Enhancing Peptide Mass Spectra Encoder through Pretraining using Contrastive Predictive Coding

Abel Legese Shibiru (abellegese@aims.ac.za)
African Institute for Mathematical Sciences (AIMS)

Supervised by: Kevin Eloff, Amandla Mabona, Jeroen Van Goey
InstaDeep, South Africa

16 June 2024

Submitted in partial fulfillment of a AI for science masters degree at AIMS South Africa

Abstract

Proteins are an important elements of life which plays an important role in several applications, including therapies and materials. Proteins are represented by a sequence, a specific order of amino acids linked together. Identification of protein sequences is necessary to perform various tasks that include designing new drugs, predicting protein function, and determining protein structure. Low cost sequencing technologies enables the identification of massive volume of new protein sequences. However, analyzing these vast datasets presents a challenge and require sophisticated tools. Database search-based algorithms have emerged as the predominant methods for sequence identification from mass spectra. However, one notable disadvantage of these approaches is the small number of peptide sequences available in the database compared to the huge array of peptides observed in nature. Deep learning techniques enabled advanced peptide mass spectrum sequencing. Despite advances in deep learning, *de novo* peptide sequencing remains challenging for peptide identification from spectra. Current methods struggle with both Post Translational Modification (PTMs) due to limited training data and noisy/incomplete spectra. One possible approach to overcome the challenge of the current methods is by using unsupervised learning to pretrain the networks. There are a massive volume of unlabelled mass spectra data that can be harnessed to pre-train the *de novo* architectures. This study aims to improve *de novo* peptide sequencing by pretraining transformer encoder using Contrastive Predictive Coding (CPC) based unsupervised learning. The result highlights CPC's effectiveness for spectra *library* search task. On the 9-species-V2 dataset, CPC outperform OpenMS by 3.73% and 4.15% in average amino acid level precision and recall, respectively. It also shows a 3.8% improvement in peptide-level recall, highlighting the effectiveness of CPC pre-training method. Comparison on inference speed revealed that CPC significantly outperforms OpenMS on various dataset size, indicating that it is scalable at large dataset size. This research has established a solid groundwork for the effective pretraining of mass spectra encoders using CPC.

Declaration

I, the undersigned, hereby declare that the work contained in this research project is my original work, and that any work done by others or by myself previously has been acknowledged and referenced accordingly.

Scan your signature

Abel Legese Shibiru, 16 June 2024

Contents

1. Introduction

Proteins play important roles in several applications, including medicines and materials (Zhang et al., 2022). Proteins are made up of a linear chain of amino acids that fold into a precise 3D shape. Understanding protein sequences is essential for many downstream modeling applications in biology, such as drug design, functional annotation, and protein structure prediction. Low-cost sequencing technologies (Ma, 2015; Ma and Johnson, 2012) enabled the discovery of a large number of novel protein sequences. Peptide sequencing is the process of determining the amino acid sequence of a peptide molecule (Aebersold and Mann, 2003). It is applied in proteomics for identifying proteins (Chi et al., 2010), in pharmaceutical research for developing peptide-based drugs (Lin et al., 2020), and in biochemistry for studying protein structure and function (Mann and Jensen, 2003). There are several methods of peptide sequencing. Database search algorithms have become the dominant approaches for identifying sequences (Ma, 2010; Noor et al., 2021). Database search involves comparing experimental data (such as peptide sequences from mass spectrometry) against a database of known sequences to identify matches or similarities (Qiao et al., 2021). However, their limitation lies in their dependency on pre-existing databases, which may not encompass all possible peptide sequences, thereby restricting the identification of novel or rare peptides that are not represented in the database. As a result, the identification of peptides using such methods could result in low performance metric] (Jin et al., 2024).

With the emergence of deep learning techniques (LeCun et al., 2015), new approaches have surfaced to advance peptide sequencing. Deep-learning have demonstrated remarkable generalization capabilities (Novak et al., 2018), learning consistent underlying patterns across the dataset. Modern deep learning techniques enhance traditional approaches by providing more accurate and generalizable predictions, thereby enabling advancements in *de novo* peptide sequencing rather than solely relying on database retrieval. As a pioneering work in applying the modern deep-learning techniques, Tran et al. (2017) introduced a peptide sequencing method that use long short-term memory (LSTM) architecture (Hochreiter and Schmidhuber, 1997) for peptide sequence prediction and convolutional neural network (CNN) (LeCun et al., 1995) to learn spectrum features. Their result showcase the capability of the deep-lerning methods over traditional methods. Recently, attention-based architectures, specifically transformers Vaswani et al. (2017) have gained attention for their efficient handling of long-range dependencies in data, resulting in superior performance in natural language processing and sequence modeling tasks. By leveragin this powerful architectures, Yilmaz et al. (2022) proposed a transformer framework, Casanovo, that exploits the self-attention mechanism that capable of learning the sequence of mass spectral peaks and the sequence of amino acids. Casanovo achieved state-of-the-art performance on diverse benchmark datasets with their method. Inspired by these accomplishments, Eloff et al. (2023) introduced InstaNovo, a transformer neural network, to improve bottom-up mass spectrometry-based proteomics. Trained on 28 million labeled spectra matched to 742k human peptides, InstaNovo outperforms current methods, having broad biological applications.

Despite the development of various deep learning methods for identifying amino acid sequences (peptides) responsible for observed spectra, challenges persist in *de novo* peptide sequencing. Firstly, prior methods struggle to identify amino acids with PTM due to their lower frequency in training data compared to canonical amino acids, further resulting in decreased peptide-level identification precision (Xia et al., 2024). Secondly, diverse types of noise and missing peaks in mass spectra reduce the reliability of training data (peptide-spectrum matches, PSMs). Despite significant advancements in deep learning methods for peptide sequencing, *de novo* sequencing still faces several challenges. One major issue is that post-translational modifications are presented at low frequencies in the training data, making them less identifiable (Xia et al., 2024). Additionally, certain amino acids occur at low frequencies in

the dataset, complicating accurate prediction. The noisy nature of mass spectra data, characterized by missing peaks and other inconsistencies, further exacerbates these challenges, limiting the reliability and accuracy of *de novo* peptide sequencing. The limitation in the development of more performant models for bottom-up mass spectrometry-based proteomics is the underutilization of unlabelled mass spectra data for pretraining these architectures in unsupervised way to learn the underlying useful features. Unlabelled data contain valuable information that could enhance the model's understanding of spectral patterns and improve its ability to generalize across a wider range of spectra.

Unsupervised learning is an effective method for extracting general representations from unlabeled data (Oord et al., 2018). But Learning a general representation using unsupervised learning across different data modalities is challenging due to each modality's unique characteristics, which require specialized techniques to capture meaningful patterns (Oord et al., 2018). One common approach to unsupervised learning is predicting missing information or future in the embedding space, which helps the model learn contextual representations and underlying patterns of the data. Predicting the future, known as predictive coding (Elias, 1955; Atal and Schroeder, 1970), draws inspiration from neuroscience to understand how the brain anticipates and processes incoming sensory information. In neuroscience, predictive coding theories suggest that the brain predicts observations at various levels of abstraction (Rao and Ballard, 1999; Friston, 2005). Inspired by predictive coding, Oord et al. (2018) introduced CPC, that offers a promising possibilities for leveraging unlabelled data in unsupervised learning contexts. CPC operates by predicting future representations of input data within a contrastive framework, where the model learns to distinguish between true future representations and predicted representations.

In this paper we aim to enhance the performance of a *de novo* peptide sequencing architecture by presenting a CPC based pretraining approach. By assuming that CPC will learn non-coherent, noisy spectra to maximize the mutual information between actual and predicted embedding, we extended its application to the domain of pre-training *de novo* peptide sequencing. Through our experiments, we demonstrated the effectiveness of our proposed approach in learning useful representation by aplying it to spectra *library* search task and also compare the inference speed. Leveraging a large amount of unlabeled mass spectra data, the model was able to learn useful feature using unsupervised learning that pave the road for the development of more robust and generalizable models for bottom-up mass spectrometry-based proteomics.

2. Literature Review

2.1 Overview of Peptide Sequencing

The workhorses of the cell, proteins are engaged in almost all cellular functions. They function as signaling molecules, structural elements, enzymes, and more (Sim et al., 2019). It is essential to identify and characterize the proteins present in a sample in order to comprehend these processes at the molecular level. Researchers can catalog the protein composition of cells, organs, and organisms by analyzing these proteins in detail thanks to peptide sequencing (Aebersold and Mann, 2003; Chi et al., 2010). Peptide sequencing is the process of directly reconstructing a peptide sequence from a tandem mass spectrum and peptide mass and charge values (Qiao et al., 2016). Several *de novo* peptide sequencing techniques have been put forth over the previous 20 years, with successful results demonstrated in the assembly of monoclonal antibody sequences (Tran et al., 2016) and the identification of antigens specific to tumors, particularly those arising from non-coding regions or alternative splicing (Faridi et al., 2018; Laumont et al., 2018). It is a key procedure in proteomics, which is necessary to determine peptides' amino acid sequences from their mass spectra (Sobolesky et al., 2016). It also forms the basis for a wide range of biological and medical research applications. One of the main application of protein sequencing is to analyze proteins in complex biological samples (Sim et al., 2019). Researchers can learn valuable information about the structure and function of proteins, which is essential to comprehending biological processes and disease mechanisms, by deciphering the exact sequence of amino acids in a peptide (Benitez-Amaro et al., 2019).

The identification of possible pharmacological targets is a further noteworthy application of peptide sequencing. Dysregulated proteins or pathways are linked to a number of diseases, such as diabetes, cancer, and neurodegenerative disorders (Chi et al., 2010). These important proteins and their alterations can be found by sequencing peptides, which gives researchers useful targets for medication development. In this technique, distinct peptide sequences linked to illness states are identified. Therapeutic medicines can then target these sequences (Lin et al., 2020). Moreover, the identification of PTMs on proteins depends on peptide sequencing. PTMs are essential for controlling the function and activity of proteins. Examples of these include ubiquitination, glycosylation, and phosphorylation (Alley et al., 2019). These changes can affect a protein's stability, behavior, and interactions with other molecules. Deciphering the intricate regulatory networks found within cells requires an understanding of PTMs which is possible using peptide sequencing. Peptide sequencing based on mass spectrometry is a potent tool for detecting and describing these alterations, providing information on how they affect protein function and are linked to both health and disease (Mann and Jensen, 2003). The ability of the *de novo* sequencing tool to reconstruct the precise sequence of a peptide is crucial for applications such as antibody sequencing or the identification of tumor-specific antigens (Tran et al., 2017). Otherwise, a vaccine or medication could be ineffective due to an amino acid discrepancy (Dančík et al., 1999).

The mass accuracy has been increased to about 1 part per million (ppm) thanks to recent developments in mass spectrometer technology (Xia et al., 2024). This indicates that the measurement error for a fragment ion of mass 1,000 Da is less than 0.001 Da (Chen et al., 2001). Accurate *de novo* peptide sequencing is made possible by such high-resolution data. However, the majority of *de novo* sequencing instruments in use today were created when mass error was more than 100 ppm (Frank and Pevzner, 2005). Using the increased precision offered by the newest mass spectrometer generation fully by such instruments is not simple (Ma et al., 2003). A better precision for spectrum graph methods, that represents mass spectrometry data with vertices as m/z values and edges as possible amino acid residues, (Dančík et al., 1999; Chen et al., 2001; Frank and Pevzner, 2005) results in fewer nodes being merged

and a larger spectrum graph being produced, which inevitably increases computational complexity (Ma et al., 2005). Moreover, prior to discretizing a spectrum to an intensity vector, neural network-based *de novo* sequencing models like DeepNovo and SMSNet must be used. For instance, when the spectrum resolution parameter is set to 50, DeepNovo represents a spectrum with a vector with a length of 150,000 (Karunratanakul et al., 2019). The creation and processing of long intensity vectors takes a significant amount of memory and CPU time. In fact, the original DeepNovo solution frequently underutilized the GPU because the program had to wait for the CPU to create and process such vectors (Qi et al., 2017). To benefit from the increased precision provided by higher-resolution spectra, DeepNovo and SMSNet must discretize spectra with a higher spectral resolution parameter (R) (Hochreiter and Schmidhuber, 1997).

Peptide sequencing is extremely important, however still conventional techniques have a lot of drawbacks (Alley et al., 2019). De novo peptide sequencing technology is still unable to distinguish between pairs of amino acids that have similar mass, such as leucine (L) and isoleucine (I), glutamine (Q) and lysine (K), or methionine sulfoxide (M(Oxi)) and phenylalanine (F) (Faridi et al., 2018). For example, some earlier research (Ma, 2015; Tran et al., 2017) evaluated if a projected amino acid matches an actual amino acid if the difference in mass between them is less than 0.1 Da and if the mass of the prefix preceding them differs by less than 0.5 Dalton (Da) (Laumont et al., 2018). This was done in order to assess the accuracy of *de novo* peptide sequencing (Ma, 2015). For instance, if a *de novo* sequencing tool reports a Q for a ground-truth K, the evaluation criteria will still classify it as correct because the mass difference between Q and K is less than 0.05 Da. Researchers have been developed several modern deep learning techniques to overcome those challeges.

