Exploring protein sequence and functional spaces using adversarial autoencoder.

Tristan Bitard-Feildel
email: tristan@bitardfeildel.fr

February 5, 2021

Abstract

Motivation: Shedding light on the relationship between protein sequences and functions is a challenging task with many implications in protein evolution, diseases, or protein design. Protein sequence / function space is however hard to comprehend due to its complexity. Generative models help to decipher complex systems thanks to their abilities to learn and recreate data specificity. Applied to protein sequences, they can point out relationships between protein positions and functions or capture the sequence patterns associated to functions.
Results: In this study, an unsupervised generative approach based on auto-encoder (AE) is proposed to generate and explore new sequences with respect to their functions. AEs are tested on three protein families known for their multiple functions. Clustering on the encoded sequences from the latent space computed by AEs display high level of homogeneity regarding the protein sequence functions. Furthermore, arithmetic and interpolations in latent space generate new intermediate protein sequences sharing sequential and functional properties of original sequences issued from different families and functions. Sequences generated by interpolation between latent space data points show the ability of the AE to generalize and to produce meaningful biological sequences from an evolutionary uncharted area of the biological sequence space. Finally, from structural homology modeling and assessment, it can be observed that the sequences generated using latent space arithmetic can display similar physico-chemical properties and folding energies when compared to structures generated from the original sequences of the families.
Availability: Code and data used for this study are freely available at https://github.com/T-B-F/aae4seq.
Contact: tristan@bitardfeildel.fr
Supplementary information: Supplementary data are available at online.

1 Introduction

Protein sequence diversity is the result of a long evolutionary process. This diversity, or sequence space, is constrained by natural selection and only a fraction of amino acid combinations out of all possible combinations are observed. Given its huge size, exploring the sequence space and understanding its hidden

constrains is highly challenging. Protein families are groups of related proteins, or part of proteins, and represent a useful description to reduce the sequence space complexity. Many resources have been developed over the years to group protein sequences into families with members sharing evidence of sequence similarity or structural similarity [10, 13, 25]. However, diversity also exists inside a family and one family may group together several proteins with different molecular functions [9]. Navigating the sequence space with respect to the functional diversity of a family is therefore a difficult task whose complexity is even increased by the low number of proteins with experimentally confirmed function. In this regard, computer models are needed to explore the relationships between sequence space and functional space of the protein families [6, 27, 33, 34, 42].

Understanding the relationships between amino acids responsible for a particular molecular function has a lot of implications in molecular engineering, functional annotation, and evolutionary biology. In this study, an unsupervised deep learning approach is proposed to model and generate protein sequences.

Previous deep learning generative models such as variational autoencoders (VAE) have been applied on biological and chemical data to explore and classify gene expression in single-cell transcriptomics data [21], to predict the impact of mutations on protein activity [31, 40] or to explore the chemical space of small molecules for drug discovery and design [17, 30]. Their ability to reduce input data complexity in a latent space and to perform inference on this reduced representation make them highly suitable to model, in an unsupervised manner, complex systems. Regarding protein science, VAE have been able to accurately models protein sequence and functional space [40], predict mutational impact [18], or design new protein [15]. In this study, Adversarial AutoEncoder (AAE) network [22] is proposed as a new and efficient way to represent and navigate the functional space of a protein family. AAE networks also have the advantage to constrain the latent space over a prior distribution which allow sampling strategies to explore the whole latent distribution.

Like VAE and other autoencoder architectures, AAEs reduce high dimensional data by projection, using an encoder, to a lower dimensional space. This space is known as a latent space or embedding representation which in turns can be reconstructed by the decoder. AAE [22] architecture corresponds to a probabilistic autoencoder, but with a constraint on the latent space of the encoder which follows a defined prior distribution. This constraint is performed using an generative adversarial network (GAN) [14] between the latent space and the prior distribution. Unlike VAE, this constrain ensures that meaningful samples can be generated from anywhere in the latent space defined by the prior distribution. Therefore, applied to biological sequences of a protein family, it is then possible to encode the sequence diversity to any prior distribution and thus, to sample and generate new protein sequence of the family from any point of the prior distribution. Ideally, the learned latent space should be representative to the functions of considered the protein family.

The clustering induced by latent space of AAE network were analyzed to verify their ability to cluster sequences according to function as observed with VAE networks. Three protein families with sub-families were used to train AAE models and to analyze the protein functional clustering of the proposed model. The three different protein families selected were the sulfatases, the

HUP and the TPP families. The sulfatases are a group of proteins acting on sulfated biomolecules. This family have been manually curated into sub-family with specific functions such as substrate specificity [2]. This group of proteins is found in various protein family databases, such as in Pfam (PF00884, Sulfatases). However, as mentioned previously, they can have different substrate specificity despite being in the same family. The SulfAtlas database [2] is a collection of curated structurally-related sulfatases centered on the classification of the substrate specificity. The majority of sulfatases (30,726 over 35,090 Version 1.1 September 2017) is found in family S1 and is sub-divided into 73 sub-families corresponding to different substrate specificities. Sub-families S1_0 to S1_12 possess experimentally characterized EC identifiers. The two other protein families, HUP and TPP families are not manually curated but were selected as they are known to have multiple functions [9]. Sampling of the latent space was therefore analyzed, with a particular focus on latent space arithmetic between proteins of different sub-families with different functions to validate the ability of the model to learn a meaningful representation of the sequence space.

VAE have often only be used to cluster and to interpret clusters of protein sequences regarding their function or evolutionary history but few experiments have been performed on their generative ability and particularly when it comes to feature transfer. Two experiments were carried in this direction using latent space interpolation and latent space arithmetic operations.

The latent space coverage of the protein family functional space was analyzed using data point interpolations between protein sequences of different sulfatase sub-families encoded in the latent space. The interpolated data points correspond to unseen proteins. A good model should be able to produce realistic protein sequences from these data points.

To go beyond the functional clustering induced by generative models such as VAEs or AAEs, this study also explores arithmetic operations with protein sequences encoded in their latent space to generate new protein sequences. Arithmetic operations on latent space have previously been reported to transfer features between images of different classes [29] and may therefore have interesting potential for molecular design. Four different strategies are explored to combine latent spaces of different sulfatase sub-families. The proteins generated from the combined latent spaces were analysed in term of sequences and structures, after being modelled by homology.

