Universidade Estadual de Campinas
Instituto de Computação

Cristina Freitas Bazzano

Modelagens para Segmentação de Imagens 3D da Densidade Eletrônica de Ligantes em Classes Químicas com Aprendizado Profundo

CAMPINAS
2022

Cristina Freitas Bazzano

Modelagens para Segmentação de Imagens 3D da Densidade Eletrônica de Ligantes em Classes Químicas com Aprendizado Profundo

Dissertação apresentada ao Instituto de Computação da Universidade Estadual de Campinas como parte dos requisitos para a obtenção do título de Mestra em Ciência da Computação.

Orientador: Prof. Dr. Guilherme Pimentel Telles
Coorientadora: Dra. Daniela Barretto Barbosa Trivella

Este exemplar corresponde à versão final da Dissertação defendida por Cristina Freitas Bazzano e orientada pelo Prof. Dr. Guilherme Pimentel Telles.

CAMPINAS
2022

Ficha catalográfica
Universidade Estadual de Campinas
Biblioteca do Instituto de Matemática, Estatística e Computação Científica
Ana Regina Machado - CRB 8/5467

Bazzano, Cristina Freitas, 1993-
B349m Modelagens para segmentação de imagens 3D da densidade eletrônica de ligantes em classes químicas com aprendizado profundo / Cristina Freitas Bazzano. – Campinas, SP : [s.n.], 2022.

Orientador: Guilherme Pimentel Telles.
Coorientador: Daniela Barretto Barbosa Trivella.
Dissertação (mestrado) – Universidade Estadual de Campinas, Instituto de Computação.

1. Cristalografia de proteínas. 2. Ligantes (Bioquímica) - Modelos matemáticos. 3. Densidade eletrônica. 4. Aprendizado profundo. 5. Segmentação semântica. 6. Produtos naturais. I. Telles, Guilherme Pimentel, 1972-. II. Trivella, Daniela Barretto Barbosa, 1980-. III. Universidade Estadual de Campinas. Instituto de Computação. IV. Título.

Informações Complementares

Título em outro idioma: Modeling for segmentation of 3D images of eletronic density of ligands in chemical classes with deep learning
Palavras-chave em inglês:
Protein cristallography
Ligand binding (Biochemistry) - Mathematical models
Electronic density
Deep learning
Semantic segmentation
Natural products
Área de concentração: Ciência da Computação
Titulação: Mestra em Ciência da Computação
Banca examinadora:
Guilherme Pimentel Telles [Orientador]
Hélio Pedrini
Paulo Sergio Lopes de Oliveira
Data de defesa: 31-08-2022
Programa de Pós-Graduação: Ciência da Computação

Identificação e informações acadêmicas do(a) aluno(a)
- ORCID do autor: https://orcid.org/0000-0002-1901-0909
- Currículo Lattes do autor: http://lattes.cnpq.br/8741957787289202

Universidade Estadual de Campinas
Instituto de Computação

Cristina Freitas Bazzano

Modelagens para Segmentação de Imagens 3D da Densidade Eletrônica de Ligantes em Classes Químicas com Aprendizado Profundo

Banca Examinadora:

- Prof. Dr. Guilherme Pimentel Telles
  IC/Unicamp
- Prof. Dr. Hélio Pedrini
  IC/Unicamp
- Prof. Dr. Paulo Sergio Lopes de Oliveira
  LNBio/CNPEM

A ata da defesa, assinada pelos membros da Comissão Examinadora, consta no SIGA/Sistema de Fluxo de Dissertação/Tese e na Secretaria do Programa da Unidade.

Campinas, 31 de agosto de 2022

Agradecimentos

Na minha vida pessoal, agradeço a minha família por todo suporte e acolhimento que me ofereceram e todo privilegio que me foi proporcionado. Tudo isso permitiu que muitas oportunidades surgissem na minha vida e que eu pudesse aproveitá-las no momento em que precisei e procurei. Obrigada por todo o amor e carinho. Amo vocês!

Agradeço à Flora, que me acompanha desde 2018 quando comecei meu estágio no CNPEM. Acho muito bonito ver como fomos mudando juntas e compartilhando os aprendizados do nosso caminho. Te amo muito!

Agradeço a todes da República 3 Pinheiros, pelo acolhimento, brisas boas e habilidades culinárias com degustações gostosas. As amizades da República Amnesia. Meu primo Caio. Os aprendizados na Casa da Neide, com seu jardim encantado cheio de gatos rajados e gatas pretas que tornaram a pandemia mais suportável. Que fiquem os bons momentos!

Agradecimentos especiais nessa reta final ao Cova, Pati e Toi, pela casinha acolhedora e gostosa que construímos, cheia de amor, risadinhas, cantorias, patucadas, plantinhas, mamonas e cachorrada doida.

No trabalho, deixo meu agradecimento especial a melhor dupla produtiva da pandemia, Luizinho você é top! Rafa obrigada pela sinceridade e conversas, aprendi muito com você. Dani obrigada por todas as oportunidades e confiança, fizemos um trabalho lindo. Também bebemos umas boas cervejas na boa companhia da Marjorie, foi divertido. Agradeço muito por todo conhecimento compartilhado, vocês ajudaram a construir as bases da biologia no meu aprendizado e toda dedicação foi especial. Obrigada Guilherme por aceitar se juntar a nós e enriquecer nossas discussões. Obrigada pela sua tranquilidade, confiança e suporte. Obrigada Bruna, Dani G, Débora, Gina, Henrique, Isa, Joane, Jonas, Leo, Marcos, Marjorie, Rafa, Pati, Paulo e Vini pelos bons momentos compartilhados e risadas na sala, no café e no almoço.

Agradeço ao grupo de computação do LNLS-CNPEM na época, Thiago Spina e Eduardo Miqueles, e ao grupo de bioinformática do LNBio-CNPEM, Helder Filho, João Guerra, José Pereira, Leandro Bortot e Paulo Oliveira, pela disponibilidade, discussões e sugestões fornecidas em apresentações sobre o andamento deste projeto no início de 2021. Em particular o uso da convolução dilatada foi sugerido por Thiago. E o uso de aumentação de dados com rotação pelo Leandro e Zé. Deixo um agradecimento especial ao Zé por fornecer todo o suporte para uso dos computadores da bioinformática, sempre muito atencioso e solícito.

Agradeço ao Instituto Serrapilheira por ter acreditado no projeto do NP^3 e ter fornecido todo apoio financeiro desde o meu estágio e agora no meu trabalho de mestrado.

E agradeço também a todos os encontros que cruzaram meu caminho nesse meio tempo e que de alguma forma deixaram lembranças e proporcionaram boas mudanças.

Resumo

A megadiversidade biológica brasileira, resultante de ecossistemas únicos manipulados e conservados por meio de um manejo agroflorestal milenar de diversos povos tradicionais do Brasil, têm um grande potencial para acelerar a descoberta de novos fármacos nacionais e representa importante fonte de inovação. Produtos naturais (PN) são a principal fonte de novos fármacos, historicamente inspirando mais de 60% dos fármacos hoje disponíveis. Os PN são moléculas orgânicas produzidas por qualquer ser vivo como decorrência de seu metabolismo secundário. Porém, a descoberta de novos fármacos a partir de PNs é muito desafiadora, os métodos convencionais para isolar o composto ativo consomem tempo, recursos e muitas vezes precisam de quilogramas do extrato do PN, o que pode ser inviável algumas vezes. Nesse contexto, surge a necessidade de novas técnicas e análises que aceleram e facilitam o processo de elucidação da estrutura molecular de moléculas bioativas desconhecidas. Atividades recentes do grupo de pesquisa deste trabalho e alguns poucos outros no mundo mostram que a cristalografia de proteínas em larga escala é uma alternativa para esse cenário. Esta técnica permite obter uma imagem 3D do contorno da molécula desconhecida em um menor tempo e a partir de poucos microgramas do extrato do PN. A incubação do cristal de uma proteína alvo com o extrato natural bioativo - que é representado por uma mistura de centenas de produtos naturais inicialmente desconhecidos - é uma técnica promissora na descoberta de novos fármacos e a interpretação da imagem 3D da densidade eletrônica é uma abordagem central neste processo. As soluções existentes resolvem o problema da interpretação da densidade eletrônica de moléculas já conhecidas e comumente encontradas nos bancos de dados de cristalografia de proteínas, o que não abrange o universo de moléculas de PNs, muitas das quais desconhecidas e com estrutura complexa pouco comum. Este trabalho fornece uma solução para o problema da reconstrução da estrutura molecular de ligantes (moléculas bioativas) desconhecidos a partir de uma modelagem com aprendizado profundo para interpretação automatizada da imagem 3D da densidade eletrônica dos dados de cristalografia de proteínas. Diversas modelagens baseadas em subestruturas químicas dos ligantes foram avaliadas para criação de bancos de dados de imagens em nuvens de pontos 3D da densidade eletrônica de ligantes rotuladas e os modelos de segmentação semântica obtidos apresentaram bom desempenho, com acurácias mIoU no teste entre 50,5% e 77,4%. As contribuições deste trabalho incluem: a primeira aplicação de aprendizado profundo 3D para imagens da densidade eletrônica de ligantes; um arcabouço de funções para implementação de outras modelagens na criação de imagens 3D da densidade eletrônica de ligantes; os bancos de dados de imagens rotuladas criados; os modelos treinados; e uma aplicação automatizada chamada *NP³ Blob Label* para detecção de ligantes na densidade eletrônica e interpretação das suas imagens 3D utilizando os modelos obtidos.