Peptide sequencing has seen revolutionary developments with the advancement of deep learning techniques (Cao et al., 2020). Significant advancements over conventional approaches have been achieved by the extraordinary ability of deep learning models, especially neural networks, to discover intricate patterns from data (LeCun et al., 2015; Novak et al., 2018). A groundbreaking deeplearning method achieved remarkable results in peptide sequencing by combining long short-term memory (LSTM) networks for sequence prediction with convolutional neural networks (CNNs) for feature extraction (Tran et al., 2017). Transformers in particular, an attention-based architectures, have been more well-known recently because of their outstanding performance in a variety of tasks (Huang et al., 2019). Transformers are well equipped to handle the sequential nature of peptide data because they model dependencies within sequences using self-attention processes. Motivated by their triumph in natural language processing, researchers have implemented transformers in peptide sequencing, yielding an improved results over other deep-learning methods (Bouatta et al., 2021).

Generally there are two main approaches to peptide sequencing: traditional database search and deep learning-based de novo methods. Traditional methods rely on comparing experimental spectra to a database of known protein sequences to identify matching peptides. Deep learning, on the other hand, offers a de novo approach, capable of sequencing peptides without relying on a pre-existing database.

2.2 Spectra *Library* Search

Mass spectra *library* search is a crucial technique in proteomics and metabolomics for identifying compounds based on their mass spectrometry (MS) data. This process involves comparing the acquired mass spectra of unknown compounds against a library of known spectra to find matches. By leveraging extensive spectral libraries, researchers can accurately identify and characterize molecules, leading to insights into biological processes and the discovery of biomarkers. The effectiveness of a database search relies on the quality and comprehensiveness of the reference database, as well as the algorithms

used for matching and scoring spectra. The synergy between large-scale DNA sequencing and mass spectrometry was investigated by Wolters et al. (2001) for protein identification using spectra *library* search. They described how mass spectrometry can determine peptide weights and sequences with high sensitivity, and how computer algorithms utilize this data to search sequence databases. The study highlights two main methods: peptide mass mapping for organisms with completed genomes and using tandem mass spectra to search protein and nucleotide databases, which is beneficial for organisms with extensive but incomplete genomic data. They emphasized the strength of tandem mass spectra in identifying proteins in complex mixtures. Recently, Wang et al. (2020) introduced the Mass Spectrometry Search Tool (MASST), a web-enabled search engine designed to search all small-molecule tandem mass spectrometry (MS/MS) data in public metabolomics repositories. MASST aims to increase the reuse of public MS data by enabling similarity searches against reference libraries and public repositories. The tool supports the comparison of a single MS/MS spectrum to identify identical or analogous spectra in public datasets, overcoming limitations of existing tools that do not allow such comprehensive searches.

The sensitivity of mass spectra database searches is a critical factor, determining the ability to detect low-abundance proteins. An improved sensitivity in mass spectra database searches, was developed by Xu and Freitas (2009), MassMatrix, a program designed to match tandem mass spectra with theoretical peptide sequences from protein databases. It employs a mass accuracy sensitive probabilistic score model to rank peptide matches. Their evaluation using high mass accuracy datasets shows that MassMatrix offers better sensitivity compared to MASCOT (Brosch et al., 2009), SEQUEST (MacCoss et al., 2002), X!Tandem (Duncan et al., 2005), and OMSSA (Geer et al., 2004) at a given specificity, with a 2% false positive rate. The presence of decoy sequences and additional variable PTMs did not significantly impact the results. MassMatrix also demonstrated competitive search times and achieved search speeds of approximately 100,000 spectra per hour on distributed memory clusters when searching against a complete human database with eight variable modifications. However, the speed and performance of database search algorithms in proteomics remained a constraint, however recently, Teo et al. (2020) introduced a high-speed parallelized deisotoping algorithm integrated into the MSFragger search engine, which was previously known for its ultrafast proteomics searches without deisotoping. This new algorithm, drawing on elements from existing methods, significantly improves the speed and performance of database searches, especially for complex methods like open or nonspecific searches. The evaluation of the deisotoping method across various instrument types and vendors shows a substantial enhancement in performance, providing an updated perspective on the role of deisotoping in modern proteomics.

Enhancing data acquisition efficiency and reduce scan times in proteome analyses is crucial for accelerating the identification and quantification of proteins, thereby enabling more comprehensive and timely insights into complex biological systems. Recently, Schweppe et al. (2020) described the development of Orbiter, a real-time database search (RTS) platform designed to address the long acquisition duty cycles of the "Sequential Window Acquisition of All Theoretical Mass Spectra with MS3" (SPS-MS3) method used in multiplexed quantitative proteome analyses. Orbiter reduces SPS-MS3 scan times by eliminating scans if no peptide matches are found in a given spectrum. Using an online proteomic analytical pipeline that includes RTS and false discovery rate analysis, Orbiter processes single spectrum database searches in under 10 milliseconds. This approach enhances the identification and quantification of peptide spectral matches using Comet (tool used for peptide identification in proteomics) (Eng et al., 2013), facilitating analysis of PTMs with validated scoring. Testing demonstrated that RTS with Orbiter increased data acquisition efficiency two-fold, quantifying over 8000 proteins across 10 proteomes in 18 hours compared to 36 hours for SPS-MS3.

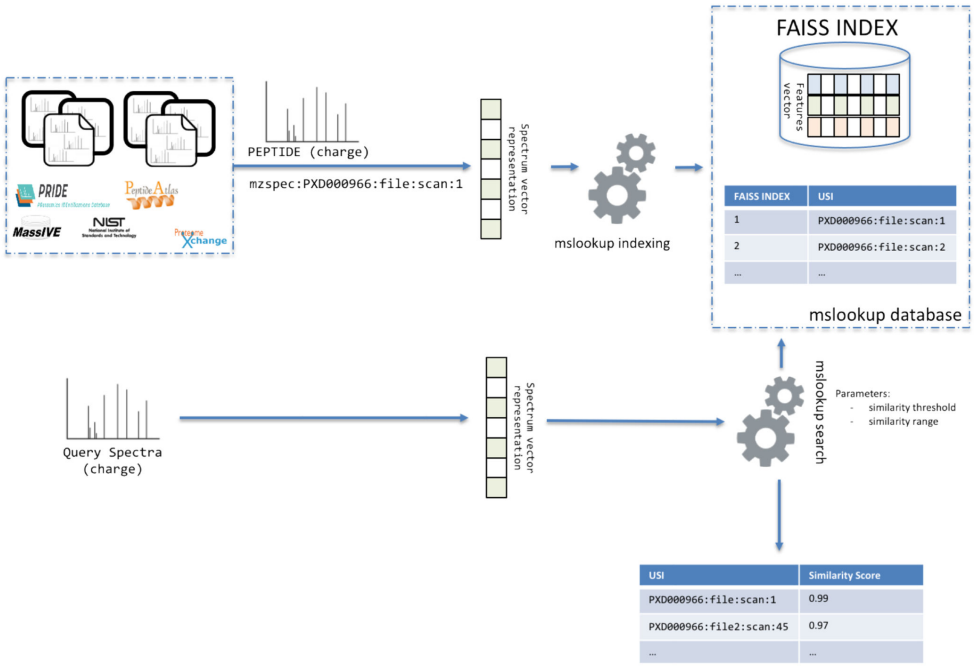


Figure 2.1: The peaks, charge, and precursor are encoded into a 32-dimensional vector using mslookups. Furthermore, the tool generates a key-value database containing the 32-D representation and the Universal Spectrum Identifier (USI). When a user uses the mslookup search tool to query a spectra file against the database, the query spectrum is encoded and compared to the vector database (faiss), yielding a list of USIs and their related similarity score (Qin et al., 2021).

Deep learning is enabling a traditional database search methods with a suite of innovative tools. These include DeepNovo (Tran et al. (2017)) for de novo peptide sequencing, pDeep (Zhou et al. (2017); Zeng et al. (2019)) and Prosit (Gessulat et al. (2019)) for peptide-protein identification using predicted spectra, and capabilities for simulating spectra for spectral library generation and spectral clustering. Deep leaning tools has been applied to learn embedding directly from a raw spectra. Deep learning-based tools excel at grouping spectra into distinct "communities" (Gessulat et al., 2019). This allows researchers to identify clusters of unknown spectra likely representing the same peptide. It also aids in uncovering misidentified spectra and assigning peptide sequences to those that have consistently eluded identification with traditional methods. This model is expected to perform well in prediction due to its deep learning network's capacity to learn implicit and effective features from large-scale training datasets (Zeng et al., 2019). However, the direct performance comparison between the deep learning model and the classical method in spectral similarity scoring is unknown, however interesting. Qin et al. (2021) proposed using deep learning to improve spectral similarity comparison in proteomics, particularly for large-scale datasets. They evaluated deep learning-based models against traditional methods like normalized dot product (NDP) and Pearson's as shown in **Figure 2.1**. Although some traditional methods still excel, their method outperforms in specific datasets, such as Yeast-UPS (Belouah et al., 2019). Deep learning methods, while less accurate in some cases, are better suited for big data analyses due to their efficient similarity calculations using low-dimensional embedding.

Recent advances have demonstrated that neural networks can successfully train embedding directly from mass spectrometry data, which are then used in database search techniques. Huber et al. (2021)

investigated the use of mass spectrometry data, essential in medicine and life sciences, to predict structural similarity between chemical compounds using MS/MS fragmentation spectra. They introduced MS2DeepScore, a Siamese neural network, trained on over 100,000 spectra from around 15,000 compounds. MS2DeepScore accurately predicts structural similarity with a root mean squared error (RMSE) of 0.15 and improves prediction reliability using Monte-Carlo Dropout. It outperforms classical spectral similarity measures in identifying chemically related compounds and can cluster large spectral datasets, demonstrating its potential for metabolic data processing. Considering the limitations of isolated peptide mass spectra analysis that makes neural network to learn them effectively, Bittremieux et al. (2022) introduced GLEAMS, a transformative deep neural network. GLEAMS learns embedding of spectra into a low-dimensional space, ensuring spectra from the same peptide are proximate. This breakthrough allows for the effective propagation of annotations from labeled to unlabeled spectra, enhancing data utility. GLEAMS also excels in identifying groups of unidentified spectra, revealing misidentified spectra, and characterizing frequently observed unidentified molecular species. The approach significantly facilitates large-scale analyses by quickly embedding additional spectra using a pre-trained model.

The completeness of the database affects database search-based methods, which compare observed mass spectra with theoretical spectra from known peptide sequences (Bepler and Berger, 2019). Only a small portion of all potential sequences can be included in current databases due to the enormous diversity of peptides found in nature. This constraint severely limits these techniques' capacity to find novel peptides that are absent from the database (Jin et al., 2024). Methods for *de novo* sequencing have been developed to address these issues. *de novo* sequencing reconstructs peptide sequences directly from mass spectra, independent of pre-existing databases, in contrast to database-dependent techniques. Heuristic algorithms were employed in early *de novo* sequencing techniques, such PEAKS (Ma et al., 2003), to analyze spectra. Nevertheless, complicated spectra and noisy data frequently presented challenges for these approaches, requiring more development.

2.3 Methods of *de novo* Peptide Sequencing

Proteomics research focuses on large-scale studies to chacterize the proteome, the entire set of proteins, in a living organism (Xia et al., 2024). Tandem mass spectrometry (MS), as the main-stream high-throughput technique to identify protein sequences, plays an essential role in proteomics research. As shown in **Figure 2.2**, in a standard identification workflow of shotgun proteomics (Wolters et al., 2001), proteins undergo initial digestion by enzymes, yielding a mixture of peptides. A tandem mass spectrometer measures mass-to-charge (m/z) ratios of each peptides in a two-scan process (Xia et al., 2024). During the first scan (MS1), the mass-to-charge (m/z) ratios of intact peptides, also known as precursors, are measured. Following this, peptides undergo fragmentation, and the resulting fragments are analyzed in a subsequent scan. In the second scan (MS2), peptides are fragmented at random locations along the peptide backbone, generating peaks corresponding to prefixes (b-ions) and suffixes (y-ions) of the peptide, each associated with a specific charge state. Consequently, the MS2 spectrum comprises a collection of peaks. Each peak is characterized by an m/z value and an associated intensity (Bassani-Sternberg et al., 2016). The intensity, though unitless, is directly proportional to the number of ions contributing to the observed peak. The m/z value is measured with remarkable precision, while the intensity is measured with comparatively lower precision (Tran et al., 2019b). The core of the above pipeline is the spectrum identification problem, where we aim to predict the peptide sequence responsible for generating the observed MS2 spectrum and the corresponding precursor information (mass and charge of the peptide) (Xia et al., 2024).