2 Methods

2.1 Protein Families

The sulfatase family. An initial seed protein multiple sequence alignment was computed from sequences of the protein structures of SulfAtlas [2] database sub-families 1 to 12. This seed was used to search for homologous sequences on the UniRef90 [41] protein sequence database using hmmsearch [12] with reporting and inclusion e-values set at $1e^{-3}$.
A label was assigned to each retrieved protein if the protein belonged to one of the 12 sub-families. The multiple sequence alignment (MSA) computed by hmmsearch was filtered to remove columns and sequences with more than 90%

and 75% gap character respectively. Proteins with multiple hits on different parts of their protein sequences were also merged into a single entry. From 105181 initial protein sequences retrieved by hmmsearch, the filtering steps led to a final set of 41901 proteins.

HUP and TPP protein families. A similar protocol was followed for the HUP and TPP protein families. Instead of using an initial seed alignment made of sequences of protein structures, the CATH protein domain HMM [24, 39] was used to search for homologous sequences in the UniRef90 database. CATH model 3.40.50.620 corresponds to the HUP protein family and model 3.40.50.970 corresponds to the TPP protein family. A sequence filtering pipeline identical to the one used for the sulfatase family was applied to each of the resulting multiple sequence alignments.
The final number of proteins in each dataset was: 25041 for the HUP family (32590 proteins before filtering) and 33693 for the TPP family (133701 before filtering).

2.2 Deep learning model

Generative Adversarial Network. A complete description of Generative Adversarial Network can be found in Goodfellow *et al.* [14]. To summarize, the GAN framework corresponds to a min-max adversarial game between two neural networks: a generator (G) and a discriminator (D). The discriminator computes the probability that an input x corresponds to a real point in the data space rather than coming from a sampling of the generator. Concurrently, the generator maps samples z from prior $p(z)$ to the data space with the objective to confuse the discriminator. This game between the generator and discriminator can be expressed as:

$$min_G \ max_D \ E_{x \sim p_{data}}[\log D(x)] + E_{z \sim p(z)}[log(1 - D(G(z))] \tag{1}$$

Adversarial auto-encoder. Adversarial autoencoders were introduced by Makhzani *et al.* [22]. The proposed model is constructed using an encoder and a decoder networks, and a GAN network to match the posterior distribution of the encoded vector with an arbitrary prior distribution. Thus, the decoder of the AAE learns from the full space of the prior distribution. The model used in this study to compute the aggregated posterior $q(z|x)$ (the encoding distribution) uses a Gaussian prior distribution. The mean and variance of this distribution is predicted by the encoder network: $z_i \sim N(\mu_i(x), \sigma_i(x))$. The re-parameterization trick introduced by Kingma and Welling [20] is used for back-propagation through the encoder network.
Three different architectures were evaluated. The general architecture is as follows (see Supplementary Table 1 and Supplementary Figure 1 for a representation of architecture number 3). The encoder comprises one or two 1D convolutional layers with 32 filters of size 7 and a stride of length 2, and one or two densely connected layers of 256 or 512 units. The output of the last layer is passed through two stacked densely connected layers of hidden size units to evaluate μ and σ of the re-parameterization trick [20].
The decoder is made of two or three densely connected layers of the length of the sequence family time alphabet units for the last layers and of 256 or 512

units for the first or the two first layers. The final output of the decoder is reshaped and a softmax activation function is applied, corresponding to a probability for each positions associated to each possible amino acids. To convert the probability matrix of the decoder into a sequence, a random sampling according to the probability output was performed at each position. The selected amino acid at a given position is therefore not necessarily the amino acid with the highest probability. The discriminator network is made of two or three densely connected layers, the last layer has only one unit and corresponds to the discriminator classification decision through a sigmoid activation function, the first or the first two layers are made of 256 or 512 units.
The network was trained for each protein family independently. The autoencoder was trained using a categorical cross-entropy loss function between the input data and the predicted sequences by the autoencoder. The discriminator was trained using binary cross-entropy loss function between the input data encoded and the samples from the prior distribution.

2.3 Generated sequences and structures analyses

Dimensionality reduction. The AAE model can be used to reduce the dimensionality of the sequence space by setting a small latent size. Two dimensionality reductions were tested with latent size of 2 and 100. Latent size of 2 can be easily visualized and a larger latent size of 100 should represent the input data more efficiently as more information can be stored.

Clustering. HDBSCAN [5, 23] was used to cluster the sequences in the latent space due to its capacity to handle clusters of different sizes and densities and its performances in high dimensional space. The Euclidean distance metric was used to compute distances between points of the latent space. A minimal cluster size of 60 was set to consider a group as a cluster as the number of protein sequences is rather large. The minimal number of samples in a neighborhood to consider a point as a core point was set to 15 to maintain relatively conservative clusters.

Functional and taxonomic analyses. Enzyme functional annotation (EC ids) and NCBI taxonomic identifiers were extracted when available from the Gene Ontology Annotation portal (January 2019) using the UniProt-GOA mapping [19]. Protein without annotation were not taken into account in these analyses.
The annotation homogeneity was computed for each cluster. Considering a cluster, the number of different EC ids and taxonomic ids were retrieved. For each different EC ids (taxonomic ids) its percentage in the cluster was computed. An EC id (taxonomic id) of a cluster with a value of 90% indicates that 90% of the cluster members have this EC id (taxonomic id). A cluster with high homogeneity values correspond to functionally or evolutionary related sequences. Homogeneous clusters computed from the AAE encoded space will therefore indicates the ability of the AAE model to capture and to distinguish protein sequences with functionally or evolutionary relevant features without supervision.

Latent space interpolation. Twenty pairs of protein sequences were randomly chosen between Sulfatases sub-families with at least 100 labeled members but with less than 1000 members (to avoid pronounced imbalance between classes) were selected: S1-0 (308 proteins), S1-2 (462 proteins), S1-3 (186 proteins), S1-7 (741 proteins), S1-8 (290 proteins) and S1-11 (669 proteins). The coordinates of the selected sequences in the encoded space with 100 dimensions were retrieved and spherical interpolation using 50 steps were performed for each of the pairs. Spherical interpolation has previously been reported to provide better interpolation for the generation of images [45]. The interpolated points were given to the decoder and new sequences were generated. Statistical analyses were carried using the generated sequences based on their dynamical changes from one family to an other to study the model ability to generalize. Analyses at the amino acid level were also performed on the interpolated sequences of two Sulfatase sub-families encoded far from one-another in the latent space.