Abstract

The Brazilian biological megadiversity, resulting from unique ecosystems manipulated and conserved through an ancient agroforestry management of several traditional populations of Brazil, has great potential to accelerate the discovery of new national drugs and represents an important source of innovation. Natural products (NP) are the main source of new drugs, historically inspiring more than 60% of the drugs available today. NPs are organic molecules produced by any living being as a result of their secondary metabolism. However, the discovery of new drugs from NPs is very challenging, conventional methods to isolate the active compound consume time, resources and frequently need kilograms of the NP extract, which can sometimes be unfeasible. In this context, there is a need for new techniques and analyzes that accelerate and facilitate the process of elucidating the molecular structure of unknown bioactive molecules. Recent activities by the research group of this work and a few others in the world show that large-scale protein crystallography is an alternative to this scenario. This technique makes it possible to obtain a 3D image of the contour of the unknown molecule in a shorter time and from a few micrograms of the NP extract. The incubation of a target protein crystal with the bioactive natural extract - which is represented by a mixture of hundreds of initially unknown natural products - is a promising technique in the discovery of new drugs and the interpretation of the 3D electron density image is a central approach in this process. Existing solutions solve the problem of interpreting the electron density of molecules already known and commonly found in protein crystallography databases, which does not cover the universe of NP molecules, many of which are unknown and with an unusual complex structure. This work provides a solution to the problem of reconstructing the molecular structure of unknown ligands (bioactive molecules) from modeling with deep learning for automatic interpretation of the 3D electron density image of protein crystallography data. Several modeling based on chemical substructures of ligands were evaluated to create labeled imaging databases in 3D point clouds of the electron density of ligands. The semantic segmentation models obtained showed good performance, with mIoU accuracies in the test between 50.5% and 77.4%. Contributions of this work include: the first 3D deep learning application for electron density imaging of ligands; a framework of functions to implement other modeling to create 3D images of the electron density of ligands; the created labeled image databases; the trained models; and an automated application named *NP^3 Blob Label* for detecting ligands in electron density and interpretation of their 3D images using the obtained models.

Lista de Figuras

Lista de Tabelas

Sumário

Os benefícios que um novo medicamento da biodiversidade brasileira pode trazer para o Brasil pode gerar um impacto social muito bom se este for distribuído de forma mais acessível para o povo ou para o Sistema Único de Saúde (SUS) [72]. Tal descoberta só auxiliará na preservação, recuperação e manutenção da biodiversidade brasileira se respeitar e garantir os direitos dos povos tradicionais sobre seus conhecimentos vinculados a biodiversidade, com consentimento prévio, livre e informado e repartição justa e equitativa dos benefícios obtidos [24, 72, 93, 92, 30]. A Sociedade Brasileira para o Progresso da Ciência (SBPC) indica caminhos a serem seguidos para alcançar tal valorização do conhecimento tradicional dos povos guardiões da biodiversidade brasileira [24]. Sendo o Estado o responsável pela coordenação dos interesses no processo de gestão do conhecimento tradicional, por meio de políticas públicas e legislações que garantam os direitos dos povos tradicionais [92]. Como fonte de informação e planos de consulta prévia para essa gestão, diversos Protocolos Comunitários estão sendo construídos pelas comunidades locais para estabelecer e exteriorizar para o Estado regras mínimas e fundamentais sobre seus modos de vida, seus usos e costumes, como ferramentas de empoderamento político, autodeterminação e luta por território [30].

Cristina Freitas Bazzano

Apesar de instrumentos jurídicos nacionais e internacionais reconhecerem o direito dos povos tradicionais [30, 24], a falta de mecanismos nacionais de proteção e valorização de seus conhecimentos representa uma ameaça adicional [30, 24]. O Conselho de Gestão do Patrimônio Genético (Cgen), órgão que coordena o acesso ao patrimônio genético e ao conhecimento tradicional associado e da repartição de benefícios no país, vem sofrendo retrocessos desde 2015 [22]. De fato esse período foi um marco para a pesquisa cientifica pois possibilitou e facilitou o uso legalizado da biodiversidade em diversos projetos de pesquisa, um processo que era muito travado até então. Foi estabelecido na Lei 13.123/2015, denominada marco legal da biodiversidade, uma autodeclaração virtual por meio do Sistema Nacional de Gestão do Patrimônio Genético e do Conhecimento Tradicional Associado (SisGen) sobre uso da biodiversidade e conhecimento tradicional associado de forma simplificada e ágil, porém sem consentimento prévio, livre e informado dos povos afetados. Essa lei foi seguida do Decreto 8.772/2016 regulamentador que isenta o pagamento de direitos se este uso não atrelar explicitamente os modos tradicionais ao produto final [30, 24]. Além disso, essa lei estabelece dois tipos de conhecimento tradicional, um identificável (origem conhecida) e outro não identificável (origem difusa, múltipla), abandonando a ideia de território, como se o patrimônio genético e o conhecimento a ele associado não tivessem lugar de existência ou origem [30]. Para esses conhecimentos ditos não identificáveis, que correspondem a maioria dos casos, quase nenhum direito é garantido e pouco esforço vem sendo desprendido pelo Estado para "identificá-los". Ou seja, a legislação brasileira está desconsiderando todo o manejo milenar dos milhões de indígenas e povos tradicionais que habitaram e que habitam esse território hoje e sem resguardar o conhecimento tradicional vinculado ao ambiente [23, 17, 34].

Não é sobre ser contra a bioprospecção, que até 2015 era muito dificultada e inviabilizada, até mesmo para difundir remédios e curas para a sociedade como um todo. O problema está em como ocorre esse acesso e apropriação do conhecimento tradicional associado. Por esses motivos e o forte *lobby* das industrias farmacêuticas, de cosméticos e a de sementes e mudas (do agronegócio) que deu origem a essa lei, ela é popularmente conhecida como "Marco da Biopirataria" [30].

A visão de que o conhecimento tradicional é uma pista para o real conhecimento, o do método científico, pode levar a biopirataria e mercantilização da biodiversidade [24, 72, 93, 92, 74]. O grande desafio para novas ameaças tecnológicas é propagar a percepção de que o conhecimento tradicional é fruto de uma forma alternativa à hegemônica de viver e de conceber o mundo [24].

Mercantilizar não é proteger! Onde tem povo tradicional, tem floresta em pé!