Early *de novo* sequencing methods used dynamic programming to score peptide sequences against

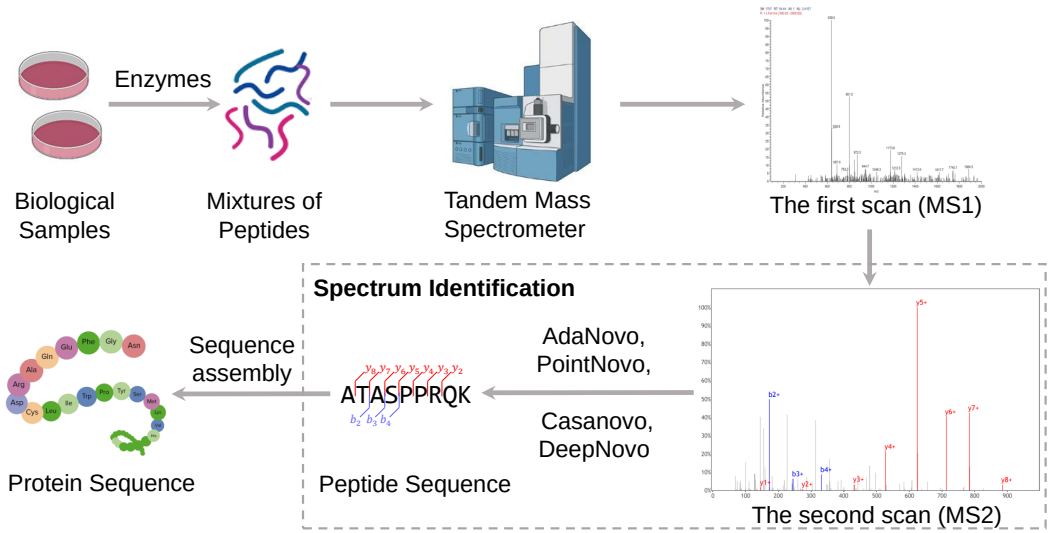


Figure 2.2: The shotgun proteomics identification workflow (Wolters et al., 2001). This work study's aim for spectrum identification is to generate the peptide sequence (such as ATASPPRQK) for the observed spectrum. Grey peaks in the spectrum signify noise or unforeseen fragmentation events, whereas peaks reflecting the b- and y-ions of the linked peptide are highlighted in color. ProteomeXchange is used to construct the spectrum annotations (Vizcaıno et al., 2016).

each observed spectrum. Ma et al. (2003) introduced PEAKS, a sophisticated dynamic programming algorithm to compute the best sequences whose fragment ions can best interpret the peaks in the MS2 spectrum. Graph-based algorithms, such as Sherenga (Dančík et al., 1999) and pNovo (Taylor and Johnson, 2001), first translated the spectrum into a "spectrum graph" where nodes in the graph correspond to peaks in the spectrum and two nodes are connected by an edge if the mass difference between the two corresponding peaks is equal to the mass of an amino acid. The *de novo* peptide sequencing problem is thus cast as finding the path in the resulting graph. Machine learning methods for *de novo* sequencing Fischer et al. (2005); Frank and Pevzner (2005) have been introduced and significantly improved the accuracy. The PepNovo (Frank and Pevzner, 2005) algorithm present a novel scoring method, which uses a probabilistic network whose structure reflects the chemical and physical rules that govern the peptide fragmentation. The Novor algorithm presented by Ma (2015) achieved improved performance by using large decision trees as score function in a dynamic programming algorithm. But those method was not be able to perform well on the high precision mass spectra.

In order to take full advantage of the high precision that the most recent mass spectrometers have to offer, Qiao et al. (2016) introduced PointNovo, a *de novo* peptide sequencing tool based on neural networks that does not vectorize the mass spectrum (Yang et al., 2019). Without any additional complexity, PointNovo is prepared to be applied to higher-resolution data that may be created in the future. Furthermore, the outcomes of thier experiment demonstrate that PointNovo performs significantly better than earlier state-of-the-art techniques (Zhu et al., 2018). PointNovo does this through using an order-invariant network structure (Qi et al., 2017) to learn from the data of such a structure, and by explicitly encoding a spectrum as a set of m/z values and intensity pairings (Shears et al., 2019). But despite the effectiveness of the early neural networks, another revolution was necessary to make this field even advance further.

The first deep neural network method for *de novo* peptide sequencing, DeepNovo (Tran et al., 2017),

treats the de novo sequencing task as an image caption task and com- bines CNN with LSTM to predict the sequence. SMSNet (Karunratanakul et al., 2019) is a hybrid approach which leverages a multi-step Sequence-Mask-Search strategy and adopts the encoder-decoder architecture, basically formulating peptide sequencing as a spectra-to-peptide language translation problem. PointNovo (Qiao et al., 2021) adopts an order invariant network structure for peptide sequencing, which focuses specifically on high-resolution mass spectrometry data. Similar to SMSNet (Karunratanakul et al., 2019), Casanovo (Yilmaz et al., 2022) frames the problem as a language translation problem and employs a transformer framework that has been widely used to process and predict sequences. Although *de novo* methods have achieved notable progress, we observe that they have difficulty in identifying the amino acids with PTMs because these amino acids occur much less frequently in datasets compared to other common amino acids, making it challenging for the model to learn. Additionally, mass spectrometry data contains a significant amount of noise typically originated from the electronic fluctuations in the instruments and other molecules in the biological samples. In other words, there are plenty of noise peaks mixing together with the real ions. All of these make the peptides labels being less reliable. These issues limit the predictive accuracy and widespread use of *de novo* methods.

2.4 Deep Learning Architectures

Peptide sequencing has advanced significantly with the introduction of deep learning. Neural networks, in particular, have shown to be capable of understanding intricate patterns from data, providing improved generalization capabilities over conventional heuristic techniques (LeCun et al., 2015; Novak et al., 2018). One of the pioneering works in this domain is DeepNovo, a deep learning system that combines convolutional neural networks (CNNs) for feature extraction with long short-term memory (LSTM) networks for sequence prediction (Tran et al., 2017). But the modern advanced sequencing method uses a powerful autoregressive model, transformer that further enhance the performance of the existing deep-learning based peptide sequencing tools.

2.5 Transformers

Natural language processing has seen a revolution after Vaswani et al. (2017) introduced Transformer models. A transformer model is a neural network architecture built for natural language processing applications. It uses self-attention mechanisms inside an encoder-decoder framework to process input data in parallel and capture long-range dependencies regardless of sequential order. The encoder block primarily consists of a multi-head self-attention mechanism and a position-wise feed-forward network (FFN). Transformer models require deeper layers to efficiently learn and handle more complex tasks. A residual connection (Mao et al., 2016) is used in each component of the model. Other important component is a Layer Normalization (Ba et al., 2016), that follows the residual connection. It helps to speed up and stabilize the training process in deep networks. In the decoder block the only difference from the decoder is that, we have additional attention mechanism called causal attention that prevents each position from attending to subsequent positions (**Figure 2.3**). Cross-attention is another component of the decoder model to process the information from the encoder.

2.5.1 Attention Mechanism

Transformers utilize an attention mechanism implemented through the Query, Key, and Value (QKV) model. The queries $Q \in \mathbb{R}^{m \times d_k}$, keys $K \in \mathbb{R}^{n \times d_k}$, and values $V \in \mathbb{R}^{n \times d_v}$ are a matrix which the scaled dot-product attention defined as below.

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^\top}{\sqrt{d_k}}\right)V = AV, \tag{2.5.1}$$

where m and n denote the lengths of queries and keys (or values); d_k and d_v denote the dimensions of keys (or queries) and values. We use the square root of the dimension of the model in attention mechanisms to keep the variance of the dot product ($QK^\top$) constant across different dimensions, thereby preventing issues such as vanishing gradients during training.

Multi-head attention divides the attention mechanism into multiple heads by independently projecting the queries, keys, and values Q_i, K_i, V_i (where i ranges from 1 to h) into d_k/h-dimensional subspaces, allowing the model to attend to different parts of the input sequence simultaneously and integrate diverse perspectives effectively. For each head, the attention mechanism computes separate sets of queries, keys, and values, enabling the model to perform distinct attention calculations in parallel across the projected subspaces.

For each of the projected queries, keys and values, an output is computed with attention according to Equation. (2.5.1). The model then concatenates all the outputs and projects them back to a D_m-dimensional representation.

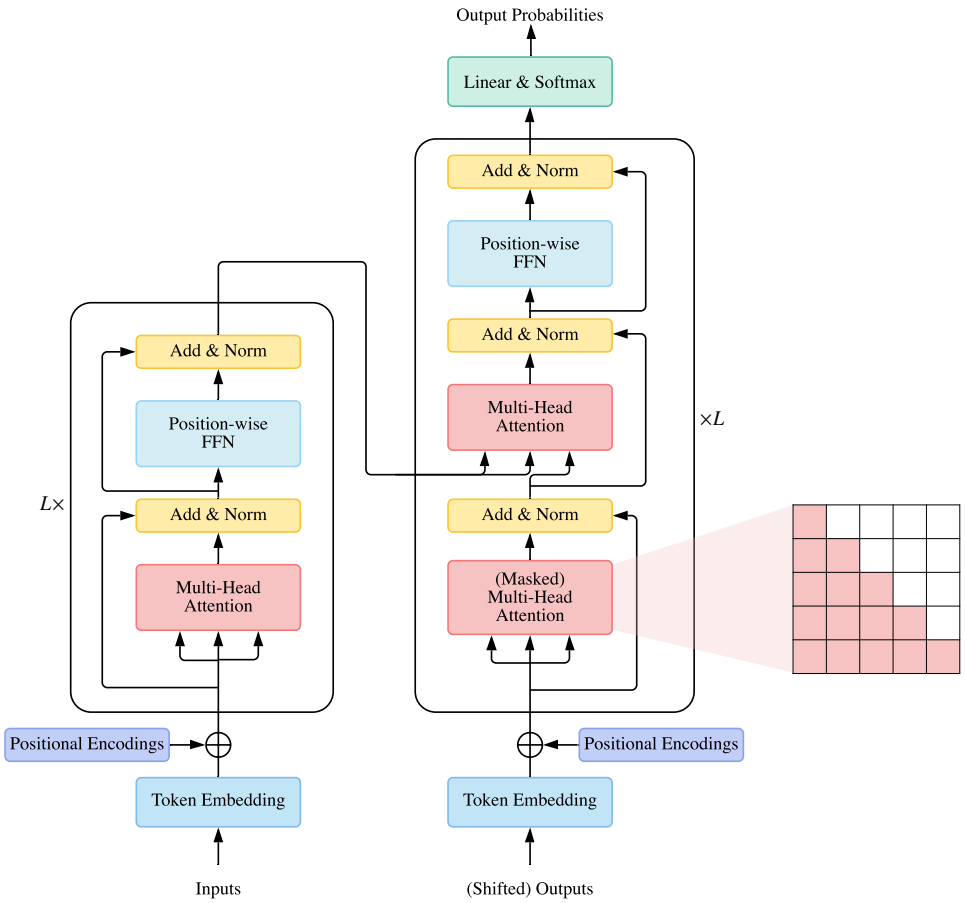


Figure 2.3: An overview of the architecture of the vanilla Transformer model (Lin et al., 2022; Vaswani et al., 2017).

$$\text{MultiHeadAttn}(Q, K, V) = \text{Concat}(\text{head}_1, \cdots, \text{head}_H)W_O, \tag{2.5.2}$$

where

$$\text{head}_i = \text{Attention}(QW_i^Q, KW_i^K, VW_i^V). \tag{2.5.3}$$

In the transformer model there three types of attention exists:

- **Self-attention**: This mechanism computes attention scores between different positions of a single sequence. It allows each position in the sequence to attend to other positions.
- **Cross-attention**: Cross-attention extends self-attention to handle interactions between two different sequences or modalities. It computes attention scores between positions in one sequence and positions in another sequence, enabling information exchange and alignment across sequences. It only used when the encoder model are applied.
- **Causal attention**: Ensures that each position can only attend to positions that precede it in the sequence. It is commonly used in autoregressive models to maintain the causality constraint, where predictions are based only on previous tokens in the sequence. In causal attention, the attention scores are computed using a mask to ensure that each position i can only attend to positions j where $j < i$. The attention weights a_{ij} are computed as:

$$a_{ij} = \begin{cases} \text{softmax}(\text{score}(q_i, k_j)), & \text{if } j < i \\ 0, & \text{otherwise} \end{cases} \tag{2.5.4}$$

  where q_i and k_j represent queries and keys at positions i and j, respectively, and $\text{score}(q_i, k_j)$ is a compatibility function between queries and keys (e.g., dot product or scaled dot product).

2.5.2 Feed Forward Network (FFN)

The feed forward network in each transformer layer applies two linear transformations with a ReLU activation in between. It serves to capture complex, non-linear relationships in the data by transforming the representation learned from the attention mechanism.

$$\text{FFN}(x) = \text{ReLU}(xW_1 + b_1)W_2 + b_2$$

2.5.3 Residual Connection and Layer Normalization

A residual connection (Mao et al., 2016) is employed around each sub-layer in the transformer (Vaswani et al., 2017), adding the input to the output before applying normalization.

$$\text{Output} = \text{LayerNorm}(x + \text{Sublayer}(x))$$

This mechanism helps in easier training of deeper networks by allowing gradients to flow more smoothly, thereby mitigating the vanishing gradient problem. Layer normalization (Ba et al., 2016) normalizes the activations of each layer independently. By stabilizing the learning process, it enables the model to converge faster and achieve better generalization performance.

$$\text{LayerNorm}(x) = \frac{x - \mu}{\sigma} \odot \gamma + \beta$$

where μ and σ are the mean and standard deviation of x, γ and β are learnable parameters.

2.5.4 Positional Encoding

Positional encoding is added to the input embeddings to provide positional information, enabling the model to understand the order of the sequence without recurrence or convolution. It uses sinusoidal functions to encode relative positions in the sequence by incorporating different frequencies and phases into the input embeddings (Vaswani et al., 2017).