Latent space arithmetic. Subtraction and addition of latent spaces have been shown to be able to transfer features specific to some sub-groups of data to other sub-groups. This property was tested on seven previously selected Sulfatase sub-families (S1-0, S1-1, S1-2, S1-3, S1-7, S1-8 and S1-11) based on their number of protein sequences. Different arithmetic strategies (Supplementary Figure 2) were tested between latent spaces of a query sub-family and a different source sub-family with the aim to transfer features of the source sub-family to the query sub-family.
A first strategy consists in adding the mean latent space, computed using the encoder on the sequences of the source sub-family, to the encoded sequences of the query sub-family. The second strategy differs from the first one by subtracting the mean background latent space, computed from the latent space of all sub-families, from the latent space of the query sub-family. The third strategy differs from the second as the background strategy is computed using all sub-families except source and query sub-families. Finally, in the fourth strategy, the subtraction is performed using a local KD-tree to only remove features shared by the closest members of a given query and the addition is performed by randomly selecting a member of the source family and its closest 10 members.
For each strategy, new sequences were generated using the latent spaces of all query proteins in the sub-families. Thus, for one original encoded protein sequences there is a direct correspondence between the original amino acid sequence and the amino acid sequences generated with the different strategies and different source sub-families. The sequences generated by latent space arithmetic are compared to the initial sub-families in terms of sequence and structural constraints.
To evaluate the generated sequences by latent space arithmetic, the sequences were compared to themselves and to the biological sequences of the two initial sub-families using protein sequence similarity computed with a Blosum 62 substitution matrix. The protein sequence similarities inside a sub-family and between sub-families are also computed. The distributions of sequence similarities allow to explore the ability of the latent space arithmetic operations and of the decoder to produce meaningful intermediate protein sequences from unexplored encoded data points.

Protein structural models are computed using the structures of the initial sub-families as template for MODELLER [44] and evaluated using the DOPE score [37]. Models are computed using the generated sequences by latent space arithmetic on template structures of their source and query sub-families. The DOPE energies of the modeled structures are compared to structural models computed as references. The first references are structural models computed using the sequences and template structures of the same sub-families, which should provide the best DOPE energies (best case scenario). The second references are structural models computed using the sequences and template structure of different sub-families (ex: sequences from source sub-family and template structures from the query sub-family or inversely, sequences from query sub-family and template structures from the source sub-family), which should provide the worst DOPE energy (worst case scenario). If the generated sequences by latent space arithmetic correspond to intermediate proteins with properties from two sub-families, they should have intermediate DOPE energies when compared to the best and worst case scenarii.

3 Results

A structurally constrained MSA was computed using Expresso from T-Coffee webserver [1, 11] between sequences of S1 sulfatases structures. This MSA was processed into a Hidden Markov Model and hmmsearch was used to retrieve aligned sequence matches against the UniRef90 sequence database. A total of 76,427 protein sequence hits were found to be matching UniRef90. The sequences were filtered to remove columns and hits with more than 90% and 75%, respectively, of gap characters. The final MSA comprised 41,901 sequences. The Sulfatases protein dataset was separated in a training, validation, and test sets with a split ratio of: 0.8, 0.1, and 0.1.

The three different AAE architecture (see Method section) were trained on the training set and evaluated on the validation set. The test set was only used on the final selected architecture. Models were evaluated by computing top k-accuracy, corresponding to the generation of the correct amino acid in the first k amino acids. Supplementary Table 2 shows the top k accuracy metric for k=1 and k=3 computed for the different autoencoders. The accuracy scores scaled down with the number of parameters, but without any large differences. To avoid over-fitting, the architecture with the fewest number of parameters (architecture 3) was therefore selected. The final accuracy scores on the test set were computed and were similar to the values observed during the model training: 62.5% and 80.2% (k=1 and k=3). The selected architecture was separately trained on the protein sequences of the HUP and TPP families.

3.1 Latent space projection

AAE can be used as a dimensional reduction and as a visualization techniques by fixing the dimension of the latent space to two or three for plotting purpose. In this section, AAE network ability to create meaningful projection is tested on Sulfatase, HUP and TPP families by clustering and analysing protein sequences

in term of enzymatic activity and phylogenetic diversity.

Starting from the final MSA of the Sulfatase family, an AAE network was trained to project the sequences in a latent space with two dimensions. For comparison, the MSA projection using the first two components of the PCA decomposition was also computed.

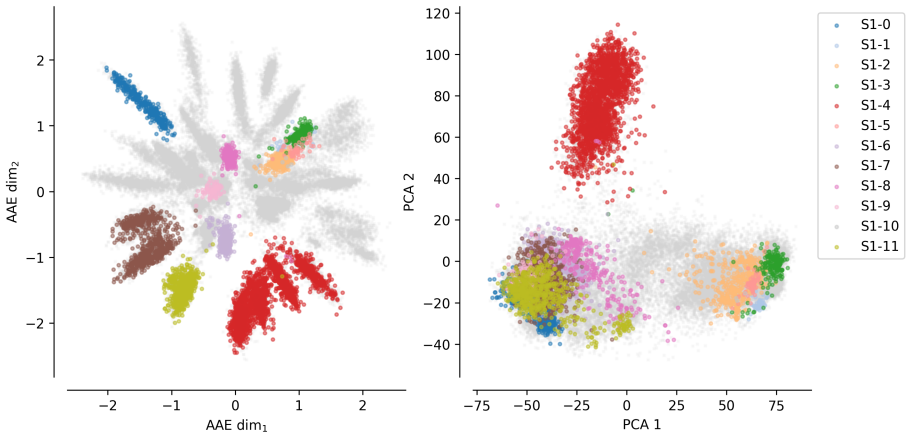

Figure 1: Sequences of the SulfAtlas MSA projected using the encoding learned using an AAE (number of latent dimension: 2) and a PCA (two first components). Gray data points correspond to protein sequences not part of the curated 12 sub-families.

Figure 1 shows the protein sequences encoded by the AAE in two latent dimensions and the PCA projections. Each dot corresponds to a protein sequence, and the dots are colored according to their sub-family. Gray dots correspond to protein sequences not belonging to any of the 12 curated sulfatases sub-families. In this figure, the AAE displays a higher capacity to disentangle the sequence and functional spaces of the S1 family than a PCA. Interestingly, it can be observed in the AAE projection some well-separated gray spikes (sequences not belonging to any curated sub-family). These spikes may correspond to groups of enzymes sharing common substrate specificity.