Cristina Freitas Bazzano

Capítulo 1

Introdução

1.1 Uma Perspectiva Etnofarmacológica

O Brasil é o país mais biodiverso do mundo. São seis biomas diferentes [49], desde áreas alagadas, a savanas biodiversas, gramíneas naturais, florestas tropicais e à caatinga do sertão semiárido. Além de uma zona costeira e marinha muito diversa. Essa biodiversidade fornece ambientes extremos de umidade, seca e muito calor, mas também com temperaturas negativas no inverno. A megadiversidade biológica do Brasil abriga e é manejada por uma megadiversa população tradicional, que inclui povos indígenas, quilombolas, camponeses, ribeirinhos e muitas outras comunidades tradicionais [28, 34]. Essa diversidade social e cultural data de pelo menos 11 mil anos para os povos indígenas e 500 anos para os povos tradicionais, trazidos e escravizados no Brasil pós-colonial [28, 23]. Povos esses que com seus diversos modos de vida e conhecimentos tradicionais conviveram e convivem com essa biodiversidade para seu bem viver, modificando-a por meio de um manejo agroflorestal de seus territórios de direito, propagando plantas de interesse (domesticação) e atuando como guardiões da manutenção da sua existência [25, 26, 23, 34]. Os conhecimentos tradicionais recebem esse nome por serem transmitidos ao longo de várias gerações, atrelados a um território e fundados na observação minuciosa do ambiente. Esses conhecimentos "estão em constante experimentação, transformação e inovação [...]", tratam-se de "[...] uma diversidade de saberes e práticas locais que, inseparáveis de modos de vida e visões de mundo, possuem suas próprias formas de produção e circulação, concepções e valores" [24].

> "[...] a floresta não é um mero espaço selvagem que se opõe a um espaço doméstico, mas parte de um sistema de articulação multiespécie, que inclui as populações humanas e não-humanas. A não compreensão do significado de "sistema ou complexo agroflorestal" conduz à miopia recorrente de atribuir-se ao modo de produção indígena um caráter primitivo e irracional, pois supostamente incapaz de ampliar a produtividade por hectare de plantio. Os sistemas agroflorestais indígenas não visam extrair o máximo de uma unidade de terreno com o mínimo de trabalho, mas reproduzir um modo de articulação interespécies que permite a reprodução das condições de existência de humanos e não-humanos no próprio ato de transformar o meio ambiente. A

> isso, damos o nome de sustentabilidade. Não se trata de paralisia, mas de ações mais sofisticadas e atentas ao mundo em que habitamos." (Experiência Kuikuro [27]).

Os conhecimentos tradicionais associados à biodiversidade são ancestrais, extremamente complexos e em total sintonia e intimidade com o ambiente em que cada povo vive, com o seu território. Por meio do modo de produção agrícola desses povos, muitas vezes baseado em um manejo agroflorestal, estão disponíveis hoje todos os produtos da chamada agrobiodiversidade brasileira [30, 34]. Que correspondem as sementes crioulas, espécies domesticadas, semidomesticadas ou simplesmente manejadas por esses povos. E estabelece o direito desses povos sobre toda riqueza de plantas e animais da agricultura tradicional e da medicina tradicional [30, 24]. Sendo de extrema importância o reconhecimento, respeito e valorização do conhecimento tradicional associado à biodiversidade restante, para repartição justa e equitativa dos benefícios mediante o consentimento livre, prévio e informado dos povos afetados [30, 24].

O benefício medicinal do conhecimento tradicional associado à biodiversidade teve e têm implicações para a melhoria da qualidade de vida e descoberta de novos tratamentos. Ele pode ser percebido na diversidade de plantas medicinais e produtos como as garrafadas e chás tradicionais, nos fitoterápicos e nos conhecimentos medicinais socialmente difundidos [93, 72]. Além disso, historicamente os produtos naturais representam a maior fonte de novos fármacos [79, 80]: mais de 60% dos novos fármacos até 2019 vieram de produtos naturais e 89 dos medicamentos derivados de plantas usados na medicina ocidental foram descobertos através do estudo de medicinas tradicionais. A coleta de plantas com base no uso tradicional leva a uma melhor taxa de acerto de programas de investigação quando comparado a coletas ao acaso (ou com base em quimiossistemática), demonstrando o valor desses conhecimentos na área de pesquisa e desenvolvimento (P&D) de fármacos [24]. No programa inicial do Instituto Nacional do Câncer dos EUA de triagem de plantas para descoberta de novos agentes anticancerígenos, a porcentagem de *hits* inicial dos gêneros/espécies ativos citados em extratos de plantas medicinais foi próxima ao dobro das de triagem feitas ao acaso [24, 79].

Os produtos naturais continuam sendo a maior fonte de novos medicamentos no mercado farmacêutico e são considerados importante fonte de inovação [24, 72]. A pesquisa em descoberta de novos fármacos é a primeira etapa no desenvolvimento de um novo medicamento. É nessa etapa que se busca descobrir qual a molécula (*hit*) responsável por determinada ação medicinal. Essa descoberta engloba desvendar a estrutura química da molécula bioativa e como é seu mecanismo de ação no organismo ou alvo biológico sendo estudado. Um alvo biológico pode ser uma proteína específica, um patógeno ou uma célula inteira, de interesse para saúde na cura ou tratamento de doenças. Portanto, a descoberta de novos fármacos tem como objetivo encontrar novas moléculas bioativas contra alvos biológicos de interesse para cura e tratamento de doenças [74]. O potencial da biodiversidade brasileira para acelerar a descoberta de novos fármacos nacionais ainda é subaproveitado e subdimensionado [24, 92].

Sem o conhecimento tradicional, muitos desses produtos naturais provavelmente nem existiriam ou não estariam tão acessíveis. Isso demonstra a interdisciplinaridade existente

entre a pesquisa em descoberta de novos fármacos e a antropologia do conhecimento tradicional. Se esses dois conhecimentos caminhassem juntos, na chamada etnofarmacologia, muito proveito poderia ser obtido [72, 92, 34].

1.2 Uma Abordagem Tecnológica

Realizar a pesquisa de descoberta de novos fármacos a partir de produtos naturais, a maioria já conhecidos e utilizados na medicina tradicional, também não é uma tarefa fácil. Os produtos naturais são substâncias quimicamente complexas e muitas vezes difíceis de serem obtidos em grandes quantidades, pois um extrato de amostra da biota contêm de centenas a milhares de moléculas. Desvendar qual é a molécula, no meio dessas milhares, responsável pela ação biológica observada requer técnicas muito avançadas da ciência para ser feita em um tempo que a sociedade ocidental atual requisita. Muitas vezes se pode chegar ao final do processo e descobrir que a molécula de interesse já era na verdade conhecida [74]. Todas essas dificuldades encarecem a pesquisa de descoberta de novos fármacos a partir de produtos naturais seguindo os métodos científicos tradicionais. Nesse cenário se torna necessário pensar em métodos inovadores para viabilizar a pesquisa de descoberta de novos fármacos a partir de produtos naturais.

A inovação científica necessária para viabilizar a pesquisa em descoberta de novos fármacos a partir de produtos naturais, deve ser aplicada em larga escala, com poucas quantidades dos produtos e em um curto período de tempo. Essa inovação deveria permitir uma varredura da biodiversidade, testá-la contra uma diversidade de alvos biológicos e catalogar as informações coletadas. Para isso, torna-se necessário o uso de tecnologias de última geração, que tenham a sensibilidade e a riqueza de informação necessárias para possibilitar tal inovação. O Centro Nacional de Pesquisa em Energia e Materiais (CNPEM) é uma instituição que possui uma infraestrutura admirável, com um acelerador de partículas de 4a geração, que possibilita a aplicação de métodos inovadores para problemas complexos da atualidade. O projeto NP^3 surgiu no Laboratório Nacional de Biociências (LNBio) do CNPEM como uma iniciativa de inovação para descoberta de novos fármacos a partir de produtos naturais da biodiversidade brasileira. Ele se baseia na utilização das tecnologias disponíveis no CNPEM para análises de dados complexos da química, física e biologia estrutural.

A ideia do projeto NP^3 é combinar o resultado vindo da análise de três dados ortogonais de técnicas diferentes para propor moléculas candidatas aos *hits* observados em projeto de descobertas de novos fármacos a partir de produtos naturais. Esses dados são as análises biológicas para detecção de *hits* (ensaios biológicos de bioatividade), a quantificação química das amostras por meio da técnica de espectrometria de massas em tandem acoplada a cromatografia líquida (LC-MS/MS) e a modelagem 3D da molécula *hit* por meio da obtenção e interpretação de imagens 3D da técnica biofísica denominada cristalografia de proteínas.

Este projeto de mestrado contribui para o projeto NP^3 com uma aplicação automatizada baseada em aprendizado de máquina profundo para auxiliar na interpretação de imagens 3D de pequenas moléculas a partir de dados de cristalografia de proteínas.