The positional encoding function $\text{PE}(pos, 2i)$ is defined as:

$$\text{PE}(pos, 2i) = \sin\left(\frac{pos}{10000^{2i/d_{\text{model}}}}\right),$$

and the function $\text{PE}(pos, 2i+1)$ is defined as:

$$\text{PE}(pos, 2i+1) = \cos\left(\frac{pos}{10000^{2i/d_{\text{model}}}}\right).$$

Here, pos represents the position in the sequence, i denotes the index of the embedding dimension, and d_{model} is the dimensionality of the model.

2.6 CPC based Unsupervised Learning

In this section, we will present the detailed mathematics and proof of CPC. The original paper did not provide a detailed explanation of how the InfoNCE loss serves as a lower bound to the mutual information, so we will present this derivation adapted from (Song and Ermon, 2020) here to clarify the theoretical foundations of CPC and demonstrate the relationship between InfoNCE loss and mutual information.

In the domain of representation learning, our focus lies in acquiring knowledge from a (potentially stochastic) network $f : X \to Z$ that maps input data $x \in X$ to a condensed representation $f(x) \in Z$. To simplify notation, we define $p(x)$ as the data distribution, $p(x, z)$ as the joint distribution encompassing both data and representations (denoted by z), $p(z)$ as the marginal distribution of representations, and X, Z as the random variables associated with data and representations (Song and Ermon, 2020). The InfoMax principle (Linsker, 1988; Hjelm et al., 2018; Bell and Sejnowski, 1995) for representation learning involves optimizing the variational mutual information $I(X; Z)$:

$$I(X; Z) := \mathbb{E}_{(x,z)\sim p(x,z)}\left[\log \frac{p(x, z)}{p(x)p(z)}\right] \tag{2.6.1}$$

Various estimators of mutual information, each with distinct bias-variance trade-offs, have been proposed for representation learning (Nguyen et al., 2010; Van den Oord et al., 2016; Belghazi et al., 2018; Poole et al., 2019). The objective is to distinguish a positive pair $(x, z) \sim p(x, z)$ from $(m-1)$ negative pairs $(x, z) \sim p(x)p(z)$. Successful classification by the optimal classifier indicates a strong association between x and z, reflecting high mutual information.

For a batch of n positive pairs $\{(x_i, z_i)\}_{i=1}^{n}$, the CPC objective is formulated as:

$$L(f) := \mathbb{E}\left[\frac{1}{n}\sum_{i=1}^{n}\log\frac{m \cdot f(x_i, z_i)}{f(x_i, z_i) + \sum_{j=1}^{m-1} f(x_i, z_{i,j})}\right] \tag{2.6.2}$$

where $f : X \times Z \to \mathbb{R}^+$ represents a positively-valued critic function. The expectation is evaluated across n positive pairs $(x_i, z_i) \sim p(x, z)$ and $n(m-1)$ negative pairs $(x_i, z_{i,j}) \sim p(x)p(z)$, with $f : X \times Z \to \mathbb{R}^+$ serving as a positive critic function (Song and Ermon, 2020) .

2.6.1 Mutual information in CPC

Mutual information (MI) quantifies the statistical dependency between two random variables (Kraskov et al., 2004). In the context of representation learning, MI measures the amount of information one variable carries about another. CPC (Oord et al., 2018), interprets MI estimation as a multi-class classification task. The objective is to discriminate between a positive pair $(x, z) \sim p(x, z)$ and $(m-1)$ negative pairs $(x, z) \sim p(x)p(z)$. Successful discrimination by the optimal classifier suggests a strong relationship between x and z, indicating high mutual information.

For a batch of n positive pairs $\{(x_i, z_i)\}_{i=1}^{n}$, the CPC objective is formulated as:

$$L(f) := \mathbb{E}\left[\frac{1}{n}\sum_{i=1}^{n}\log\frac{m \cdot f(x_i, z_i)}{f(x_i, z_i) + \sum_{j=1}^{m-1} f(x_i, z_{i,j})}\right], \tag{2.6.3}$$

where $f : X \times Z \to \mathbb{R}^+$ is a positive-valued critic function. The expectation is taken over n positive pairs $(x_i, z_i) \sim p(x, z)$ and $n(m-1)$ negative pairs $(x_i, z_{i,j}) \sim p(x)p(z)$.

Oord et al. (2018) constructed the CPC objective as an approximation to the lower bound of mutual information. However, their formal proof only applies under a specific approximation where $-\mathbb{E}[\log g]$ is substituted with $-\log \mathbb{E}[g]$, limiting the assertion of CPC as a definitive lower bound for MI. Poole et al. (2019) established a lower bound when $m = n$ and negative samples are connected to other positive samples within the batch. To reconcile the theoretical concept of CPC as InfoMax with practical implementations where negative samples, such as in MoCo (Cao et al., 2019), can be independently selected from positive samples within a batch, Song and Ermon (2020) provided an alternative proof for the general CPC objective as expressed in $L(g)$. They demonstrated that variational lower bounds of Kullback-Leibler (KL) divergences between general distributions, using batches of negative samples for divergence estimation, support the proposition that mutual information represents a KL divergence between two specific distributions. Thus, the argument for CPC serving as a lower bound naturally follows.

For all probability distributions P, Q over sample space X such that $P \ll Q$, the following inequality holds for all functions $r : X \to \mathbb{R}^+$ and integers $m \geq 2$:

$$D_{KL}(P \parallel Q) \geq \mathbb{E}_{x\sim P, z_{1:m-1}\sim Q^{m-1}}\left[\log\frac{m \cdot r(x)}{r(x) + \sum_{i=1}^{m-1} r(z_i)}\right]. \tag{2.6.4}$$

Proof. See Appendix A, utilizing variational representations of f-divergences Nguyen et al. (2010) and Prop. 1.

To summarize, breakthroughs in peptide sequencing have been considerably enhanced by modern deep-learning techniques, particularly attention-based architectures. We also saw that traditional spectrum

database searches have been enhanced to include deep learning approaches, which improve peptide identification accuracy and efficiency. Transformer models have significantly improved search speed and accuracy. Furthermore, we presented the detailed math behind CPC and its theoretical guarantee for learning representation.

3. Method

In the this section we presented the methods for training the mass spectra encoder using CPC and spectra database search. Formally, we denote mass spectrum peaks as $x = \{(m_i, I_i)\}_{i=1}^{M}$, where each peak (m_i, I_i) forms a tuple representing the m/z and intensity value, and M is the number of peaks that can vary across different mass spectra. Also, we denote the precursor as $z = \{(m_{\text{prec}}, c_{\text{prec}})\}$, consisting of the total mass $m_{\text{prec}} \in \mathbb{R}$ and charge state $c_{\text{prec}} \in \{1, 2, \ldots, 10\}$ of the spectrum. Also we presented the dataset that was used to pretrain the encoder.

3.1 Dataset

For this work we used a new nine species dataset originally created by Tran et al. (2017). This dataset was compiled from nine PRIDE projects (PXD004948, PXD005025, PXD004565, PXD003868, PXD004536, PXD004947, PXD004325, PXD004467, PXD004424) and consists of 2.8 million Peptide Spectra Match (PSMs) extracted from 343 RAW files. It contains 229,984 unique peptides, with 3797 (1.7%) peptides identified that occur in more than one species. We split the dataset into training, validation, and test sets with ratios of 85%, 10%, and 5%, respectively. To ensure good evaluation of the model, the validation set was created to guarantee that each unique peptide instance had at least two peptides represented. This approach enhances the reliability of model performance assessments and to generalize to unseen data. The validation split at species level staritification, get the validation split from each species and combine them.

3.2 Model Architecture

3.2.1 Encoder

Transformers require discrete inputs, however mass spectrometry data is continuous that limit the use of vanilla transformers that are designed to process discrete token sequences (such as words in a phrase). The continuous mass-to-charge (m/z) and intensity information must be converted into discrete representations that the transformer can process. These embeddings use sine and cosine functions to encode positional information while retaining data points' order. This paper applied a numerical peak embedding method that uses a sinusoidal embedding scheme first developed by Voronov et al. (2022), defined as follows:

$$SE(m/z, 2i; d) = \sin\left(2\pi \left[\frac{\lambda_{\text{min}}}{\left(\frac{\lambda_{\text{max}}}{\lambda_{\text{min}}}\right)^{2i/(d-2)}}\right]^{-1} \frac{m/z}{2}\right) \tag{3.2.1}$$

$$SE(m/z, 2i+1; d) = \cos\left(2\pi \left[\frac{\lambda_{\text{min}}}{\left(\frac{\lambda_{\text{max}}}{\lambda_{\text{min}}}\right)^{2i/(d-2)}}\right]^{-1} \frac{m/z}{2}\right) \tag{3.2.2}$$

The frequencies were selected such that wavelengths are logarithmically spaced from $\lambda_{\text{min}} = 10^{-2.5}$ Daltons to $\lambda_{\text{max}} = 10^{3.3}$ Daltons, aligning with the mass scales to be resolved. Similar to Equations 3.1.1-2, the sinusoidal peak embedding presented below:

$$PE_{\text{sin}}(m/z, I) = \text{FF}(\text{FF}(SE(m/z)) \parallel I) \tag{3.2.3}$$

The sinusoidally encoded spectra further processed using two fully connected feedforward neural networks (FFNs) to produce embeddings that represented spectral properties. These embeddings were then combined with intensity values, resulting in a single representation for further investigation. The encoded peaks are processed by a transformer encoder layer, which allows the model to self-attend and extract relational information between them. The encoder's output is combined with previously learned latent spectra and an encoded representation of the precursor. The precursor mass m_{prec} and charge c_{prec} are encoded using a sinusoidal encoding and an embedding layer, respectively, and then summed to generate the precursor embedding.

3.2.2 Contrastive Predictive Coding

In its raw form, mass spectra data are extremely noisy with missing peaks, which makes the extraction of meaningful features with traditional unsupervised techniques hard. We assumed that the probabilistic contrastive loss helps to learn these discriminative features, which are targeted to be captured in the latent space, to best predict the future samples. Figure 3.1 shows the structure of CPC models in detail. Initially, the input mass spectra of observations, x_t, is transformed by a nonlinear encoder, g_{enc} (mentioned in encoder section above), into a series of latent representations, $z_t = g_{\text{enc}}(x_t)$. Then an autoregressive model, g_{ar}, takes these latent representations up to time t and create a summarized representation with lower dimensional space. This process produces a contextual latent representation given by $c_t = g_{\text{ar}}(z \leq t)$.

Predicting future embedding is computationally expensive but CPC predict future on low dimensional latent space. This involves applying a model with density ratio that keeps the mutual information between x_{t+k} and c_t. The density ratio f is expressed as:

$$f_k(x_{t+k}, c_t) \propto p(x_{t+k}|c_t) \cdot p(x_{t+k}) \tag{3.2.4}$$

As a positive score function we choose a simple log-bilinear model as originally used by Oord et al. (2018):

$$f_k(x_{t+k}, c_t) = \exp\left(z_{t+k}^T W_k c_t\right) \tag{3.2.5}$$

Where W_k are a weights of predictor network and c_t are a context representation generated from the autoregressive model. We apply a linear transformation $W_k^T c_t$, using a different W_k for each step k to predict on the latent space. The transformer autoregressive model has been used to create a contextual representation.

The encoder module's output embedding is fed directly into the decoder transformer. The decoder transformer model that we used in this model doesn't have a cross-attention component instead focuses on creating coherent and contextually relevant output embedding.

3.2.3 Construction of Loss Function

The encoder and the autoregressive model trained together to improve a loss function called Noise-Contrastive Estimation (NCE) presented by Oord et al. (2018), InfoNCE in short. Given a set of random mass spectra instances $X = \{x_1, \ldots, x_N\}$, with one correct sample from $p(x_{t+k}|c_t)$ and $N-1$ incorrect samples from the 'proposal' distribution $p(x_{t+k})$, we aim to optimize the loss below:

$$L_N = -\mathbb{E}_X \left[\log \frac{f_k(x_{t+k}, c_t)}{\sum_{x_j \in X} f_k(x_j, c_t)}\right] \tag{3.2.6}$$

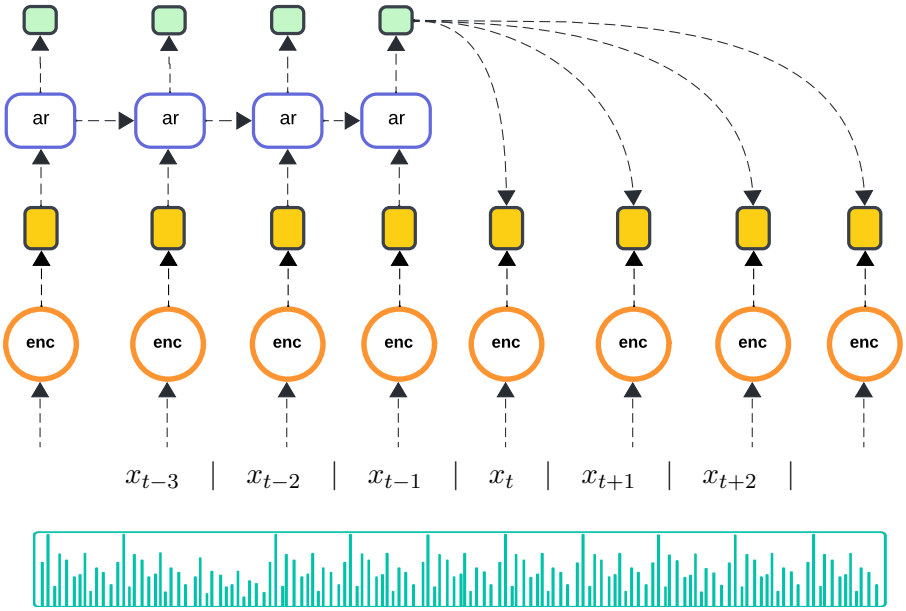


Figure 3.1: Overview of CPC for mass spectrometry representation learning

Optimizing this loss will result in $f_k(x_{t+k}, c_t)$ estimating the density ratio in equation (3.2.4). This can be shown as follows.