In some cases, sub-families with identical functions are projected closely on the encoded space. For instance, sub-families S1_6 (light magenta) and S1_11 (yellow) have both the EC 3.1.6.14 activity (N-acetylglucosamine-6-sulfatase) and are closely located in the encoded space. Moreover, some sub-family projections appear entangled such as the S1-1 sub-family (light blue, Cerebroside sulfatase activity, EC 3.1.6.8), the S1-2 (orange) and the S1-3 (green) sub-families (Steryl-sulfatase activity, EC 3.1.6.2), the S1-5 (pink) sub-family (N-acetylgalactosamine-6-sulfatase activity, EC 3.1.6.4), and the S1-10 (gray) sub-family (Glucosinolates sulfatase activity EC 3.1.6.-). The five families correspond to four different functions but are made of Eukaryotic protein sequences only and their entanglement may be due to their shared common evolutionary history. This separation based on the sequence kingdoms can clearly be visualized in the PCA projections with Eukaryotic sequences on the right side on sub-families with a majority of Bacteria sequences on the left side. The example of protein B6QLZ0_PENMQ is also interesting. The protein is projected (yellow

dot corresponding to the S1-11 sub-family) at coordinates (0.733, -1.289), inside the space of the S1-4 family (red). This may look like an error by closer inspection shows that this protein is part of both the S1-4 and S1-11 sub-families of the SulfAtlas database.

Projections of sequences into latent spaces using AAE with two dimensions were also tested on the HUP and TPP families. The AAE projections can be visualized on Supplementary Figure 3. There are fewer functional annotations for these two families, but a strong separation between the major functions of the families can clearly be seen .

Latent spaces were evaluated for each protein family based on enzyme classification (EC) and taxonomic homogeneity. Given a set of protein sequences, the encoded sequences in a latent space of a 100 dimensions were clustered using HDBSCAN.
For the sulfatase family, 27 clusters were found, for which taxonomic and EC annotations could be extracted (Supplementary Table 3). All these clusters displayed either strong taxonomic or EC homogeneity. Enzymatic homogeneity was higher than taxonomic homogeneity for 16 clusters, found equal in one cluster and lower for 10 clusters.
In the HUP family, all clusters had very high EC homogeneity (Supplementary table 4). Only two clusters out of 47 could be found with higher taxonomic homogeneity than EC homogeneity and for this two clusters enzymatic homogeneity values were high and only marginally different (cluster 5, taxonomic homogeneity of 100% an EC homogeneity of 99% and cluster 31, taxonomic homogeneity of 99 % and EC homogeneity of 97%). Five clusters were found with equal taxonomic and EC homogeneity.
Similarly, in the TPP family, all clusters had also very high EC homogeneity (Supplementary table 5). Five clusters out of 51 could be found with higher taxonomic homogeneity than EC homogeneity. For these 5 clusters the differences between taxonomic homogeneity and EC homogeneity were higher than the differences observed for the HUP clusters. Six clusters were found with equal taxonomic and EC homogeneity.
These results highlight the ability of the encoding space to capture amino acid functional properties instead of signal due to taxonomic relationships.

3.2 Protein latent space interpolation

Interpolation between encoded sequences can be used to "navigate" between proteins of two sub-families. For this task, twenty pairs of protein sequences were randomly selected between all combinations of protein sub-families to test the capacity of the encoded space and 50 intermediates data points were generated between each pair. At each step the sequence similarities were computed between the generated protein sequences from the interpolated latent space and the query and target protein sequences of the sub-families. It is thus possible to measure the amino acid sequence drift from one protein to another one.

The observed amino acid transition from the query sub-family to the target sub-family is very smooth for all combinations of sub-families and has a logistic function shape as shown in Figure 2-A. The smooth transition between points point out to the ability of AAE network to encode the sequences in smooth

latent space and thus to "fill" the gap between protein sequence sub-families.

The Shannon entropy was computed for each group of sequences: interpolated sequences between query and target sub-families, sequences of the query sub-families and sequences of the target sub-families. Figure 2-B shows the Shannon entropy distribution for the S1-0 and S1-11 sequences and their interpolated sequences. Interestingly, the figure shows lower entropy for interpolated sequences than for original sequences. This trend is true for all interpolated sequences between all sub-families as reported in the Supplementary Table 6. Lower entropy indicates fewer variation in the interpolated amino acid than in biological sequences which point out to restricted paths between sub-families. This is in agreement with molecular evolution theory and experiments that describe protein families as bassin in fitness landscape [3, 4, 38].

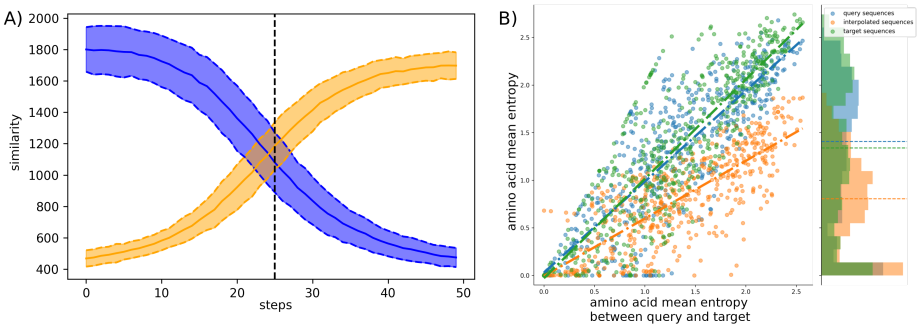

Figure 2: Interpolation analyses between sub-families S1-0 and S1-11. A) Sequence similarity distributions between interpolated sequences and query proteins (blue) or target proteins (orange). B) Distribution of amino acid Shannon entropy for interpolated sequences (orange) between sub-families S1-0 (blue) and S1-11 (green) over the amino acid mean Shannon entropy of query and target sub-families.

A closer inspection of interpolations between sub-families S1-0 and S1-4 (respectively blue and red data points in figure 1) were also performed to study changes at the amino acid level as the two sub-families are in opposite spaces in the two dimensional projection. It can be observed in Supplementary Figure 5 that gaped area found in the query sequence but not in the target sequence (and inversely) are progressively filled (or broken down) starting from the flanking amino acids to the center of the gap (or inversely from the center to the flanking amino acids). This indicates an organized and progressive accumulation of amino acids (or gaps) that extend (or shrink) the region of the sequence previously without (or with) residues. For instance gap reduction can be observed in the generated sequences between sequence ID 2 of the sulfatase S1-0 family (query) and sequence ID 2196 of the sulfatase S1-4 family (target) at positions 75 to 86.