A cristalografia de proteínas de raios X é a principal metodologia utilizada no âmbito da biologia estrutural. Essa técnica permite obter imagens 3D com detalhes atômicos de macromoléculas biológicas, como as proteínas. Isso torna possível visualizar e entender os sítios de ação da proteína, e sua interação com pequenas moléculas químicas (ligantes), DNA, outras proteínas, etc. Logo, auxilia na compreensão de diferentes mecanismos químico-biológicos de grande valor para a interpretação funcional dessas macromoléculas, de particular interesse na área da saúde.

A imagem 3D da cristalografia de proteínas é a densidade eletrônica. Ela representa a nuvem eletrônica dos elétrons dos átomos da proteína e de seus ligantes. A densidade eletrônica ilustrada na Figura 1.1 à esquerda, representa o espaço ocupado pelos átomos e se assemelha a um invólucro ao redor dos átomos que constitui a molécula ali presente. Na pesquisa de descoberta de novos fármacos, a obtenção dessa imagem 3D da molécula de interesse (ligante) permite visualizar o formato e a silhueta da estrutura química desse ligante antes de se saber quem ele é. Quando a estrutura é conhecida, também permite entender como acontece sua interação com a proteína alvo, sendo a base para o planejamento racional de fármacos baseado em estrutura. Quando o ligante é desconhecido, pode ser feita a interpretação dessa imagem 3D do ligante, que consiste na reconstrução da sua estrutura química de forma a explicar sua imagem e preencher a nuvem eletrônica observada (Figura 1.1 à direita).

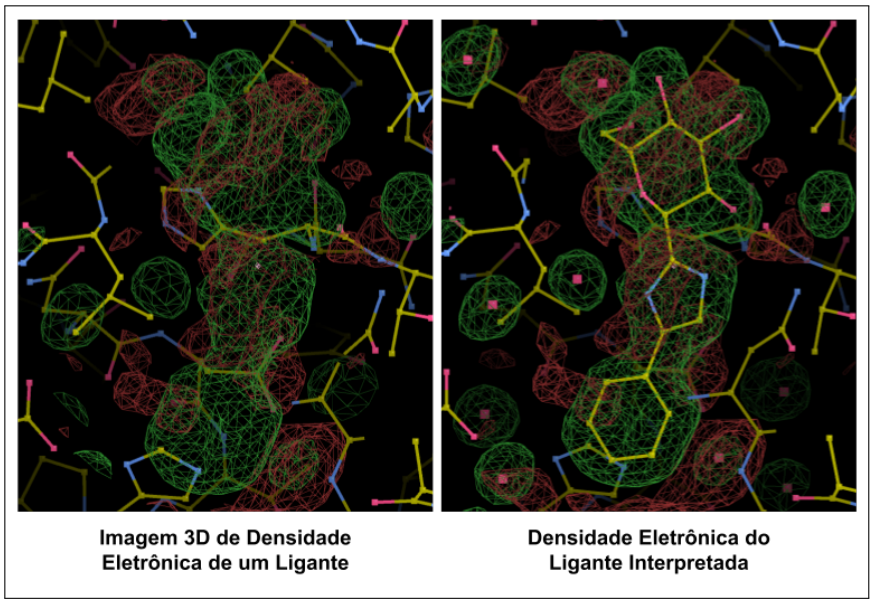

Figura 1.1: À esquerda, está representada uma região em verde do mapa da densidade eletrônica de um cristal de proteína onde contém um ligante, antes de ser interpretado. À direita, está o mesmo mapa da densidade eletrônica após ser interpretado com a estrutura do ligante 6MY (átomos de carbono em amarelo, de oxigênio em vermelho e de nitrogênio em azul). Também foram interpretadas as densidades eletrônicas de águas adjacentes ao ligante (pontos em vermelho referente ao átomo de oxigênio da molécula H_2O). As estruturas sem densidade eletrônica correspondem à estrutura da proteína. Imagem feita com a entrada 5JTT do Banco Mundial da Dados de Proteínas (PDB, *Protein Data Bank*).

A interpretação da imagem do ligante pode ser feita manualmente, onde cristalógrafos adicionam átomo por átomo em ferramenta de visualização como o Coot [38] até construir a molécula inteira, ou utilizando ferramentas já existentes para esse problema, como o Arp/wArp [16], o Phenix [99] e o CheckMyBlob [61]. Entretanto, as soluções existentes foram feitas para reconstruir a estrutura química de moléculas já conhecidas e comumente encontradas nos bancos de dados de cristalografia de proteínas. Quando a molécula de interesse possui uma estrutura inovadora ou pouco comum, a acurácia das soluções automatizadas fica comprometida, e logo, a interpretação da imagem obtida precisa ser feita manualmente. Isso é o que pode ocorrer quando se trabalha com amostras bioativas desconhecidas da biodiversidade brasileira, nas quais não se sabe quais os produtos naturais ali presentes e nem quem é a molécula de interesse. Mesmo que essas amostras já tenham moléculas conhecidas, não se sabe quais são elas e quais são de fato inovadoras.

Nesse cenário, torna-se necessário pensar em abordagens alternativas para facilitar a reconstrução da estrutura química de moléculas bioativas desconhecidas a partir de imagens 3D da densidade eletrônica. Este projeto de mestrado se insere nesse contexto como uma tentativa de auxiliar a interpretação das imagens 3D de ligantes desconhecidos. Uma das contribuições deste projeto foi a criação de modelos de aprendizado profundo ("deep learning") para segmentação semântica da imagem 3D da densidade eletrônica de ligantes. Com esses modelos é possível rotular a imagem dos ligantes com as subestruturas químicas ou os átomos presentes em cada região da sua densidade eletrônica. Essa rotulação pode servir como ponto de partida para a reconstrução manual completa da estrutura do ligante (ilustrado na Figura 1.2). E permite pensar em automatizações futuras baseadas nos resultados desses modelos para sugerir estruturas químicas completas para o ligante desconhecido.

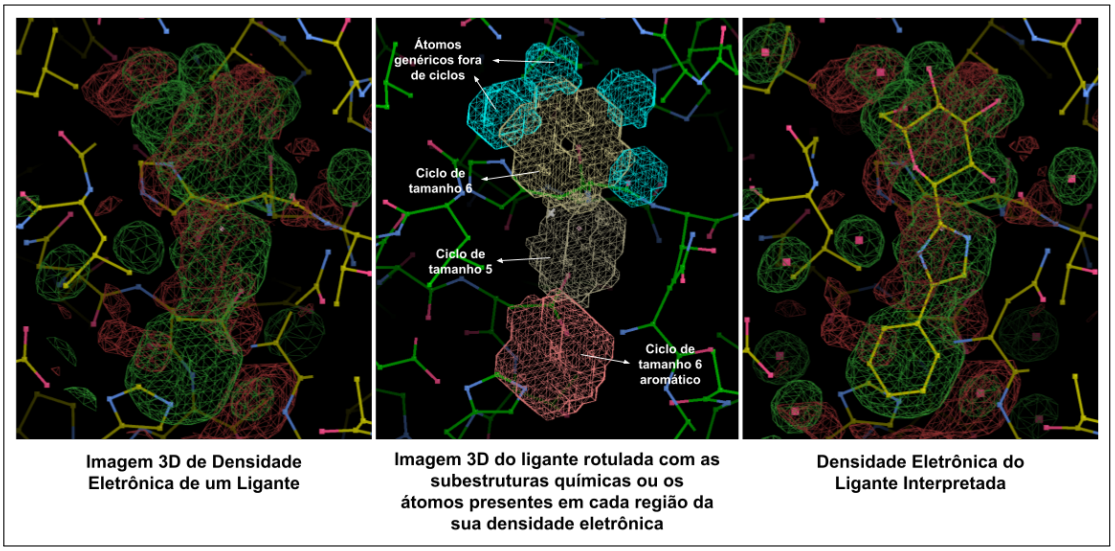

Figura 1.2: À esquerda está representada uma região em verde do mapa da densidade eletrônica de um cristal de proteína que contém um ligante antes de ser interpretado. Ao centro está o mesmo mapa da densidade eletrônica rotulado com um dos modelos de aprendizado profundo obtidos neste trabalho. À direita está o mesmo mapa da densidade eletrônica após ser interpretado com a estrutura do ligante 6MY (átomos de carbono em amarelo, de oxigênio em vermelho e de nitrogênio em azul). As rotulações mostradas ao centro facilitam a interpretação manual desse ligante e fornecem um ponto de partida para a sugestão de estruturas químicas para essa molécula. Também foram interpretadas as densidades eletrônicas de águas adjacentes ao ligante (pontos em vermelho referente ao átomo de oxigênio da molécula H_2O). As estruturas sem densidade eletrônica correspondem à estrutura da proteína. Imagem feita com a entrada 5JTT do Banco Mundial da Dados de Proteínas (PDB, *Protein Data Bank*).

