):34887–34899, 2024.

[14] R. C. de L. Milton, Saskia C. F. Milton, and Paul A. Adams. Prediction of difficult sequences in solid-phase peptide synthesis. *Journal of the American Chemical Society*, 112(16):6039–6046, 1990.

[15] Stephen B. H. Kent. Chemical Synthesis of Peptides and Proteins. *Annual Review of Biochemistry*, 57:957–989, 1988.

[16] Joy Bedford, Carolyn Hyde, T. Johnson, Wen Jun, D. Owen, M. Quibell, and R.c. Sheppard. Amino acid structure and "difficult sequences" in solid phase peptide synthesis. *International Journal of Peptide and Protein Research*, 40(3-4):300–307, 1992.

[17] P. Y. Chou and G. D. Fasman. Conformational parameters for amino acids in helical, beta-sheet, and random coil regions calculated from proteins. *Biochemistry*, 13(2):211–222, 1974.

[18] P. Y. Chou and G. D. Fasman. Prediction of protein conformation. *Biochemistry*, 13(2):222–245, 1974.

[19] Somesh Mohapatra, Nina Hartrampf, Mackenzie Poskus, Andrei Loas, Rafael Gómez-Bombarelli, and Bradley L. Pentelute. Deep Learning for Prediction and Optimization of Fast-Flow Peptide Synthesis. *ACS Central Science*, 6(12):2277–2286, 2020.

[20] Bálint Tamás, Pietro Luigi Willi, Héloïse Bürgisser, and Nina Hartrampf. A robust data analytical method to investigate sequence dependence in flow-based peptide synthesis. *Reaction Chemistry & Engineering*, 9(4):825–832, 2024.

[21] N. Hartrampf, A. Saebi, M. Poskus, Z. P. Gates, A. J. Callahan, A. E. Cowfer, S. Hanna, S. Antilla, C. K. Schissel, A. J. Quartararo, X. Ye, A. J. Mijalis, M. D. Simon, A. Loas, S. Liu, C. Jessen, T. E. Nielsen, and B. L. Pentelute. Synthesis of proteins by automated flow chemistry. *Science*, 368(6494):980–987, 2020.

[22] Monica Dettin, Stefano Pegoraro, Paolo Rovero, Silvio Bicciato, Andrea Bagno, and Carlo Di Bello. SPPS of difficult sequences. *The Journal of Peptide Research*, 49(1):103–111, 1997.

[23] Alexander J. Mijalis, Dale A. Thomas, Mark D. Simon, Andrea Adamo, Ryan Beaumont, Klavs F. Jensen, and Bradley L. Pentelute. A fully automated flow-based approach for accelerated peptide synthesis. *Nature Chemical Biology*, 13(5):464–466, 2017.

[24] Dan Ofer, Nadav Brandes, and Michal Linial. The language of proteins: NLP, machine learning & protein sequences. *Computational and Structural Biotechnology Journal*, 19:1750–1758, 2021.

[25] Jia-Ying Chen, Jing-Fu Wang, Yue Hu, Xin-Hui Li, Yu-Rong Qian, and Chao-Lin Song. Evaluating the advancements in protein language models for encoding strategies in protein function prediction: a comprehensive review. *Frontiers in Bioengineering and Biotechnology*, 13, 2025.

[26] Zeming Lin, Halil Akin, Roshan Rao, Brian Hie, Zhongkai Zhu, Wenting Lu, Nikita Smetanin, Robert Verkuil, Ori Kabeli, Yaniv Shmueli, Allan dos Santos Costa, Maryam Fazel-Zarandi, Tom Sercu, Salvatore Candido, and Alexander Rives. Evolutionary-scale prediction of atomic-level protein structure with a language model. *Science*, 379(6637):1123–1130, 2023.

[27] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding, 2019. arXiv:1810.04805.

[28] Leo Breiman. Random Forests. *Machine Learning*, 45(1):5–32, 2001.

[29] Tianqi Chen and Carlos Guestrin. XGBoost: A Scalable Tree Boosting System, 2016. arXiv:1603.02754.

[30] V. N. Rajasekharan Pillai and Manfred Mutter. Conformational studies of poly(oxyethylene)-bound peptides and protein sequences. *Accounts of Chemical Research*, 14(4):122–130, 1981.

[31] Scott Lundberg and Su-In Lee. A Unified Approach to Interpreting Model Predictions, 2017. arXiv:1705.07874.

[32] Thomas Haack and Manfred Mutter. Serine derived oxazolidines as secondary structure disrupting, solubilizing building blocks in peptide synthesis. *Tetrahedron Letters*, 33(12):1589–1592, 1992.

[33] James S. Nowick and Shabana Insaf. The Propensities of Amino Acids To Form Parallel β-Sheets. *Journal of the American Chemical Society*, 119(45):10903–10908, 1997.

[34] Jiaqi Wang, Zihan Liu, Shuang Zhao, Tengyan Xu, Stan Z. Li, and Wenbin Li. Aggregation Rules of Short Peptides. *JACS Au*, (9):3567 – 3580, 2024.

[35] H. L. Morgan. The Generation of a Unique Machine Description for Chemical Structures-A Technique Developed at Chemical Abstracts Service. *Journal of Chemical Documentation*, 5(2):107–113, 1965.

[36] Markus Löning, Anthony Bagnall, Sajaysurya Ganesh, Viktor Kazakov, Jason Lines, and Franz J. Király. sktime: A Unified Interface for Machine Learning with Time Series, 2019. arXiv:1909.07872.

[37] Matthew Middlehurst, James Large, Michael Flynn, Jason Lines, Aaron Bostrom, and Anthony Bagnall. HIVE-COTE 2.0: a new meta ensemble for time series classification, 2021. arXiv:2104.07551.

[38] Patrick Schäfer and Ulf Leser. WEASEL 2.0 – A Random Dilated Dictionary Transform for Fast, Accurate and Memory Constrained Time Series Classification, 2023. arXiv:2301.10194.

[39] Houtao Deng, George Runger, Eugene Tuv, and Martyanov Vladimir. A Time Series Forest for Classification and Feature Extraction, 2013. arXiv:1302.2277.

[40] Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Andreas Müller, Joel Nothman, Gilles Louppe, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, Jake Vanderplas, Alexandre Passos, David Cournapeau, Matthieu Brucher, Matthieu Perrot, and Édouard Duchesnay. Scikit-learn: Machine Learning in Python, 2018. arXiv:1201.0490.