Moreover, family-specific amino acids family are progressively replaced in key positions. In the previous interpolation between query and target sequences, it can thus be observed at positions 21 and 22 of the MSA a replacement of residues S and C by G and A (Supplementary Figure 5). Most transitions are

not abrupt, and do not occur at the 25th-generated intermediate sequences but are smooth and correspond to plausible sequences. The ability of the AAE to generate interpolated sequences with emerging or disappearing features of two sub-families, highlights its capacity to generalize the decoding of latent space points not corresponding to encoded sequences and thus never observed during training, and outside the structured organization of the computed latent space.

3.3 Protein latent space arithmetic

It has been shown that latent space arithmetic was able to transfer learned features between different classes. In this work, this ability has been evaluated in the case of protein sequences to transfer features of a first sub-family to a second sub-family while maintaining the general property (such as its structure) of the second sub-family. The Sulfatases sub-families S1-0, S1-2, S1-3, S1-7, S1-8 and S1-11 were chosen to test this hypothesis.
Different arithmetic strategies (see Methods and Supplementary Figure 2) were tested between latent spaces of a query sub-family and a different source sub-family with the aim to transfer features of the source sub-family to the query sub-family.
In the following section, the terminology $\widehat{\text{Seq}}$. S1-XmY will correspond to a sequence generated using a combination of latent space of the sub-family S1-Y added to the latent space of the sub-family S1-X. The X and Y sub-families will be referred to as the query and source sub-families. The source latent space is obtained using the mean value of the latent space corresponding to the sequences of the sub-family

Supplementary Figure 6 displays logo plots of two regions corresponding to Prosite motifs PS00523 and PS00149 to illustrate the amino acid content of the protein sequences generated by latent space arithmetic. These regions correspond to the most conserved regions of the sulfatase family and have been proposed as signature patterns for all the sulfatases in the Prosite database.

Different amino acid patterns can be observed between the sequence groups that can be classified as "competition", "taking over", or "balanced" pattern. A competition pattern of amino acids corresponds to equivalent frequency of two different amino acids in the generated sequences. A taking over pattern corresponds to an amino of one of the original sequences being the most frequent in the generated sequences. Finally, a balanced pattern corresponds to a maintained equilibrium between amino acids in the generated sequences. Other positions are however displaying much more complex patterns and cannot be summarized as a frequency competition between source and query sub-families. These behaviors can be observed several times through the logo plots but are still position-specific, meaning that the bits scores pattern observed in the source sub-families (Panels A and D) do not necessary allow to predict the amino acids bits scores in the generated sequences (Panels B and C).

Protein sequence similarities were computed to evaluate the diversity of the generated sequences and to compare their diversity with the original sub-families. Protein sequence similarities were computed between:

- sequences of a group of generated sequences,

- sequences of a sulfatase sub-family used to generate protein sequences,
- generated sequences and their query sulfatase sub-family,
- generated sequences and their source sulfatase sub-family,
- query and source sequences of sulfatase sub-families.

Figure 3 shows the mean and variance distribution of computed protein sequence similarities between these different groups for generated sequences computed using the first strategy. The first, second, and third strategies display a similar pattern and their corresponding figures are available in the Supplementary Information (Supplementary Figures 10, 12 and 14).

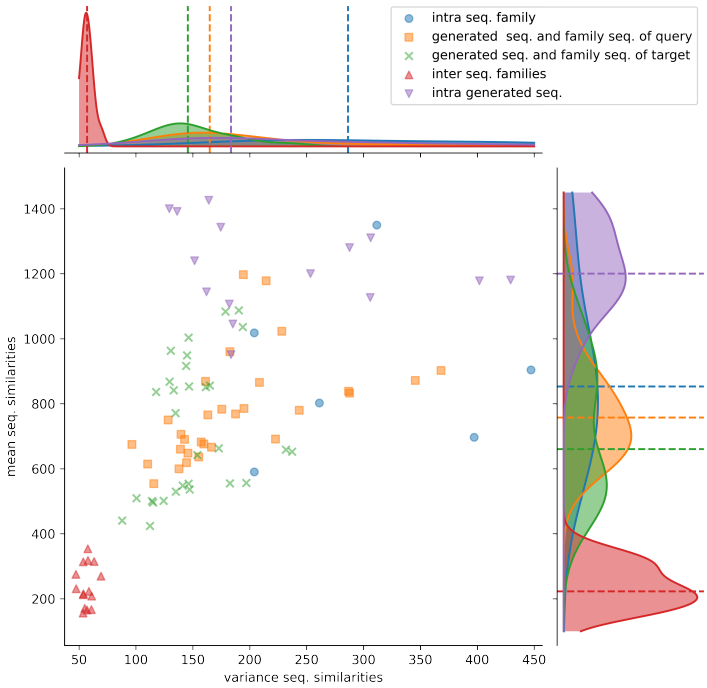

Figure 3: **Distributions of protein sequence similarities**. Blue dots: protein sequence similarity computed between sequences of the same protein sub-family. Orange squares: similarity computed between generated sequences and the sequences of their query sub-family (ex: S1-0m2 generated sequences and S1-0 sub-family sequences). Green x: similarity computed between generated sequences and the sequences of their target sub-family (ex: S1-0m2 generated sequences and S1-2 sub-family sequences). Red upper triangles: similarity computed between sequences of two different sub-families (ex: S1-0 sequences and S1-2 sequences). Magenta lower triangles: similarity computed between sequences of the same generated sequence group. The variance and the mean of each distribution are displayed on the horizontal and vertical axes.

Protein sequence similarities between different sub-families have lower similarity scores and lower variances than the other distributions. Protein sequence

similarities between sequences of a sub-family have the highest mean and variance values observed. However, since only 6 sub-families were kept for analysis (sub-families 0, 2, 3, 7, 8, and 11), trends must therefore be taken with precaution. Generated protein sequences compared to themselves have mean and variance protein sequence similarities higher than when compared to their query or sub-families. The last two have mean and variance values spread between the blue and red distributions.
These distributions indicate that protein sequences generated by latent space arithmetic have an intrinsic diversity similar to the biological sub-families. Moreover, the generated sequences are less similar to the sequences from their query and source sub-families than to themselves. The generated sequences are also globally as similar to the sequences of their query sub-family as to the sequence of their source sub-family. The generation process is therefore able to capture the features of the selected query and source sub-families and to generate a protein sequence diversity similar to the original sub-families.