Outra contribuição deste projeto de mestrado foi fornecer a primeira modelagem da imagem 3D da densidade eletrônica de ligantes utilizando aprendizado de máquina profundo. Aqui também inclui-se um arcabouço de funções para manipular essas imagens e modificar as modelagens propostas. Além de bancos de dados de imagens 3D da densidade eletrônica de ligantes rotuladas, que permitem que diferentes arquiteturas de aprendizado profundo e configurações sejam testadas. Finalmente, uma aplicação chamada *NP³ Blob Label* foi desenvolvida para buscar por ligantes na densidade eletrônica e rotular suas imagens.

1.3 Introdução à Técnica e ao Dado de Cristalografia de Proteínas

Para a aplicação da técnica de cristalografia de proteínas de raios X é necessário obter cristais da proteína, para posterior realização de experimentos de difração de raios X, e finalmente o cálculo e a obtenção da estrutura 3D da proteína, com os detalhes atômicos. Os cristais de proteínas são compostos por muitas moléculas de proteínas individuais e constituem uma rede de organização periódica em 3 dimensões espaciais (x, y, z), tipicamente com dimensões em torno de $10 - 200\mu m$. A periodicidade da rede permite localizar

os núcleos internos de organização e simetria cristalinos. A unidade assimétrica (ASU) representa a menor unidade biológica sem simetria do cristal. A ASU pode ser representada por um monômero da proteína, um dímero assimétrico, um heterodímero, ou outras unidades oligoméricas sem relação de simetria cristalina entre si. Esta ASU, por outro lado, pode ser relacionada a outras ASU que constituem o cristal, obedecendo operações de simetria rotacionais e translacionais em torno dos eixos cristalinos. Os eixos cristalinos tipicamente obedecem simetria rotacional de ordem 2, 3, 4 ou 6. Já os eixos de translação obedecem movimentações de frações de uma cela unitária, podendo ser $1/2, 1/3, 1/4, 1/6$ e divisões entre estas proporções. A cela unitária é a menor unidade repetitiva na rede periódica do cristal, que contém um grupo único de ASUs relacionadas pelas operações de simetria supracitadas. O cristal completo pode ser representado por operações de translação da cela unitária nas dimensões espaciais x, y e z do cristal. Sabendo-se os eixos de simetria operando na cela unitária (eixos cristalinos de rotação e translação) e suas dimensões $(a, b, c, \alpha, \beta, \gamma)$, é possível representar todo o cristal a partir da ASU (Figura 1.3).

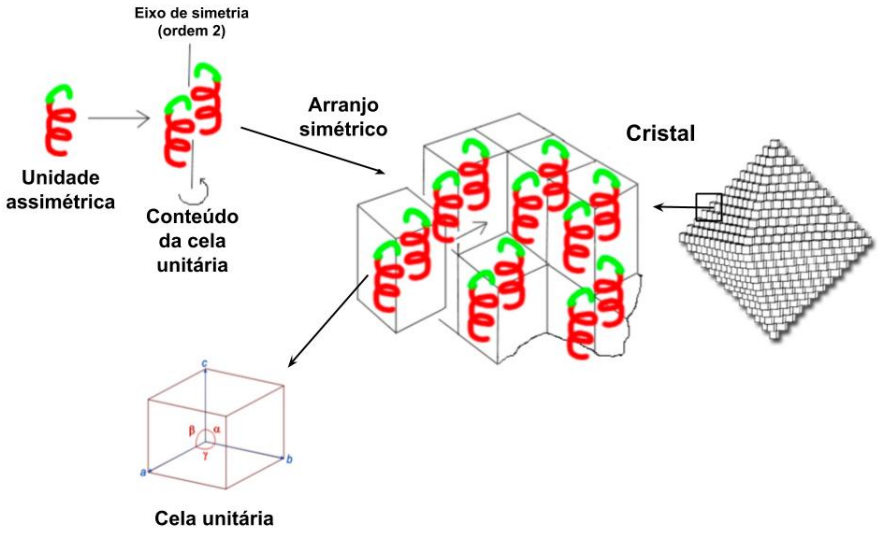

Figura 1.3: Representação esquemática da organização de um cristal de proteína. A cela unitária, a menor unidade repetitiva do cristal, é definida em uma caixa com eixos a,b,c e ângulos α, β, γ que se repete de forma periódica ao longo dos eixos x, y e z do cristal. A unidade assimétrica (ASU), representa a menor unidade sem simetria do cristal, a qual pode ser repetida na cela unitária utilizando os eixos cristalinos de roto-translação da cela unitária. Assim, conhecendo-se a ASU, os eixos de roto-translação da cela unitária e suas dimensões pode-se representar a cela unitária e o cristal completo. Imagens adaptadas e modificadas de https://www.xtal.iqfr.csic.es/Cristalografia/parte_07-en.html

Durante a determinação da estrutura de uma proteína, o primeiro passo é inferir as dimensões e estas operações de simetria da cela unitária e determinar a ASU. As dimensões da cela unitária são obtidas a partir das relações geométricas das reflexões do padrão de difração. Esta etapa, denominada indexação, deriva os parâmetros $a, b, c, \alpha, \beta, \gamma$ e a posição e índice das reflexões hkl. Uma vez acertados esses pontos, a determinação da estrutura da proteína é focada na determinação das posições atômicas da ASU apenas, simplificando o problema de determinação da posição de todos os átomos do cristal.

Vale comentar que além da proteína em si, o cristal é constituído de cerca de 50% de solvente. Assim, inicialmente a determinação da estrutura da proteína apresenta regiões vazias, que são os canais de solvente. A região estruturada e visível pode conter também outras moléculas que não são referentes à proteína em si. Estas podem ser pequenas moléculas ligantes, adicionadas propositalmente ao cristal ou oriundas das condições de cristalização. Ainda, alguns complexos da proteína com ácido desoxirribonucleico (DNA), ácido ribonucleico (RNA) e outras proteínas podem também ser obtidos e determinados com detalhes atômicos.

A imagem 3D gerada pela cristalografia de proteínas é a densidade eletrônica. Esta, por sua vez, é obtida a partir do processamento de dados experimentais de difração de raios X de cristais de proteínas. O mapa de densidade eletrônica é uma função contínua $\rho(x, y, z)$, que representa o espaço ocupado pela nuvem eletrônica dos elétrons dos átomos da proteína e seus ligantes. Esta nuvem eletrônica interage com os raios X, espalhando-os em todas as direções espaciais. Dada a organização periódica do cristal em uma rede, ele constitui uma rede de difração. Os raios X espalhados sofrem então interferência construtiva apenas em algumas direções específicas. Isto faz com que o espalhamento de raios X pelo cristal apresente um padrão de difração bem definido, com espalhamento discreto, nas chamadas reflexões hkl. O padrão de difração representa o espaço recíproco, enquanto o cristal, suas moléculas de proteína e seus átomos representam o espaço real.