The loss in Equation (3.2.6) is the categorical cross-entropy of classifying the positive sample correctly, with $\frac{f_k}{\sum_X f_k}$ being the prediction of the model.

3.2.4 Training Procedure

Our model was trained on four Tesla T4 GPUs, taking advantage of fully sharded distributed approaches to maximize resource consumption (GPU memory) and efficiency. We used the Flax (Heek et al., 2023) and JAX (Bradbury et al., 2018) libraries to implement the neural network. Optax (DeepMind et al., 2020) library was used to define optimizer components. AdamW (Loshchilov and Hutter, 2017) with an initial learning rate of 10^{-4}. We also implemented a cosine warm up scheduler (Loshchilov and Hutter, 2016) and gradient clipping. The transformer model was implemented with model dimension of 64, but we also investigated the effects of model dimensions of 128 and 256. Each transformer used four attention heads, and we also investigated the effects of increasing the number of heads. The transformer was configured with a single layer. However, the implications of adding more layers were also studied. The model was trained with six species by keeping three species as held out to use them for evaluation letter on. Those species are *Candidatus endoloripes*, *Homo sapiens*, and *Mus musculus*.

3.3 Spectra *Library* Search

Spectra *library* search was performed using CPC and OpenMS methods. Our approach utilizes a transformer encoder pre-trained with Contrastive Predictive Coding (CPC) to perform database searches on mass spectra data. The summarized methodology is structured as follows (**Figure 3.2**): We pretrain the transformer encoder using a large volume of mass spectrometry data. We use the encoder layer to create an embedding for both database and query spectra followed by similarity metrics calculation between query and all spectrum embeddings. Then most similar peptide is filtered from the database.

Finally we compute the performance metrics followed by the method from (Eloff et al., 2023).

3.3.1 Mass Spectra Preparation

To provide robust validation of our model, we separated the dataset into two clearly defined subsets. The first subset, known as the 'Database,' contains randomly selected peptides from each PSM. The second subset, identified as the 'Query' set, encompasses the remaining spectra, and so acts as a specialized query dataset..

3.3.2 Inference

Both the database and query subsets are passed through the pre-trained transformer encoder to obtain their respective embeddings. These embeddings represent the spectra in a high-dimensional space, preserving the learned features from the CPC pretraining. We computed the cosine similarity between the query embeddings and the database embeddings. For each query in the query set, we calculate its cosine similarity with all the embeddings in the database. The mass spectra data was normalized cosine similarity was computed using the custom vectorized method.

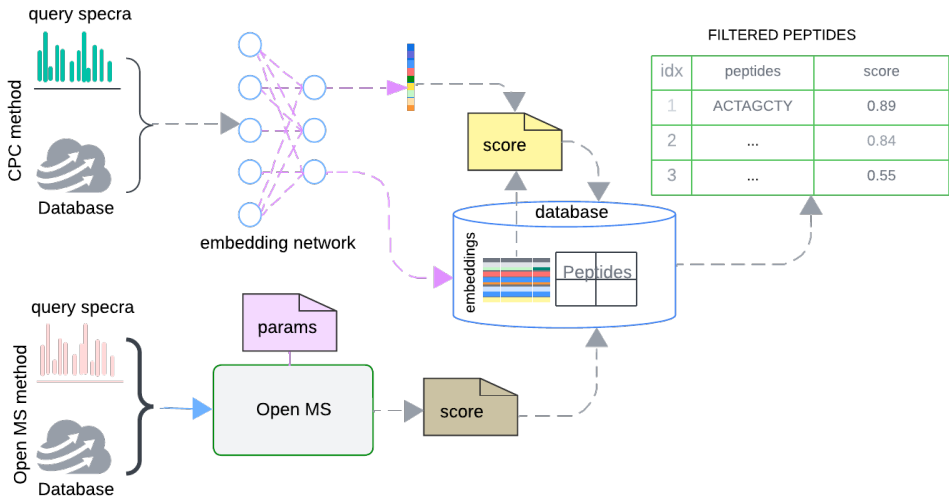


Figure 3.2: A general database search workflow we used to perform database search for both CPC and OpenMS

The performance of the database search was evaluated using metrics such as amino acid (AA) precision, AA recall and peptide recall. The unique peptide identifications from the database were cross-validated with known peptide sequences to assess the accuracy and reliability of our approach. Tnted he detail mechanism of spectra matching OpenMS library is presented in the next section.

3.3.3 Baseline

We compared our method with OpenMS tool (Pfeuffer et al., 2024), an open-source C++ library with Python bindings for managing, analyzing, and visualizing LC/MS data. It empowers rapid development of mass spectrometry related software (OpenMS, 2024). Spectra alignment corrected for instrumental variations using the OpenMS feature alignment algorithm, aligning peaks based on retention time and

m/z values.

$$\text{SpectraAlignmentScore} = \frac{\sum_{i=1}^{n} \text{intensity}_i^1 \times \text{intensity}_i^2}{\sqrt{\sum_{i=1}^{n}(\text{intensity}_i^1)^2 \times \sum_{i=1}^{n}(\text{intensity}_i^2)^2}} \tag{3.3.1}$$

where $intensity$ is a vector of intensity values from the spectra. Scoring employed a simple equation (3.2.1) integrated into OpenMS, enabling precise spectral comparison and ranking essential for accurate peptide identification and quantification.

3.3.4 Model Performance Evaluation Metrics

In our assessment of model predictions, we employ precision calculated at both the amino acid and peptide levels and peptide recall, following methodologies presented by previous works (Ma et al., 2003; Tran et al., 2017; Eloff et al., 2023; Yilmaz et al., 2022). These precision metrics serve as performance measures, gauging the quality of a given model's predictions based on coverage over the test set. For each spectrum, we compare the predicted sequence to the ground truth peptide obtained from the database search. Consistent with DeepNovo (Yilmaz et al., 2022), our approach to amino acid-level measures begins by calculating the number N_a^{match} of matched amino acid predictions. These are defined as predicted amino acids that (1) exhibit a mass difference of < 0.1 Da from the corresponding ground truth amino acid and (2) have either a prefix or suffix with a mass difference of no more than 0.5 Da from the corresponding amino acid sequence in the ground truth peptide. Amino acid-level precision is then defined as $\frac{N_a^{\text{match}}}{N_a^{\text{pred}}}$, where N_a^{pred} represents the number of predicted amino acids.

To summerize, we described our methods for training the mass spectra encoder using CPC and performing spectra *library* searches. We used a dataset of 2.8 million PSMs from nine PRIDE projects, consisting of 229,984 unique peptides. The dataset was split into training, validation, and test sets with ratios of 85%, 10%, and 5%, ensuring each unique peptide instance had at least two peptides in the validation set. The encoder utilized sinusoidal embedding to convert continuous *m/z* and intensity values into discrete representations suitable for transformers. For inference, database and validation subsets were passed through the pre-trained encoder to obtain embeddings, followed by cosine similarity computation to identify the most similar peptides. Model performance was evaluated using amino acid and peptide-level precision and recall metrics. Our method was compared to the OpenMS tool for baseline performance.

4. Result and Discussion

In this section we presented the performance comparison of our method in contrast to established baseline method (OpenMS) using 9-species-V2 datasets. The result showcase that CPC method achieved a higher performance metrics. We conducted further studies on the speed and efficiency comparison of the two methods. Also we reported about the architecture hyperparameter effect on the performance metrics of the CPC method on selected species.

4.1 Comparative performance evaluation

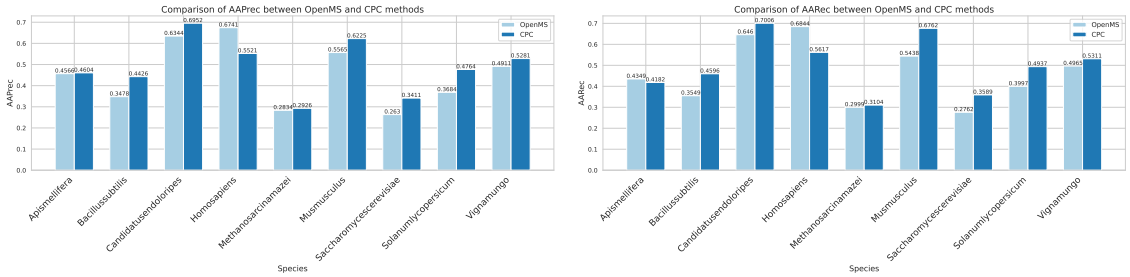
The nine species dataset V2 has been applied for several bench-marking tasks of peptide sequencing methods. It first introduced in DeepNovo's paper and has gained attention after that. In this research, our main objective was to show that CPC can learn very useful representation in comparison to OpenMS baseline. The detailed comparison is tabulated in Table 4.1. The result showcase that CPC outperforms on seven species, both at the amino acid level and peptide level metrics.

Table 4.1: Comparison of the performance of CPC method and OpenMS on 9-species-V2 test set. The bold font indicates the best performance.

Species	**OpenMS**			**CPC**		
	AA Prec	**AA Rec**	**Pep Rec**	**AA Prec**	**AA Rec**	**Pep Rec**
Apis mellifera	0.4566	**0.4349**	**0.3871**	**0.4604**	0.4182	0.3608
Bacillus subtilis	0.3478	0.3549	0.2993	**0.4426**	**0.4596**	**0.3883**
Candidatus endoloripes	0.6344	0.6460	0.6074	**0.6952**	**0.7006**	**0.6456**
Homo sapiens	**0.6741**	**0.6844**	**0.6132**	0.5521	0.5617	0.4933
Methanosarcina mazei	0.2834	0.2999	0.2446	**0.2926**	**0.3104**	**0.2893**
Mus musculus	0.5565	0.5438	0.5206	**0.6225**	**0.6762**	**0.6450**
Saccharomyces cerevisiae	0.2630	0.2762	0.2308	**0.3411**	**0.3589**	**0.3129**
Solanum lycopersicum	0.3684	0.3997	0.3636	**0.4764**	**0.4937**	**0.4490**
Vigna mungo	0.4911	0.4965	0.4856	**0.5281**	**0.5311**	**0.5101**
Average	**0.4528**	**0.4596**	**0.4169**	**0.4901**	**0.5011**	**0.4549**

On particular, precision at the average amino acid level and recall across seven species, CPC perform better in contrast to OpenMS by 3.73% and 4.15% respectively. In the field of proteomics, peptide-level recall is frequently of high interest. CPC method shows better performance over OpenMS on recall by 3.8%.

Regarding species level evaluation of the CPC model, it achieved a lowered performance in terms of amino acid (AA) recall and peptide recall for both *Apis mellifera* and *Homo sapiens* species. For *Apis mellifera*, the CPC model achieved a lower AA recall and peptide recall in contrast to OpenMS but still achieved comparable result. Also for *Homo sapiens*, the CPC model achieved a lowest performance metrics across all evaluated categories, including AA recall, peptide recall, precision in contrast to OpenMS. The lower performance of our model compared to OpenMS on *Homo sapiens* may be due to its absence in the training set. As a result, the network predicts peptides using representations from other species. OpenMS outperforms our model likely because it compares spectra peak by peak, not relying on shared features across species..



(a) AAPrec Comparison (b) AARec Comparison

Figure 4.1: Comparison of OpenMS and CPC methods for AAPrec and AARec.

Figure 4.1 and 4.2 depict the comparative performance of the CPC model across nine species, focusing on AA recall, AA precision, and Peptide recall metrics. The bar graphs highlight that CPC achieved the highest performance metrics on seven of the species evaluated that showcase a relevant features embedding learned by the model.