Finally, protein structural modeling was performed to assess and compare the properties of the generated sequences by latent space arithmetic and the protein sequences of the natural sub-families. For each sub-family, 100 original sequences were randomly selected along the corresponding generated sequences. All the generated sequences were aligned to protein structures of their corresponding source and query sub-families, and the alignments were used to create structural models by homology. The models were evaluated with the DOPE function of MODELLER.

Supplementary Figure 7 shows an example of the energy distribution computed from models using the second strategy with query sub-family S1-0 and source sub-family S1-2. The lowest energies (best models) were found on modelled structures using the original protein sequences of a sub-family to the structural templates of the same sub-family (*Struct. 0 Seq. 0* and *Struct. 2 Seq. 2*). Inversely, the highest energies are found on modelled structures using the original protein sequences of a sub-family to the structural templates of another sub-family (*Struct. 0 Seq. 2* and *Struct. 2 Seq. 0*). Interestingly, sequences generated using addition and subtractions of latent spaces have intermediate energy distributions. This can be clearly observed in Figure 4, where generated sequences are mostly situated between the two dotted lines. Dots on the right side of the vertical line at 0 correspond to modeled structures using sequences of the latent space with lower energy than the modeled structures using sequences from their original sub-family. Dots on the left side of the vertical line at 0 are modeled structures using sequences of the latent space with higher energy than the modeled structures using sequences from their original sub-family. The diagonal line on the top-left corner corresponds to the difference in energy between modeled structures using sequences from their original sub-family and modeled structures using sequences from biological sequences of another sub-family. Generated sequences modeled on structures belonging to the same sub-family than their query latent space sub-family (ex: *Struct. 0 Seq.S1-0m2* and *Struct. 2 Seq. S1-2m0* on Supplementary Figure 7 and M_{QS}/Q on Figure 4) have slightly lower energy than when modeled on structures corresponding to the sub-family of their source latent space sub-family (ex: *Struct. 0 Seq. S1-2m0* and *Struct. 2 Seq. S1-0m2* on Supplementary Figure 7 and M_{SQ}/Q on Figure 4). This trend is true for all query / source pairs of sub-families and all strategies except for

sequences generated using the fourth strategy (local background subtraction of query latent space using a KD-tree and the addition of source latent space), see Supplementary Figures 9, 11, 13 and Methods.
In this strategy, the modeled structures using generated sequences do not display energy distributions in-between the energy distributions of the original sequences modeled on structures of the query or of the source sub-families (dotted lines). The energy distribution of generated sequences modeled on structures belonging to the sub-family of their query latent space sub-family (ex: *Struct. 0 Seq.S1-0m2*, blue dots M_{QS}/Q) with the fourth strategy is closer to the energy distribution of the modeled structures using a sequence and a structure template from the same sub-families. The energy distribution of generated sequences modeled on structures corresponding to the sub-family of their source latent space (ex: *Struct. 2 Seq.S1-0m2*, orange dots M_{SQ}/Q) with the fourth strategy is closer to the energy distribution of the modeled structures using a sequence and a structure template from different sub-families. This indicates that the fourth strategy is less robust to latent space arithmetic than the other three strategies. No clear differences could be observed between the first, second, and third strategy.

4 Discussion

In this study, a new framework is presented to analyze and explore the protein sequence space regarding functionality and evolution. Some previous attempts, such as the FunFam database [7, 8], were built upon CATH protein families and trained in a supervised manner to construct a model that is specific of a functionality. Variational Autoencoder (VAE) have previously been reported and used to disentangle complex biological information and for classification tasks (such as single cell gene expression data) [26, 43] or generation of new molecules (such as drugs) [17, 28, 30, 35, 36]. AAEs are able to disentangle the information contained in protein sequences to capture functional and evolutionary specific features and more importantly without supervision. They also have the advantage over VAE to constrain the latent space over a prior distribution which allows sampling strategies to explore the whole latent distribution. It can also be noted that Restricted Boltzmann Machine [42] has also been proposed in a similar task showing very promising results despite the difficulty to train RBM which required Markov chain sampling. A comparison of deep learning architectures for protein sequence family modeling is outside of the scope of this study whose focus is on the ability of one of the generative architecture to model and infer protein sequences.
In this study, AAEs are trained on protein sequence families known to have different sub-functions. The results highlight the ability of AAEs to separate sequences according to functional and taxonomic properties for the three studied family. This emphasized the ability of the AAEs to extracted and encoded features in the latent space which are biological relevant.
Furthermore, and contrary to dimensional reduction techniques, AAE can be used to generate new protein sequences. Latent space arithmetic has been used in image generation tasks with striking ability to produce relevant images with intermediate features. Latent space arithmetic is an interesting concept for protein generation particularly in the context of extracting and transferring features

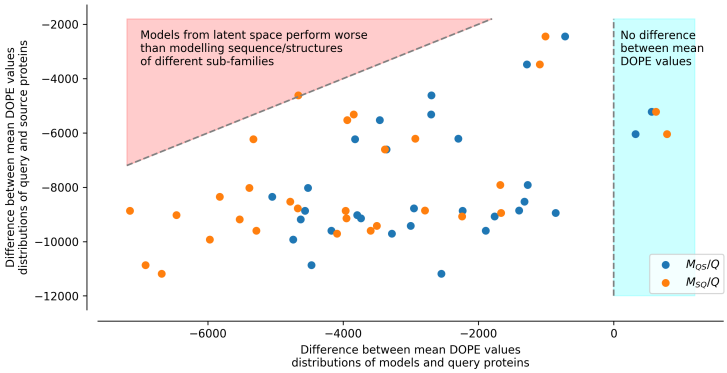

Figure 4: Difference between mean DOPE distributions. Mean value for each distribution, such as the distributions presented in Figure Supplementary 7, were computed. The y axis represents the difference between the mean values computed for query sequences modeled on structures of the same sub-family and mean values computed for source sequences modeled on structures of the query sub-family (ex: differences between mean of Struct. 0 Seq. 0 and mean of Struct. 0 Seq. 2 distributions in Supplementary Figure 7). The x axis corresponds to the difference between the mean values computed for query sequences modeled on structures of the same sub-family and mean values computed for query sequences to which latent spaces of the source sub-family sequences have been added and modeled on structures of the query sub-family (M_{QS}/Q), or source sequences to which latent spaces of the query sub-family sequences have been added and modeled on structures of the source sub-family (M_{SQ}/Q) (ex: differences between mean of Struct. 0 Seq. S1-0m2 and mean of Struct. 0 Seq. 0 distributions Supplmementary in Figure 7).Points in the red area correspond to mean distribution values from generated sequences whose modeled structures have a higher energy than models created using pairs of sequences/structures from different sub-families. Points in the blue area correspond to mean distribution values from generated sequences whose modeled structures have a lower energy than models created using pairs of sequences/structures from the same sub-family.