A densidade eletrônica $\rho(x, y, z)$ é medida em elétrons por Angstroms ao cubo ($e\text{Å}^{-3}$). Esta função é calculada a partir dos fatores de estrutura $F(h, k, l)$ (Equação 1.1). Os $F(h, k, l)$ definem um número complexo constituído por amplitude $|F(h, k, l)|$ e fase $\phi(h, k, l)$. $|F(h, k, l)|$ representa a amplitude da onda espalhada em uma dada direção, estando diretamente relacionada à intensidade de uma dada reflexão hkl. O índice hkl, refere-se ao plano cristalino no espaço recíproco, denominado índice de Miller [75], e está relacionado à posição da reflexão hkl no padrão de difração: quanto mais longe do centro menor a distância entre planos de espalhamento. A fase da onda espalhada em cada direção é perdida no experimento de difração, porém pode ser calculada por métodos de faseamento experimentais como *multiple isomorphous replacement* (MIR), *multiple/single isomorphous replacement plus anomalous scattering* (MIRAS/SIRAS), *single isomorphous replacement* (SIR), *multi-wavelength anomalous dispersion* (MAD), *single-wavelength anomalous dispersion* (SAD) ou combinações destes [98], ou podem ser obtidas por substituição molecular [91].

$$\rho(x, y, z) = \frac{1}{V} \sum_{h,k,l} |F(h, k, l)| \cdot e^{-2\pi i[h \cdot x + k \cdot y + l \cdot z - \phi(h,k,l)]} \tag{1.1}$$

Os fatores de estrutura $F(hkl)$ são relacionados ao espaço real (densidade eletrônica x, y, z no cristal) através de uma transformada de Fourier nas 3 dimensões, relacionando assim o espaço recíproco com o espaço real [60]. Desta forma, o valor da densidade eletrônica é definido nos eixos x, y, z da cela unitária do cristal, a qual é representada por uma caixa periódica de volume V e representa o espaço real. Cada átomo no cristal contribui para cada reflexão hkl e assim para cada fator de estrutura.

O resultado é um mapa do cristal que mostra a distribuição dos elétrons em cada ponto do espaço real (x, y, z). Este é o mapa da densidade eletrônica observado (Fo), que

quando visualizado a um certo contorno (por exemplo $\rho(x, y, z) > 2\sigma$, onde σ é o desvio padrão da densidade do mapa) mostra um invólucro ao redor da estrutura química das moléculas cristalizadas, que podem ser a proteína, águas, íons, solventes, reagentes de cristalização e possíveis ligantes (Figura 1.4). A partir do mapa de densidade eletrônica calculado é possível inferir a posição dos átomos no cristal, formando as estruturas 3D das proteínas, localizando moléculas de água estruturais e identificando outras moléculas ligantes desta proteína.

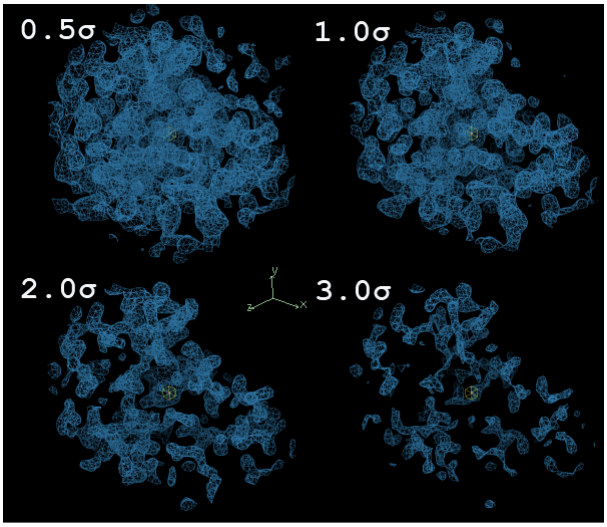

Figura 1.4: Mapa de densidade eletrônica para uma região do cristal da Ribonuclease pancreática bovina A (RNase A) em quatro níveis de contorno diferentes em unidades de desvio padrão acima da média: $0,5\sigma$, 1σ, 2σ e 3σ são mostrados. Fonte: https://www.quora.com/What-does-the-sigma-level-refer-to-in-electron-density-mapping/answer/Alex-Siegel

Com a obtenção da densidade eletrônica, inicia-se a etapa mais laboriosa da cristalografia de proteínas, que é a interpretação dos mapas da densidade eletrônica. Essa etapa consiste na criação de um modelo atômico, em três dimensões, contendo as posições dos átomos das moléculas cristalizadas que explicam (preenchem) os invólucros observados. A interpretação da densidade é tipicamente um processo iterativo, no qual um software de refinamento [110, 78, 97] constrói uma parte do modelo e depois o refina (Figura 1.5). O programa de refinamento fará pequenas alterações no modelo, ajustando parâmetros (como as coordenadas atômicas) e aplicando restrições geométricas, o que melhora a capacidade do modelo de explicar os dados experimentais e garante que ele será quimicamente razoável [59]. Os cristalógrafos especialistas devem verificar o resultado e, se necessário, refinar manualmente eventuais erros encontrados no modelo atômico (divergências entre o modelo e o mapa da densidade eletrônica). Com um modelo aprimorado, novos mapas podem ser calculados, revelando mais detalhes à densidade eletrônica, como por exemplo a presença de ligantes.

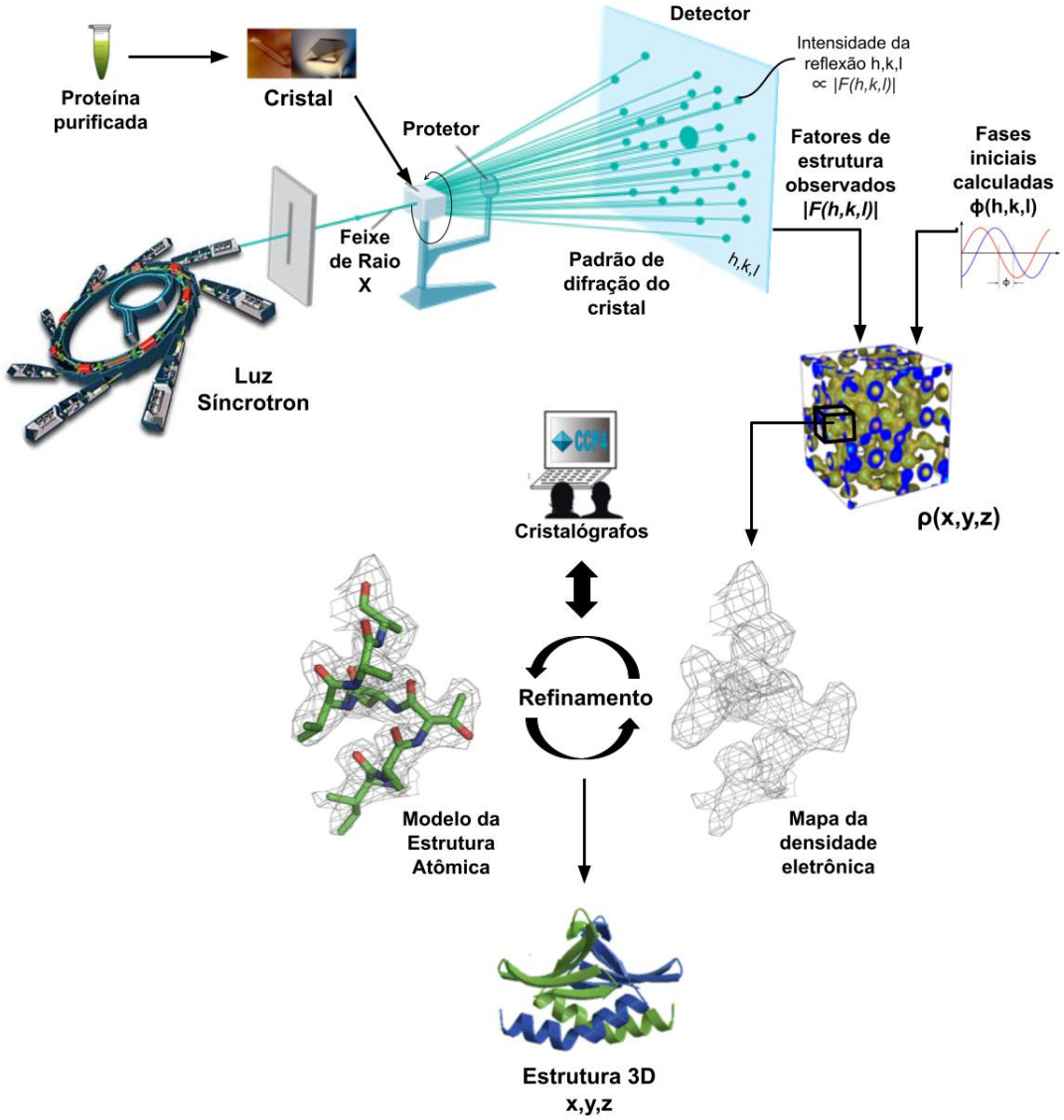

Figura 1.5: Experimento de difração de raios X e processamento dos dados de cristalografia de proteínas. Imagens adaptadas e modificadas de: https://www.xtal.iqfr.csic.es/Cristalografia/parte_07-en.html e https://www.researchgate.net/figure/Crystal-structure-of-C12A7-The-electron-density-isosurface-05-10-6-epm-3-in_fig4_261067822

Ligantes são pequenas moléculas que interagem e se ligam à proteína em regiões denominadas sítios de ligação, ou cavidades. Os ligantes podem modular a atividade de proteínas (muitas vezes ativar ou inibir algum dos seus mecanismos biológicos), e nestes casos, essas pequenas moléculas são denominados bioativas.