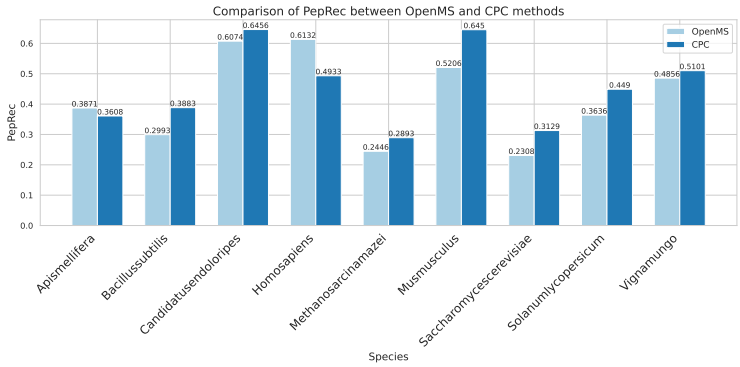
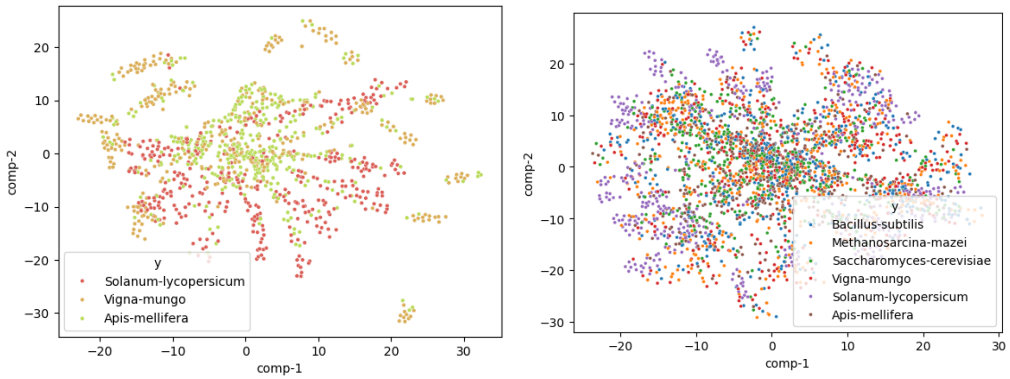


Figure 4.2: Comparison of OpenMS and CPC methods for PepRec.

We created an embedding for the samples selected from two groups of species, one with three species and the other with six species. Each sample passed through the trained encoder, and TSNE was applied to reduce the embedding to two dimensions.



(a) Embedding for the three selected species (b) Embedding for the six selected species

Figure 4.3: TSNE visualization of the learned embedding from the encoder.

Figure 4.3 shows TSNE plots created from the embeddings of mass spectra data for different species. The first figure displays data for three species: *Solanum lycopersicum*, *Vigna mungo*, and *Apis mellifera*. In this figure, *Solanum lycopersicum* and *Vigna mungo* appear tightly clustered, while *Apis mellifera* is a bit more scattered but still forms a distinct cluster. The second figure includes six species. Here, *Bacillus subtilis* is scattered, while the other species are generally clustered together, though not as tightly. These clusters suggest that each species might share some common evolutionary traits, as they sometimes cluster in the same area.

4.2 Comparative Inference Speed

We further evaluated the two methods on the inference speed for spectra *library* search. In order to perform a spectra *library* search using CPC, the input spectra has to be convert to embedding by passing them through the encoder. This can be run both on CPU and GPU architectures. We evaluated both method on same CPU architecture.

Table 4.2 showed that CPC outperforms OpenMS by significant marigin in terms of inference speed across four dataset sizes. CPC achieves faster inference times, measured in second, compared to OpenMS's times, which are in hours, for datasets ranging from 1000 to 10,000 spectra. For OpenMS the time required to perform matching explodes as larger dataset size which makes it inapplicable at larger dataset size. OpenMS spectra matching, each comparison between spectra involves a peak-by-peak comparison, where M spectra from the dataset are compared against N spectra in the database. We can hypothesize that time complexity of assuming each spectrum has an average of P_1 peaks in the dataset and P_2 peaks in the database, the time complexity for this operation is $O(M \times N \times P_1 \times P_2)$ which is significantly larger than time complexity of matrix multiplication on the optimized linear algebra libraries.

Table 4.2: Inference Speed Comparison between OpenMS and CPC

Dataset Size	OpenMS (time in minutes)	CPC (time in seconds)
1000	5 min	3.23 sec
1500	7 min	3.78 sec
3000	11 min	5.15 sec
10000	$<$ 240 min	15.71 sec

In contrast, CPC employs an embedding-based approach where spectra are represented as embeddings, and similarity is computed using matrix multiplication, leveraging the highly parallelizable nature of linear algebra operations. This parallel computation significantly accelerates inference speeds, making CPC more efficient for large-scale proteomic data analysis.

4.3 Hyperparameter evaluation

We conducted a hyperparameter study to evaluate the performance of a transformer model on the species *Candidatus endoloripes*. The hyperparameters varied include model dimension (64, 128, 256), number of layers (1, 2, 3, 4, 5, 6), and number of attention heads (4, 8, 16). Our findings indicate that increasing these hyperparameters generally leads to a decrease in model performance metrics, specifically in terms of recall. Also **Table 4.3** shows that as we increase the model dimension, number of layers, and

number of attention heads, the recall metric decreases consistently. This suggests that for *Candidatus endoloripes*, a simpler model configuration (e.g., lower dimensions, fewer layers, and heads) might be more effective in capturing relevant features in the mass spectra data.

Table 4.3: Effect of hyperparameter on a *Candidatus endoloripes* species peptide recall

Model Dimension	Layers	Heads	Peptide Recall
64	1	4	0.645
64	2	8	0.621
64	3	16	0.432
128	4	4	0.356
128	5	8	0.301
128	6	16	0.284
256	3	4	0.221
256	4	8	0.201
256	5	16	0.115

The peptide recall was very sensitive to two specific hyperparameters, namely the number of heads and the model dimension. Specifically, the number of heads had a significant effect. As we can see from **Table 4.3** in the last two rows, the peptide recall dropped from 0.201 to 0.115 when we increased the number of heads from 8 to 16 while only increasing the layer by 1. This implies that a low-dimensional model for mass spectra pre-training is recommended. But it might depend on the nature of the dataset so it requires a study to better understand the effect.

5. Summary & Conclusion

Despite their advanced performance, deep-learning based *de novo* peptide sequencing faces challenges. Thse challenges mostly originated from the low frequency PTMs in training data and noisy nature of mass spectra that poses a constraint on accurate identification of peptide sequences. Unsupervised learning can enhance *de novo* peptide sequencing architectures by learning a massive amount of mass spectra data. This research has laid the foundation for effective pre-training of mass spectra encoder using CPC. We first assumed that despite the intricate properties of the mass spectra, CPC can learn the representation that could maximizes mutual information between the context representation and the actual data. This way the method allows the encoder to learn useful representations from unlabeled mass spectra data.

We compared our pretrained encoder with OpenMS for two evaluation cases: 1) for spectra *library* search, 2) inference speed comparison. On the first case CPC outperform on seven species on all categories of metrics. Specifically, at the average amino acid level, CPC demonstrated higher precision and recall across seven species, surpassing OpenMS by 3.73% and 4.15%, respectively. It also attained a high peptide-level recall, outperforming OpenMS by a margin of 3.8%. We conducted inference speed comparisons between both methods on the same compute architecture (CPU) and found that CPC significantly outperforms OpenMS. CPC achieves faster inference times in seconds, while OpenMS requires hours as dataset sizes scale from 1000 to 10,000 spectra. We also conducted hyperparameter evaluations for number of layer, model dimension and number of head. The result indicated that increasing these parameters generally leads to a decrease in model performance metrics, particularly in terms of recall. Increasing these hyperparameters cause the model to learn noise from the data which leads to reduced generalization ability of the model. We also found out that the model performance was highly sensitive to number of head. This suggest that small dimension learn better representation that the big model.

Pre-training deep learning-based mass spectra encoders from extensive unlabeled data is a relatively novel approach. While CPC shows promising performance, there are still challenges, particularly due to the nature of the mass spectra. More specifically the mass spectra data contains diverse noise types and missing peaks which is difficult to learn sequential patterns between the peaks. Even if our method outperform the OpenMS method, its performance was not satisfactory. There are several uncertainty arise from the nature of the mass spectra data. Future research could find a way to use probabilistic models such as Variational Autoencoders (VAEs) and latent-based diffusion models as a part of the pre-training pipeline. Probabilistic models like VAE can handle uncertainty in the data that allow them to effectively learn latent representations and to model and impute missing data points.

Acknowledgements

I want to acknowledge AIMS and it's funders for supporting this work, as well as my supervisors, Kevin Eloff, Amandla Mabona, Jeroen Van Goey from InstaDeep, South Africa.

References

Aebersold, R. and Mann, M. Mass spectrometry-based proteomics. *Nature*, 422(6928):198–207, 2003.

Alley, E. C., Khimulya, G., Biswas, S., AlQuraishi, M., and Church, G. M. Unified rational protein engineering with sequence-based deep representation learning. *Nature methods*, 16(12):1315–1322, 2019.

Atal, B. S. and Schroeder, M. R. Adaptive predictive coding of speech signals. *Bell System Technical Journal*, 49(8):1973–1986, 1970.

Ba, J. L., Kiros, J. R., and Hinton, G. E. Layer normalization. *arXiv preprint arXiv:1607.06450*, 2016.

Bassani-Sternberg, M., Bräunlein, E., Klar, R., Engleitner, T., Sinitcyn, P., Audehm, S., Straub, M., Weber, J., Slotta-Huspenina, J., Specht, K., et al. Direct identification of clinically relevant neoepitopes presented on native human melanoma tissue by mass spectrometry. *Nature communications*, 7 (1):13404, 2016.

Belghazi, M. I., Baratin, A., Rajeswar, S., Ozair, S., Bengio, Y., Courville, A., and Hjelm, R. D. Mine: mutual information neural estimation. *arXiv preprint arXiv:1801.04062*, 2018.

Bell, A. J. and Sejnowski, T. J. An information-maximization approach to blind separation and blind deconvolution. *Neural computation*, 7(6):1129–1159, 1995.

Belouah, I., Blein-Nicolas, M., Balliau, T., Gibon, Y., Zivy, M., and Colombié, S. Peptide filtering differently affects the performances of xic-based quantification methods. *Journal of proteomics*, 193: 131–141, 2019.

Benitez-Amaro, A., Pallara, C., Nasarre, L., Rivas-Urbina, A., Benitez, S., Vea, A., Bornachea, O., de Gonzalo-Calvo, D., Serra-Mir, G., Villegas, S., et al. Molecular basis for the protective effects of low-density lipoprotein receptor-related protein 1 (lrp1)-derived peptides against ldl aggregation. *Biochimica et Biophysica Acta (BBA)-Biomembranes*, 1861(7):1302–1316, 2019.

Bepler, T. and Berger, B. Learning protein sequence embeddings using information from structure. *arXiv preprint arXiv:1902.08661*, 2019.

Bittremieux, W., May, D. H., Bilmes, J., and Noble, W. S. A learned embedding for efficient joint analysis of millions of mass spectra. *Nature methods*, 19(6):675–678, 2022.

Bouatta, N., Sorger, P., and AlQuraishi, M. Protein structure prediction by alphafold2: are attention and symmetries all you need? *Acta Crystallographica Section D: Structural Biology*, 77(8):982–991, 2021.

Bradbury, J., Frostig, R., Hawkins, P., Johnson, M. J., Leary, C., Maclaurin, D., Necula, G., Paszke, A., VanderPlas, J., Wanderman-Milne, S., and Zhang, Q. JAX: composable transformations of Python+NumPy programs, 2018. URL http://github.com/google/jax.

Brosch, M., Yu, L., Hubbard, T., and Choudhary, J. Accurate and sensitive peptide identification with mascot percolator. *Journal of proteome research*, 8(6):3176–3181, 2009.

Cao, K., Wei, C., Gaidon, A., Arechiga, N., and Ma, T. Learning imbalanced datasets with label-distribution-aware margin loss. *Advances in neural information processing systems*, 32, 2019.

Cao, Y., Geddes, T. A., Yang, J. Y. H., and Yang, P. Ensemble deep learning in bioinformatics. *Nature Machine Intelligence*, 2(9):500–508, 2020.

Chen, T., Kao, M.-Y., Tepel, M., Rush, J., and Church, G. M. A dynamic programming approach to de novo peptide sequencing via tandem mass spectrometry. *Journal of Computational Biology*, 8(3): 325–337, 2001.

Chen, T., Kornblith, S., Norouzi, M., and Hinton, G. A simple framework for contrastive learning of visual representations. In *International conference on machine learning*, pages 1597–1607. PMLR, 2020.

Chi, H., Sun, R.-X., Yang, B., Song, C.-Q., Wang, L.-H., Liu, C., Fu, Y., Yuan, Z.-F., Wang, H.-P., He, S.-M., et al. pnovo: de novo peptide sequencing and identification using hcd spectra. *Journal of proteome research*, 9(5):2713–2724, 2010.

Dančík, V., Addona, T. A., Clauser, K. R., Vath, J. E., and Pevzner, P. A. De novo peptide sequencing via tandem mass spectrometry. *Journal of computational biology*, 6(3-4):327–342, 1999.

DeepMind, Babuschkin, I., Baumli, K., Bell, A., Bhupatiraju, S., Bruce, J., Buchlovsky, P., Budden, D., Cai, T., Clark, A., Danihelka, I., Dedieu, A., Fantacci, C., Godwin, J., Jones, C., Hemsley, R., Hennigan, T., Hessel, M., Hou, S., Kapturowski, S., Keck, T., Kemaev, I., King, M., Kunesch, M., Martens, L., Merzic, H., Mikulik, V., Norman, T., Papamakarios, G., Quan, J., Ring, R., Ruiz, F., Sanchez, A., Sartran, L., Schneider, R., Sezener, E., Spencer, S., Srinivasan, S., Stanojević, M., Stokowiec, W., Wang, L., Zhou, G., and Viola, F. The DeepMind JAX Ecosystem, 2020. URL http://github.com/google-deepmind.

Duncan, D. T., Craig, R., and Link, A. J. Parallel tandem: a program for parallel processing of tandem mass spectra using pvm or mpi and x! tandem. *Journal of proteome research*, 4(5):1842–1847, 2005.