between protein families. Three out of the four different experiments carried on were able to generate sequences with intermediate features as measured from their protein sequence similarity distributions and modeling energy assessments. Biological experiments will be needed to confirm the functional relevance of the transferred features, but the strategies could have many applications should it be validated.

The absence of measured differences between three out of four strategies used to generate intermediate sequences may also indicate that more optimal approaches could be designed. In this regard, the model architecture could also be improved. Currently, the model used as input is a filtered MSA, but improved architectures could probably benefit from full protein sequences of different sizes without filtering. It is for instance known that some motifs in the TPP and HUP family plays important roles in the family sub-functions [9]. As protein specific motifs, they are not necessary conserved and may not reach filtering thresholds.

Recent advances have been made regarding the application of deep learning for sequential data [16, 32, 46, 47] and transferring these progresses to the field of protein sequence generation could greatly benefit the design of functional proteins.
The results of this study show that AAE, in particular, and deep learning generative models in general, can provide original solutions for protein design and functional exploration.

5 Acknowledgements

TBF's work was suported by ISCD, Sorbonne Université, Paris, France. TBF thanks the NVIDIA society for providing a TitanXp GPU to perform computations.

References

[1] Armougom, F., Moretti, S., Poirot, O., Audic, S., Dumas, P., Schaeli, B., Keduas, V., and Notredame, C. (2006). Expresso: automatic incorporation of structural information in multiple sequence alignments using 3D-Coffee. *Nucleic Acids Res.*, **34**(Web Server issue), W604–608.

[2] Barbeyron, T., Brillet-Gueguen, L., Carre, W., Carriere, C., Caron, C., Czjzek, M., Hoebeke, M., and Michel, G. (2016). Matching the Diversity of Sulfated Biomolecules: Creation of a Classification Database for Sulfatases Reflecting Their Substrate Specificity. *PLoS ONE*, **11**(10), e0164846.

[3] Bornberg-Bauer, E. and Chan, H. S. (1999). Modeling evolutionary landscapes: mutational stability, topology, and superfunnels in sequence space. *Proceedings of the National Academy of Sciences*, **96**(19), 10689–10694.

[4] Boucher, J. I., Cote, P., Flynn, J., Jiang, L., Laban, A., Mishra, P., Roscoe, B. P., and Bolon, D. N. (2014). Viewing protein fitness landscapes through a next-gen lens. *Genetics*, **198**(2), 461–471.

[5] Campello, R. J., Moulavi, D., and Sander, J. (2013). Density-based clustering based on hierarchical density estimates. In *Pacific-Asia conference on knowledge discovery and data mining*, pages 160–172. Springer.

[6] Copp, J. N., Akiva, E., Babbitt, P. C., and Tokuriki, N. (2018). Revealing unexplored sequence-function space using sequence similarity networks. *Biochemistry*, **57**(31), 4651–4662.

[7] Das, S. and Orengo, C. A. (2016). Protein function annotation using protein domain family resources. *Methods*, **93**, 24–34.

[8] Das, S., Sillitoe, I., Lee, D., Lees, J. G., Dawson, N. L., Ward, J., and Orengo, C. A. (2015a). Cath funfhmmer web server: protein functional annotations using functional family assignments. *Nucleic acids research*, **43**(W1), W148–W153.

[9] Das, S., Dawson, N. L., and Orengo, C. A. (2015b). Diversity in protein domain superfamilies. *Curr. Opin. Genet. Dev.*, **35**, 40–49.

[10] Dawson, N. L., Lewis, T. E., Das, S., Lees, J. G., Lee, D., Ashford, P., Orengo, C. A., and Sillitoe, I. (2016). Cath: an expanded resource to predict protein function through structure and sequence. *Nucleic acids research*, **45**(D1), D289–D295.

[11] Di Tommaso, P., Moretti, S., Xenarios, I., Orobitg, M., Montanyola, A., Chang, J. M., Taly, J. F., and Notredame, C. (2011). T-Coffee: a web server for the multiple sequence alignment of protein and RNA sequences using structural information and homology extension. *Nucleic Acids Res.*, **39**(Web Server issue), W13–17.

[12] Eddy, S. R. (2011). Accelerated profile hmm searches. *PLoS computational biology*, **7**(10), e1002195.

[13] El-Gebali, S., Mistry, J., Bateman, A., Eddy, S. R., Luciani, A., Potter, S. C., Qureshi, M., Richardson, L. J., Salazar, G. A., Smart, A., *et al.* (2018). The pfam protein families database in 2019. *Nucleic acids research*, **47**(D1), D427–D432.

[14] Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., and Bengio, Y. (2014). Generative adversarial nets. *Advances in neural information processing systems*, **27**, 2672–2680.

[15] Greener, J. G., Moffat, L., and Jones, D. T. (2018). Design of metalloproteins and novel protein folds using variational autoencoders. *Scientific reports*, **8**(1), 1–12.

[16] Guo, J., Lu, S., Cai, H., Zhang, W., Yu, Y., and Wang, J. (2017). Long Text Generation via Adversarial Training with Leaked Information. *arXiv e-prints*, page arXiv:1709.08624.

[17] Gómez-Bombarelli, R., Wei, J. N., Duvenaud, D., Hernández-Lobato, J. M., Sánchez-Lengeling, B., Sheberla, D., Aguilera-Iparraguirre, J., Hirzel, T. D., Adams, R. P., and Aspuru-Guzik, A. (2018). Automatic chemical design using a data-driven continuous representation of molecules. *ACS Central Science*, **4**(2), 268–276.