Com as soluções existentes para construção de modelos de proteínas [66, 47, 19], as regiões da estrutura macromolecular correspondentes às cadeias polipeptídicas (proteínas) ou polinucleotídicas (DNA e RNAs) podem ser construídas com alta precisão e velocidade. Por outro lado, essa etapa é especialmente desafiadora quando a estrutura da proteína contém ligantes não identificados anteriormente (desconhecidos) ou com uma estrutura complexa. Um ligante com uma estrutura complexa é formado pela composição de diversas subestruturas mais simples em arranjos não cobertos pelos ligantes mais comumente

encontrados. Tais ligantes são geralmente modelados manualmente e sua identificação correta requer bom julgamento e conhecimento de químicos e biólogos especialistas. Neste caso, a interpretação manual da densidade eletrônica, além de desafiadora, é uma atividade muito suscetível a um viés do cristalógrafo especialista, porém muito explanativa para se compreender a estrutura química da substância bioativa (ligante) presente no cristal, bem como as interações moleculares desta com a proteína alvo, em resolução atômica e 3D.

Diferentes mapas de densidade eletrônica podem ser calculados, dependendo do objetivo da análise, sendo os mais comuns os mapas $2Fo - Fc$ e $Fo - Fc$. Este último, denominado de mapa de densidade eletrônica residual, ou mapa de diferença, é obtido pela diferença da função de Fourier $\Delta\rho$ entre os fatores da estrutura (Fo) dos dados experimentais de difração de raios X e a densidade eletrônica calculada a partir do modelo atômico proposto para a estrutura da proteína (Fc). Os mapas $Fo - Fc$ positivos, portanto, mostram as densidades eletrônicas extras (denominadas "blobs") presentes em uma estrutura cristalográfica. Ou seja, revelam a presença de átomos presentes na estrutura cristalográfica real (representada por Fo), não adicionados ao modelo atômico da proteína (representado por Fc). Este é o caso, por exemplo, dos ligantes e está ilustrado na Figura 1.6.

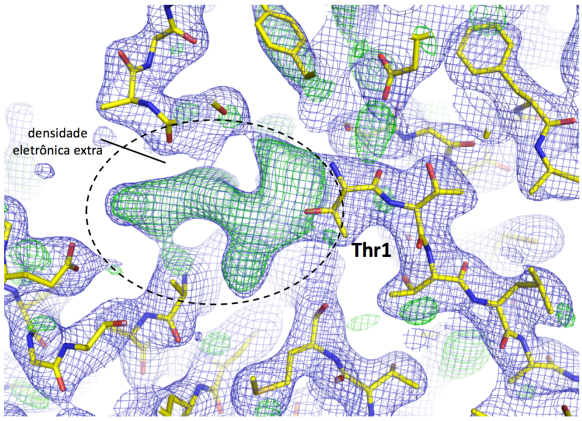

Figura 1.6: Densidade eletrônica residual em verde, também chamada de "blob", evidencia a presença de um ligante. Fonte: [42].

Na descoberta de fármacos, que busca encontrar novas moléculas que possam ser usadas como medicamento, a introdução controlada de ligantes em cristais é usada experimentalmente com frequência: onde substâncias químicas conhecidas e moduladoras (bioativas) da proteína de interesse biológico são incubadas com os cristais da proteína, em um procedimento denominado "soaking". Após esse procedimento obtém-se o cristal com os ligantes incubados, que serão visíveis em "blobs" no mapa de densidade eletrônica residual. Mais recentemente, foram introduzidas abordagens onde incuba-se o cristal da proteína alvo com misturas de moléculas químicas, conhecidas ou não, a fim de se obter complexos cristalográficos da proteína com seu melhor ligante [2]. O cristal captura dentro da mistura a substância química de maior afinidade à proteína e, consequentemente, revela seu invólucro no mapa de densidade eletrônica residual.

No caso da presença de ligantes "inesperados" ou ligantes desconhecidos proposital-

mente adicionados ao cristal deve-se interpretar o mapa de densidade eletrônica "extra" a fim de elucidar-se a estrutura das substâncias químicas que interagem com a proteína. Esse problema é conhecido como a reconstrução da estrutura molecular de ligantes desconhecidos e sua solução provê informações valiosíssimas para a compreensão da biologia química da proteína em estudo e gera bases para o desenvolvimento de sondas químicas ou novos fármacos.

A técnica de cristalografia de proteínas, especialmente os procedimentos de "soaking" e interpretação da densidade eletrônica, são abordagens centrais neste processo e já vêm mostrando casos de sucesso na literatura e no grupo de pesquisa do LNBio-CNPEM no âmbito do projeto NP^3. Neste caso, misturas de produtos naturais bioativas são submetidas ao processo de "soaking" com os cristais da proteína de interesse e o "blob" do produto natural bioativo deve ser interpretado no mapa de densidade eletrônica residual para que sua estrutura química seja reconstruída.

Uma solução automatizada com alta acurácia para o problema de reconstrução da estrutura molecular de ligantes desconhecidos pode acelerar o processo de descoberta de um novo medicamento e permitir que técnicas de *high throughput* sejam viáveis de serem implementadas na interpretação de densidades residuais de ligantes desconhecidos. Esta abordagem é inclusive um dos braços principais da plataforma NP^3, apoiada pelo Instituto Serrapilheira, na qual o presente projeto está relacionado. Esta estratégia se encaixa na abordagem de *high throughput crystallography*, em implementação na linha Manacá do Sirius (LNLS-CNPEM), e também nas estratégias para a descoberta de fármacos a partir da biodiversidade brasileira, envolvida na plataforma NP^3.

Entretanto, o universo dos ligantes é bastante complexo, suas estruturas químicas podem ser formadas por qualquer combinação de átomos e ligações químicas, o que dificulta uma modelagem automatizada. Além disso, a densidade eletrônica é uma imagem 3D estática que representa um sistema dinâmico com muitas conformações e, por isso, a distribuição de elétrons é mais ampla do que a distribuição gaussiana teórica (Fc) aproximada [46, 96], centrada no núcleo de cada átomo (ilustrada no canto superior esquerdo da Figura 1.7).

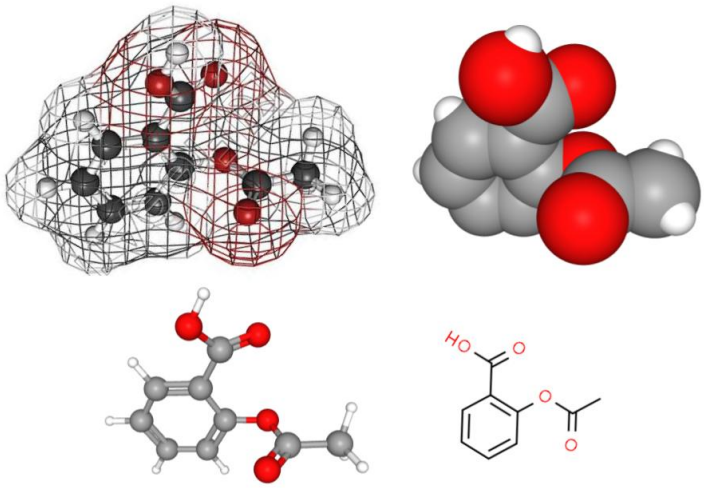

Figura 1.7: Ilustrações da molécula do ácido acetilsalicílico (fármaco popularmente conhecido como aspirina), cuja formula molecular é $C_9H_8O_4$ e SMILES é *CC(=O)OC1=CC=CC=C1C(=O)O*. As esferas em cinza escuro representam átomos de carbono, em vermelho átomos de oxigênio e em branco átomos de hidrogênio. A imagem no canto superior esquerdo ilustra a distribuição teórica da densidade eletrônica para essa molécula. No canto superior direito está a representação da molécula utilizando preenchimento de espaço com esferas proporcionais ao raio teórico de cada átomo. A imagem no canto inferior esquerdo é o modelo de bola e bastão. No canto inferior direito é a representação 2D da aspirina.