Elias, P. Predictive coding–i. *IRE transactions on information theory*, 1(1):16–24, 1955.

Eloff, K., Kalogeropoulos, K., Morell, O., Mabona, A., Jespersen, J. B., Williams, W., van Beljouw, S. P., Skwark, M., Laustsen, A. H., Brouns, S. J., et al. De novo peptide sequencing with InstaNovo: Accurate, database-free peptide identification for large scale proteomics experiments. *bioRxiv*, pages 2023–08, 2023.

Eng, J. K., Jahan, T. A., and Hoopmann, M. R. Comet: an open-source ms/ms sequence database search tool. *Proteomics*, 13(1):22–24, 2013.

Faridi, P., Li, C., Ramarathinam, S. H., Vivian, J. P., Illing, P. T., Mifsud, N. A., Ayala, R., Song, J., Gearing, L. J., Hertzog, P. J., et al. A subset of hla-i peptides are not genomically templated: Evidence for cis-and trans-spliced peptide ligands. *Science immunology*, 3(28):eaar3947, 2018.

Fischer, B., Roth, V., Roos, F., Grossmann, J., Baginsky, S., Widmayer, P., Gruissem, W., and Buhmann, J. M. Novohmm: a hidden markov model for de novo peptide sequencing. *Analytical chemistry*, 77 (22):7265–7273, 2005.

Frank, A. and Pevzner, P. Pepnovo: de novo peptide sequencing via probabilistic network modeling. *Analytical chemistry*, 77(4):964–973, 2005.

Friston, K. A theory of cortical responses. *Philosophical transactions of the Royal Society B: Biological sciences*, 360(1456):815–836, 2005.

Geer, L. Y., Markey, S. P., Kowalak, J. A., Wagner, L., Xu, M., Maynard, D. M., Yang, X., Shi, W., and Bryant, S. H. Open mass spectrometry search algorithm. *Journal of proteome research*, 3(5): 958–964, 2004.

Gessulat, S., Schmidt, T., Zolg, D. P., Samaras, P., Schnatbaum, K., Zerweck, J., Knaute, T., Rechenberger, J., Delanghe, B., Huhmer, A., et al. Prosit: proteome-wide prediction of peptide tandem mass spectra by deep learning. *Nature methods*, 16(6):509–518, 2019.

He, K., Zhang, X., Ren, S., and Sun, J. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

Heek, J., Levskaya, A., Oliver, A., Ritter, M., Rondepierre, B., Steiner, A., and van Zee, M. Flax: A neural network library and ecosystem for JAX, 2023. URL http://github.com/google/flax.

Hjelm, R. D., Fedorov, A., Lavoie-Marchildon, S., Grewal, K., Bachman, P., Trischler, A., and Bengio, Y. Learning deep representations by mutual information estimation and maximization. *arXiv preprint arXiv:1808.06670*, 2018.

Hochreiter, S. and Schmidhuber, J. Long short-term memory. *Neural computation*, 9(8):1735–1780, 1997.

Huang, C., Li, Y., Loy, C. C., and Tang, X. Learning deep representation for imbalanced classification. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5375–5384, 2016.

Huang, C., Li, Y., Loy, C. C., and Tang, X. Deep imbalanced learning for face recognition and attribute prediction. *IEEE transactions on pattern analysis and machine intelligence*, 42(11):2781–2794, 2019.

Huber, F., van der Burg, S., van der Hooft, J. J., and Ridder, L. Ms2deepscore: a novel deep learning similarity measure to compare tandem mass spectra. *Journal of cheminformatics*, 13(1):84, 2021.

Ioffe, S. and Szegedy, C. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *International conference on machine learning*, pages 448–456. pmlr, 2015.

Jin, Z., Xu, S., Zhang, X., Ling, T., Dong, N., Ouyang, W., Gao, Z., Chang, C., and Sun, S. Contranovo: A contrastive learning approach to enhance de novo peptide sequencing. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 144–152, 2024.

Karunratanakul, K., Tang, H.-Y., Speicher, D. W., Chuangsuwanich, E., and Sriswasdi, S. Uncovering thousands of new peptides with sequence-mask-search hybrid de novo peptide sequencing framework. *Molecular & Cellular Proteomics*, 18(12):2478–2491, 2019.

Kraskov, A., Stögbauer, H., and Grassberger, P. Estimating mutual information. *Physical review E*, 69 (6):066138, 2004.

Laumont, C. M., Vincent, K., Hesnard, L., Audemard, É., Bonneil, É., Laverdure, J.-P., Gendron, P., Courcelles, M., Hardy, M.-P., Côté, C., et al. Noncoding regions are the main source of targetable tumor-specific antigens. *Science translational medicine*, 10(470):eaau5516, 2018.

LeCun, Y., Bengio, Y., et al. Convolutional networks for images, speech, and time series. *The handbook of brain theory and neural networks*, 3361(10):1995, 1995.

LeCun, Y., Bengio, Y., and Hinton, G. Deep learning. *nature*, 521(7553):436–444, 2015.

Lin, E., Lin, C.-H., and Lane, H.-Y. Relevant applications of generative adversarial networks in drug design and discovery: molecular de novo design, dimensionality reduction, and de novo peptide and protein design. *Molecules*, 25(14):3250, 2020.

Lin, T., Wang, Y., Liu, X., and Qiu, X. A survey of transformers. *AI open*, 3:111–132, 2022.

Linsker, R. Self-organization in a perceptual network. *Computer*, 21(3):105–117, 1988.

Loshchilov, I. and Hutter, F. Sgdr: Stochastic gradient descent with warm restarts. *arXiv preprint arXiv:1608.03983*, 2016.

Loshchilov, I. and Hutter, F. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017.

Ma, B. Challenges in computational analysis of mass spectrometry data for proteomics. *Journal of Computer Science and Technology*, 25(1):107–123, 2010.

Ma, B. Novor: real-time peptide de novo sequencing software. *Journal of the American Society for Mass Spectrometry*, 26(11):1885–1894, 2015.

Ma, B. and Johnson, R. De novo sequencing and homology searching. *Molecular & cellular proteomics*, 11(2), 2012.

Ma, B., Zhang, K., Hendrie, C., Liang, C., Li, M., Doherty-Kirby, A., and Lajoie, G. Peaks: powerful software for peptide de novo sequencing by tandem mass spectrometry. *Rapid communications in mass spectrometry*, 17(20):2337–2342, 2003.

Ma, B., Zhang, K., and Liang, C. An effective algorithm for peptide de novo sequencing from ms/ms spectra. *Journal of Computer and System Sciences*, 70(3):418–430, 2005.

MacCoss, M. J., Wu, C. C., and Yates, J. R. Probability-based validation of protein identifications using a modified sequest algorithm. *Analytical chemistry*, 74(21):5593–5599, 2002.

Mann, M. and Jensen, O. N. Proteomic analysis of post-translational modifications. *Nature biotechnology*, 21(3):255–261, 2003.

Mao, J., Huang, J., Toshev, A., Camburu, O., Yuille, A. L., and Murphy, K. Generation and comprehension of unambiguous object descriptions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 11–20, 2016.

Nguyen, X., Wainwright, M. J., and Jordan, M. I. Estimating divergence functionals and the likelihood ratio by convex risk minimization. *IEEE Transactions on Information Theory*, 56(11):5847–5861, 2010.

Noor, Z., Ahn, S. B., Baker, M. S., Ranganathan, S., and Mohamedali, A. Mass spectrometry–based protein identification in proteomics—a review. *Briefings in bioinformatics*, 22(2):1620–1638, 2021.

Novak, R., Bahri, Y., Abolafia, D. A., Pennington, J., and Sohl-Dickstein, J. Sensitivity and generalization in neural networks: an empirical study. *arXiv preprint arXiv:1802.08760*, 2018.

Oord, A. v. d., Li, Y., and Vinyals, O. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*, 2018.

OpenMS. OpenMS. https://openms.de/, 2024. Accessed: June 16, 2024.

Pfeuffer, J., Bielow, C., Wein, S., Jeong, K., Netz, E., Walter, A., Alka, O., Nilse, L., Colaianni, P. D., McCloskey, D., et al. Openms 3 enables reproducible analysis of large-scale mass spectrometry data. *Nature Methods*, pages 1–3, 2024.

Poole, B., Ozair, S., Van Den Oord, A., Alemi, A., and Tucker, G. On variational bounds of mutual information. In *International Conference on Machine Learning*, pages 5171–5180. PMLR, 2019.

Qi, C. R., Su, H., Mo, K., and Guibas, L. J. Pointnet: Deep learning on point sets for 3d classification and segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 652–660, 2017.

Qiao, R., Tran, N. H., Xin, L., Chen, X., Li, M., Shan, B., and Ghodsi, A. Complete de novo assembly of monoclonal antibody sequences. *Scientific Reports*, 6(1), 2016. doi: $10.1038/\mathrm{srep31730}$.

Qiao, R., Tran, N. H., Xin, L., Chen, X., Li, M., Shan, B., and Ghodsi, A. Computationally instrument-resolution-independent de novo peptide sequencing for high-resolution devices. *Nature Machine Intelligence*, 3(5):420–425, 2021.

Qin, C., Luo, X., Deng, C., Shu, K., Zhu, W., Griss, J., Hermjakob, H., Bai, M., and Perez-Riverol, Y. Deep learning embedder method and tool for mass spectra similarity search. *Journal of proteomics*, 232:104070, 2021.

Quesada, D., Bielza, C., and Larrañaga, P. Structure learning of high-order dynamic bayesian networks via particle swarm optimization with order invariant encoding. In *International Conference on Hybrid Artificial Intelligence Systems*, pages 158–171. Springer, 2021.

Rao, R. P. and Ballard, D. H. Predictive coding in the visual cortex: a functional interpretation of some extra-classical receptive-field effects. *Nature neuroscience*, 2(1):79–87, 1999.

Schweppe, D. K., Eng, J. K., Yu, Q., Bailey, D., Rad, R., Navarrete-Perea, J., Huttlin, E. L., Erickson, B. K., Paulo, J. A., and Gygi, S. P. Full-featured, real-time database searching platform enables fast and accurate multiplexed quantitative proteomics. *Journal of proteome research*, 19(5):2026–2034, 2020.

Shears, M. J., Sekhar Nirujogi, R., Swearingen, K. E., Renuse, S., Mishra, S., Jaipal Reddy, P., Moritz, R. L., Pandey, A., and Sinnis, P. Proteomic analysis of plasmodium merosomes: the link between liver and blood stages in malaria. *Journal of proteome research*, 18(9):3404–3418, 2019.

Sim, S. Y., Choi, Y. R., Lee, J. H., Lim, J. M., Lee, S.-E., Kim, K. P., Kim, J. Y., Lee, S. H., and Kim, M.-S. In-depth proteomic analysis of human bronchoalveolar lavage fluid toward the biomarker discovery for lung cancers. *PROTEOMICS–Clinical Applications*, 13(5):1900028, 2019.

Sobolesky, P., Parry, C., Boxall, B., Wells, R., Venn-Watson, S., and Janech, M. G. Proteomic analysis of non-depleted serum proteins from bottlenose dolphins uncovers a high vanin-1 phenotype. *Scientific reports*, 6(1):33879, 2016.

Song, J. and Ermon, S. Understanding the limitations of variational mutual information estimators. *arXiv preprint arXiv:1910.06222*, 2019.

Song, J. and Ermon, S. Multi-label contrastive predictive coding. *Advances in Neural Information Processing Systems*, 33:8161–8173, 2020.

Taylor, J. A. and Johnson, R. S. Implementation and uses of automated de novo peptide sequencing by tandem mass spectrometry. *Analytical chemistry*, 73(11):2594–2604, 2001.

Teo, G. C., Polasky, D. A., Yu, F., and Nesvizhskii, A. I. Fast deisotoping algorithm and its implementation in the msfragger search engine. *Journal of proteome research*, 20(1):498–505, 2020.

Tran, N., Qiao, R., Xin, L., Chen, X., Shan, B., and Li, M. Identifying neoantigens for cancer vaccines by personalized deep learning of individual immunopeptidomes. *Nat. Mach. Intell*, 2:764–771, 2019a.

Tran, N. H., Rahman, M. Z., He, L., Xin, L., Shan, B., and Li, M. Complete de novo assembly of monoclonal antibody sequences. *Scientific reports*, 6(1):31730, 2016.

Tran, N. H., Zhang, X., Xin, L., Shan, B., and Li, M. De novo peptide sequencing by deep learning. *Proceedings of the National Academy of Sciences*, 114(31):8247–8252, 2017.

Tran, N. H., Qiao, R., Xin, L., Chen, X., Liu, C., Zhang, X., Shan, B., Ghodsi, A., and Li, M. Deep learning enables de novo peptide sequencing from data-independent-acquisition mass spectrometry. *Nature methods*, 16(1):63–66, 2019b.

Van den Oord, A., Kalchbrenner, N., Espeholt, L., Vinyals, O., Graves, A., et al. Conditional image generation with pixelcnn decoders. *Advances in neural information processing systems*, 29, 2016.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., and Polosukhin, I. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

Vizcaıno, J. A., Csordas, A., del Toro, N., Dianes, J. A., Griss, J., Lavidas, I., Mayer, G., Perez-Riverol, Y., Reisinger, F., Ternent, T., et al. 2016 update of the pride database and its related tools. 2016.