[18] Hopf, T. A., Ingraham, J. B., Poelwijk, F. J., Schärfe, C. P., Springer, M., Sander, C., and Marks, D. S. (2017). Mutation effects predicted from sequence co-variation. *Nature biotechnology*, **35**(2), 128–135.

[19] Huntley, R. P., Sawford, T., Mutowo-Meullenet, P., Shypitsyna, A., Bonilla, C., Martin, M. J., and O'donovan, C. (2014). The goa database: gene ontology annotation updates for 2015. *Nucleic acids research*, **43**(D1), D1057–D1063.

[20] Kingma, D. P. and Welling, M. (2014). Auto-encoding variational bayes. *Oral presentation at the International Conference on Learning Representations, Banff, Alberta, Canada*.

[21] Lopez, R., Regier, J., Cole, M. B., Jordan, M. I., and Yosef, N. (2018). Deep generative modeling for single-cell transcriptomics. *Nature methods*, **15**(12), 1053.

[22] Makhzani, A., Shlens, J., Jaitly, N., and Goodfellow, I. J. (2015). Adversarial autoencoders. *CoRR*, **abs/1511.05644**.

[23] McInnes, L. and Healy, J. (2017). Accelerated hierarchical density based clustering. In *Data Mining Workshops (ICDMW), 2017 IEEE International Conference on*, pages 33–42. IEEE.

[24] Orengo, C. A., Michie, A., Jones, S., Jones, D. T., Swindells, M., and Thornton, J. M. (1997). Cath–a hierarchic classification of protein domain structures. *Structure*, **5**(8), 1093–1109.

[25] Pandurangan, A. P., Stahlhacke, J., Oates, M. E., Smithers, B., and Gough, J. (2018). The superfamily 2.0 database: a significant proteome update and a new webserver. *Nucleic acids research*, **47**(D1), D490–D494.

[26] Pierson, E. and Yau, C. (2015). ZIFA: Dimensionality reduction for zero-inflated single-cell gene expression analysis. *Genome Biol.*, **16**, 241.

[27] Poelwijk, F. J., Socolich, M., and Ranganathan, R. (2019). Learning the pattern of epistasis linking genotype and phenotype in a protein. *Nature communications*, **10**(1), 1–11.

[28] Popova, M., Isayev, O., and Tropsha, A. (2018). Deep reinforcement learning for de novo drug design. *Science Advances*, **4**(7).

[29] Radford, A., Metz, L., and Chintala, S. (2015). Unsupervised representation learning with deep convolutional generative adversarial networks. *arXiv preprint arXiv:1511.06434*.

[30] Rampasek, L., Hidru, D., Smirnov, P., Haibe-Kains, B., and Goldenberg, A. (2017). Dr.VAE: Drug Response Variational Autoencoder. *arXiv e-prints*, page arXiv:1706.08203.

[31] Riesselman, A. J., Ingraham, J. B., and Marks, D. S. (2018). Deep generative models of genetic variation capture the effects of mutations. *Nat. Methods*, **15**, 816–822.

[32] Rives, A., Goyal, S., Meier, J., Guo, D., Ott, M., Zitnick, C. L., Ma, J., and Fergus, R. (2019). Biological structure and function emerge from scaling unsupervised learning to 250 million protein sequences. *bioRxiv*.

[33] Russ, W. P., Figliuzzi, M., Stocker, C., Barrat-Charlaix, P., Socolich, M., Kast, P., Hilvert, D., Monasson, R., Cocco, S., Weigt, M., *et al.* (2020). An evolution-based model for designing chorismate mutase enzymes. *Science*, **369**(6502), 440–445.

[34] Salinas, V. H. and Ranganathan, R. (2018). Coevolution-based inference of amino acid interactions underlying protein function. *elife*, **7**, e34300.

[35] Sanchez-Lengeling, B. and Aspuru-Guzik, A. (2018). Inverse molecular design using machine learning: Generative models for matter engineering. *Science*, **361**(6400), 360–365.

[36] Sanchez-Lengeling, B., Outeiral, C., Guimaraes, G. L., and Aspuru-Guzik, A. (2017). Optimizing distributions over molecular space. an objective-reinforced generative adversarial network for inverse-design chemistry (organic).

[37] Shen, M. Y. and Sali, A. (2006). Statistical potential for assessment and prediction of protein structures. *Protein Sci.*, **15**(11), 2507–2524.

[38] Sikosek, T., Chan, H. S., and Bornberg-Bauer, E. (2012). Escape from adaptive conflict follows from weak functional trade-offs and mutational robustness. *Proceedings of the National Academy of Sciences*, **109**(37), 14888–14893.

[39] Sillitoe, I., Dawson, N., Lewis, T. E., Das, S., Lees, J. G., Ashford, P., Tolulope, A., Scholes, H. M., Senatorov, I., Bujan, A., *et al.* (2018). Cath: expanding the horizons of structure-based functional annotations for genome sequences. *Nucleic acids research*, **47**(D1), D280–D284.

[40] Sinai, S., Kelsic, E., Church, G. M., and Nowak, M. A. (2017). Variational auto-encoding of protein sequences. *arXiv preprint arXiv:1712.03346*.

[41] Suzek, B. E., Wang, Y., Huang, H., McGarvey, P. B., Wu, C. H., and Consortium, U. (2014). Uniref clusters: a comprehensive and scalable alternative for improving sequence similarity searches. *Bioinformatics*, **31**(6), 926–932.

[42] Tubiana, J., Cocco, S., and Monasson, R. (2019). Learning protein constitutive motifs from sequence data. *Elife*, **8**, e39397.

[43] Way, G. P. and Greene, C. S. (2017). Extracting a biologically relevant latent space from cancer transcriptomes with variational autoencoders. *bioRxiv*.

[44] Webb, B. and Sali, A. (2014). Comparative protein structure modeling using modeller. *Current protocols in bioinformatics*, **47**(1), 5–6.

[45] White, T. (2016). Sampling generative networks: Notes on a few effective techniques. *CoRR*, **abs/1609.04468**.

[46] Yu, L., Zhang, W., Wang, J., and Yu, Y. (2016). SeqGAN: Sequence Generative Adversarial Nets with Policy Gradient. *arXiv e-prints*, page arXiv:1609.05473.

[47] Zhang, Y., Gan, Z., Fan, K., Chen, Z., Henao, R., Shen, D., and Carin, L. (2017). Adversarial Feature Matching for Text Generation. *arXiv e-prints*, page arXiv:1706.03850.