Voronov, G., Lightheart, R., Davison, J., Krettler, C. A., Healey, D., and Butler, T. Multi-scale sinusoidal embeddings enable learning on high resolution mass spectrometry data. *arXiv preprint arXiv:2207.02980*, 2022.

Wang, M., Jarmusch, A. K., Vargas, F., Aksenov, A. A., Gauglitz, J. M., Weldon, K., Petras, D., da Silva, R., Quinn, R., Melnik, A. V., et al. Mass spectrometry searches using masst. *Nature biotechnology*, 38(1):23–26, 2020.

Wolters, D. A., Washburn, M. P., and Yates, J. R. An automated multidimensional protein identification technology for shotgun proteomics. *Analytical chemistry*, 73(23):5683–5690, 2001.

Wu, Z., Xiong, Y., Yu, S. X., and Lin, D. Unsupervised feature learning via non-parametric instance discrimination. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3733–3742, 2018.

Wyner, A. D. A definition of conditional mutual information for arbitrary ensembles. *Information and Control*, 38(1):51–59, 1978.

Xia, J., Chen, S., Zhou, J., Lin, T., Du, W., Liu, S., and Li, S. Z. Adanovo: Adaptive\emph {De Novo} peptide sequencing with conditional mutual information. *arXiv preprint arXiv:2403.07013*, 2024.

Xu, H. and Freitas, M. A. Massmatrix: A database search program for rapid characterization of proteins and peptides from tandem mass spectrometry data. *PROTEOMICS*, 9(6):1548–1555, 2009. doi: https://doi.org/10.1002/pmic.200700322. URL https://analyticalsciencejournals.onlinelibrary.wiley.com/doi/abs/10.1002/pmic.200700322.

Yang, H., Chi, H., Zeng, W.-F., Zhou, W.-J., and He, S.-M. pnovo 3: precise de novo peptide sequencing using a learning-to-rank framework. *Bioinformatics*, 35(14):i183–i190, 2019.

Yilmaz, M., Fondrie, W., Bittremieux, W., Oh, S., and Noble, W. S. De novo mass spectrometry peptide sequencing with a transformer model. In *International Conference on Machine Learning*, pages 25514–25522. PMLR, 2022.

Zeng, W.-F., Zhou, X.-X., Zhou, W.-J., Chi, H., Zhan, J., and He, S.-M. Ms/ms spectrum prediction for modified peptides using pdeep2 trained by transfer learning. *Analytical chemistry*, 91(15):9724–9731, 2019.

Zhang, Z., Xu, M., Jamasb, A., Chenthamarakshan, V., Lozano, A., Das, P., and Tang, J. Protein representation learning by geometric structure pretraining. *arXiv preprint arXiv:2203.06125*, 2022.

Zhou, X.-X., Zeng, W.-F., Chi, H., Luo, C., Liu, C., Zhan, J., He, S.-M., and Zhang, Z. pdeep: predicting ms/ms spectra of peptides with deep learning. *Analytical chemistry*, 89(23):12690–12697, 2017.

Zhu, Y., Dou, M., Piehowski, P. D., Liang, Y., Wang, F., Chu, R. K., Chrisler, W. B., Smith, J. N., Schwarz, K. C., Shen, Y., et al. Spatially resolved proteome mapping of laser capture microdissected tissue with automated sample transfer to nanodroplets. *Molecular & Cellular Proteomics*, 17(9): 1864–1874, 2018.

Appendix A. Proofs

A.1 Preliminary Lemma and Propositions

To prove the main results, we need the following Lemma and Propositions 1 and 2. The Lemma is a special case of the dual representation of f-divergences discussed in Nguyen et al. (2010).

Lemma 1 ((Nguyen et al., 2010)). *$\forall P, Q \in \mathcal{P}(X)$ such that $P \ll Q$,*

$$D_{\mathrm{KL}}(P\|Q) = \sup_{T \in L^\infty(Q)} \mathbb{E}_P[T] - \mathbb{E}_Q[e^T] + 1$$

Proof (Sketch). Please refer to (Nguyen et al., 2010) for a more formal proof. Denote $f(t) = t \log t$ whose convex conjugate is $f^*(u) = \exp(u-1)$, we have that

$$\begin{aligned} D_{\mathrm{KL}}(P\|Q) &= \mathbb{E}_{x\sim Q}\left[f\left(\frac{dP}{dQ}(x)\right)\right] = \mathbb{E}_{x\sim Q}\left[\sup_u \left(u \cdot \frac{dP}{dQ}(x) - f^*(u)\right)\right] \\ &= \sup_{T \in L^\infty(Q)} \mathbb{E}_{x\sim Q}\left[T(x) \cdot \frac{dP}{dQ}(x) - f^*(T(x))\right] \\ &= \sup_{T \in L^\infty(Q)} \mathbb{E}_P[T(x)] - \mathbb{E}_Q[f^*(T(x))] \\ &= \sup_{T \in L^\infty(Q)} \mathbb{E}_P[T(x)+1] - \mathbb{E}_Q[\exp(T(x))] \end{aligned}$$

which completes the proof. □

A.1.1 Preliminary Lemma and Propositions

To prove the main results, we need the following Lemma and Propositions 1 and 2. The Lemma is a special case of the dual representation of f-divergences discussed in (Nguyen et al., 2010).

Lemma 2 (Nguyen et al. (Nguyen et al., 2010)). *$\forall P, Q \in \mathcal{P}(X)$ such that $P \ll Q$,*

$$D_{\mathrm{KL}}(P\|Q) = \sup_{T \in L^\infty(Q)} \mathbb{E}_P[T] - \mathbb{E}_Q[e^T] + 1$$

Proof (Sketch). Please refer to (Nguyen et al., 2010) for a more formal proof. Denote $f(t) = t \log t$ whose convex conjugate is $f^*(u) = \exp(u-1)$, we have that

$$\begin{aligned} D_{\mathrm{KL}}(P\|Q) &= \mathbb{E}_{x\sim Q}\left[f\left(\frac{dP}{dQ}(x)\right)\right] = \mathbb{E}_{x\sim Q}\left[\sup_u \left(u \cdot \frac{dP}{dQ}(x) - f^*(u)\right)\right] \\ &= \sup_{T \in L^\infty(Q)} \mathbb{E}_{x\sim Q}\left[T(x) \cdot \frac{dP}{dQ}(x) - f^*(T(x))\right] \\ &= \sup_{T \in L^\infty(Q)} \mathbb{E}_P[T(x)] - \mathbb{E}_Q[f^*(T(x))] \\ &= \sup_{T \in L^\infty(Q)} \mathbb{E}_P[T(x)+1] - \mathbb{E}_Q[\exp(T(x))] \end{aligned}$$

which completes the proof. □

Proposition 1. *For all positive integers $n \geq 1$, $m \geq 2$, and for any collection of positive random variables $\{X_i\}_{i=1}^n, \{X_{i,j}\}_{j=1}^m$ such that $\forall i \in [n], X_i, X_{i,1}, X_{i,2}, \ldots, X_{i,m-1}$ are exchangeable, then $\forall \alpha \in \left(0, \frac{2m}{m+1}\right]$, the following is true:*

$$\mathbb{E}\left[\frac{1}{n}\sum_{i=1}^{n}\frac{mX_i}{\alpha X_i + (m-\alpha)\sum_{j=1}^{m-1}X_{i,j}}\right] \leq \frac{1}{\alpha}.$$

Proof. First, for $\alpha \in \left(0, \frac{2m}{m+1}\right]$ we have:

$$\begin{aligned}
&n\mathbb{E}\left[\frac{1}{n}\sum_{i=1}^{n}\frac{mX_i}{\alpha X_i + (m-\alpha)\sum_{j=1}^{m-1}X_{i,j}}\right] \\
&= \mathbb{E}\left[\sum_{i=1}^{n}\frac{m(m-1)}{m-\alpha}X_i\left(\Sigma_i\right)^{-1}\left(1-\frac{m-1}{m-\alpha}\alpha\right)\right] \\
&= \mathbb{E}\left[\sum_{i=1}^{n}\frac{m(m-1)}{m-\alpha}\Sigma_i/X_i - \left(1-\frac{m-1}{m-\alpha}\alpha\right)\right] \\
&= \frac{m(m-1)}{m-\alpha}\mathbb{E}\left[\sum_{i=1}^{n}\sum_{p=0}^{\infty}\left(\frac{X_i}{\Sigma_i}\right)^{p+1}\left(1-\frac{m-1}{m-\alpha}\alpha\right)^p\right] \quad \text{(Taylor expansion)} \\
&= \frac{m(m-1)}{m-\alpha}\sum_{i=1}^{n}\sum_{p=0}^{\infty}\mathbb{E}\left[\left(\frac{X_i}{\Sigma_i}\right)^{p+1}\right]\left(1-\frac{m-1}{m-\alpha}\alpha\right)^p
\end{aligned}$$

where we simplify the notation with $\Sigma_i := X_i + \sum_{j=1}^{m-1} X_{i,j}$. Furthermore, we note that the Taylor series converges because $\left(1-\frac{m-1}{m-\alpha}\alpha\right) \in (-1,1)$. Since the random variables are exchangeable, switching the ordering of $X_i, X_{i,1}, \ldots, X_{i,m-1}$ does not affect the joint distribution, and the summing function is permutation invariant. Therefore, for all... □

Proof. For $i \in [n]$, $p \geq 0$,

$$\mathbb{E}\left[\left(\frac{X_i}{\Sigma_i}\right)^{p+1}\right] = \frac{1}{m}\mathbb{E}\left[\left(\frac{X_i}{\Sigma_i}\right)^{p+1} + \sum_{j=1}^{m-1}\left(\frac{X_{i,j}}{\Sigma_i}\right)^{p+1}\right] \leq \frac{1}{m}\mathbb{E}\left[\left(\frac{X_i + \sum_{j=1}^{m-1}X_{i,j}}{\Sigma_i}\right)^{p+1}\right] = \frac{1}{m},$$

where the last inequality comes from the fact that $\frac{X_i + \sum_{j=1}^{m-1}X_{i,j}}{\Sigma_i} = 1$ and all the random variables are positive. Continuing from Eq.(19), we have:

$$n\mathbb{E}\left[\frac{1}{n}\sum_{i=1}^{n}\frac{mX_i}{\alpha X_i + (m-\alpha)\sum_{j=1}^{m-1}X_{i,j}}\right] \leq \frac{nm(m-1)}{m-\alpha}\sum_{p=0}^{\infty}\frac{1}{m}\left(1-\frac{m-1}{m-\alpha}\alpha\right)^p = \frac{m-1}{m-\alpha}n\alpha\frac{m-1}{m-\alpha} = \frac{n}{\alpha}.$$

Dividing both sides by n completes the proof for $\alpha \in (0, \frac{2m}{m+1}]$.

The case for $\alpha \in [1, \frac{m}{2}]$ is apparent from Proposition 1. For $\alpha \in (\frac{2m}{m+1}, \frac{m}{2}]$, we have for all $t \in [m-1]$:

$$\begin{aligned}
&\mathbb{E}\Big[\frac{1}{n}\sum_{i=1}^{n}\frac{mX_i}{\alpha X_i + (m-\alpha)\sum_{j=1}^{m-1}X_{i,j}}\Big] \\
&\quad\leq \mathbb{E}\Big[\frac{1}{n}\frac{1}{m-1}\sum_{i=1}^{n}\sum_{j=1}^{m-1} tX_i\frac{t\alpha}{m}X_i + \sum_{j=t}^{t-1}X_{i,j}\Big] \\
&\quad= \mathbb{E}\Big[\frac{1}{n}\frac{1}{m-1}\sum_{i=1}^{n}\sum_{j=1}^{m-1} tX_i\frac{t\alpha}{m}X_i + \sum_{j=t}^{t-1}\frac{X_{i,j}}{\sum_{k=j}^{j+t-1}X_{i,k}}\Big] \\
&\quad\leq \frac{m}{t\alpha} \leq 1.
\end{aligned}$$

This proves the result. □

A.1.2 Proof for CPC

Proof. From Lemma 1, we have that:

$$\mathsf{DKL}(P\|Q) \geq \mathbb{E}_{y_{1:m-1}\sim Q^{m-1}}\left[\mathbb{E}_{x\sim P}\left[\frac{\log m\cdot r(x)}{r(x)+\sum_{i=1}^{m-1}r(y_i)}\right] - \mathbb{E}_{x\sim Q}\left[\frac{m\cdot r(x)}{r(x)+\sum_{i=1}^{m-1}r(y_i)}\right] + 1\right] \tag{A.1.1}$$

$$\geq \mathbb{E}_{y_{1:m-1}\sim Q^{m-1}}\left[\mathbb{E}_{x\sim P}\left[\frac{\log m\cdot r(x)}{r(x)+\sum_{i=1}^{m-1}r(y_i)}\right]\right] - 1 + 1 \tag{A.1.2}$$

$$= \mathbb{E}_{x,y_{1:m-1}\sim Q^{m-1}}\left[\frac{\log m\cdot r(x)}{r(x)+\sum_{i=1}^{m-1}r(y_i)}\right], \tag{A.1.3}$$

where the second inequality comes from Proposition 1 where $x \sim Q$ and $y_{1:m-1} \sim Q^{m-1}$ are exchangeable, thus proving the statement. □