A resolução dos dados é o fator de qualidade da coleta de dados, é a medida do nível de detalhamento do padrão de difração e é limitado pela capacidade do instrumento e pela qualidade do cristal (sendo comumente designado em cristalografia < 1.5 Å alta resolução e > 2.5 Å baixa resolução) (Figura 1.8). O ambiente em torno das moléculas no cristal é modelado por interações eletrostáticas e de Van der Waals. Essas interações geram forças de atração e repulsão, que causam movimento nas moléculas [82], e adicionam um ruído intrínseco ao padrão de difração detectado. Esse ruído também pode ser devido à vibração dos átomos, a diferenças entre as muitas ASU da estrutura cristalina ou a resolução dos dados [10].

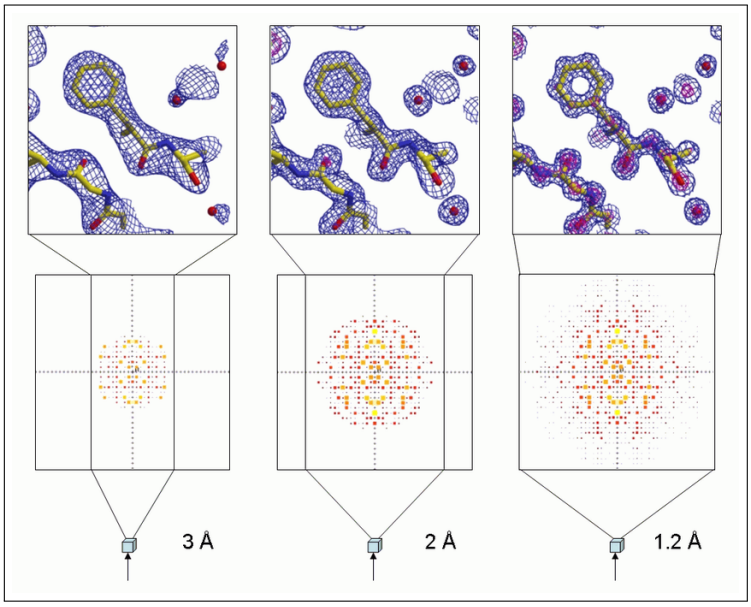

Figura 1.8: Difração e densidade eletrônica em resolução crescente. Os cristais (embaixo) difratam os raios X cada vez melhor (limite de difração ou resolução de 3.0, 2.0 e 1.2 Å da esquerda para a direita). Difração crescente (correspondente a amostragem mais fina). Fonte: https://www.researchgate.net/figure/Data-quality-determines-structural-detail-and-accuracy-The-qualitative-relation-between_fig3_277706893

A densidade eletrônica observada incluirá a média de todos esses pequenos movimentos, produzindo uma imagem levemente borrada das moléculas. Os movimentos das moléculas no cristal são incorporados ao modelo atômico por um fator B, ou fator de temperatura, e são proporcionais à magnitude desse valor. Além disso, na construção do modelo atômico, é também estimada a ocupância, parâmetro que indica a fração de moléculas que ocupam cada sítio, ou que possuem cada uma das conformações. A soma do valor de ocupâncias fracionárias para cada átomo deve ser 1.0. O fator B e a ocupância de cada átomo das diferentes conformações de cada molécula podem ser otimizados na etapa de refinamento do modelo atômico.

Em alguns casos, o método experimental pode não detectar alguns átomos presentes no cristal e suas coordenadas estarão ausentes no modelo atômico. Por exemplo, átomos que se encontram em regiões muito flexíveis e átomos de hidrogênio, que usualmente não são observados nos experimentos de raios X de proteínas, e logo, não são modelados. Em outros casos, apenas uma parte da molécula pode estar visível e será incluída no modelo [10].

Algumas soluções dos grupos do Arp/wArp [16], Phenix [99] e *Check My Blob* [61] já se propuseram a resolver o problema de reconstrução de ligantes conhecidos. A maioria delas extrai informações topológicas da densidade eletrônica, geram descritores do formato da densidade a um certo contorno e utilizam algoritmos de aprendizado de máquina, busca e *matching* para fazer as predições. Algumas delas também usam como restrição a geometria do ligante, os ângulos de torção das suas ligações químicas, a cavidade da proteína em que se encontra o ligante e a energia da estrutura construída. Porém, todas essas soluções apresentam uma acurácia baixa quando o ligante é desconhecido e se trata de uma molécula inovadora e de estrutura química complexa, diferente daquelas dos ligantes

mais comumente encontrados. Este é o caso dos produtos naturais, moléculas complexas e muitas vezes com estruturas desconhecidas. Além disso, em resoluções muito baixas as soluções existentes produzem resultados ainda menos confiáveis. Na Seção 1.6 sobre Trabalhos Relacionados as soluções existentes serão descritas com maior detalhamento.

Os métodos de aprendizado de máquina profundo ("deep learning") têm mostrado resultados surpreendentes na descoberta de estruturas complexas em dados de alta dimensão e, portanto, são aplicáveis a muitos domínios da ciência em problemas que resistiram às melhores tentativas da comunidade de inteligência artificial por muitos anos [65]. Uma arquitetura de aprendizado profundo convolucional é uma pilha multicamada de módulos simples, todos (ou a maioria) sujeitos a aprendizado sendo que muitos deles calculando mapeamentos não-lineares entre entrada e saída. Esse aprendizado é supervisionado, para toda entrada existe uma saída esperada, ou seja, os dados precisam estar rotulados e o entendimento do modelo preditivo será construído a partir da representação e modelagem dos dados utilizada, que poderá permitir que correlações no mapeamento sejam aprendidas pelo modelo. Durante o processo de treinamento, os pesos de cada um dos filtros convolucionais são otimizados para detectar padrões espaciais locais e isso pode auxiliar a capturar melhor as informações bioquímicas da modelagem presentes na estrutura do ligante.

Este projeto de mestrado se propôs a resolver o problema de reconstrução da estrutura molecular de ligantes desconhecidos, em 3D, a partir da densidade eletrônica extra observada no mapa de densidade eletrônica residual de um experimento de cristalografia de proteínas, onde ligantes desconhecidos são propositalmente adicionados. A solução construída neste trabalho foi baseada em uma modelagem de aprendizado profundo para imagens 3D da densidade eletrônica residual de ligantes com a premissa de identificar padrões na densidade eletrônica induzidos pela presença de certas subestruturas químicas (denominadas fragmentos) no ligante. Um modelo de aprendizado profundo treinado para identificar fragmentos na imagem de ligantes na densidade residual pode reduzir o espaço químico de possibilidades a partir dos fragmentos preditos e facilitar a interpretação dos "blobs" de ligantes. A reconstrução da estrutura molecular de ligantes desconhecidos pode se guiar pelos resultados fornecidos por esta abordagem.

As contribuições deste projeto de mestrado começam por fornecer um banco de dados de *grids* de *voxels* 3D representando a densidade eletrônica residual de ligantes devidamente rotulados e prontos para serem utilizados em *frameworks* de aprendizado profundo para imagens 3D. Esse banco de dados inclui um arcabouço de funções que permitem criar as imagens 3D dos ligantes na densidade residual e implementar propostas de modelagem para rotulação dos ligantes e das suas imagens correspondentes. Seguindo a definição do banco de dados de imagens 3D da densidade eletrônica residual de ligantes, modelos preditivos foram treinados para classificar os *grids* das densidades em possíveis subestruturas químicas, seguindo rotulações propostas a partir da estrutura química dos ligantes. Finalmente, uma aplicação completa chamada *NP^3 Blob Label* foi desenvolvida para empregar essa metodologia em novos dados coletados.

1.4 Conjunto de Dados

O Banco de Dados de Proteínas (PDB) [10], gerido pela organização internacional do Banco Mundial de Dados de Proteínas (wwPBD, www.wwpdb.org) [11], é o repositório central de estruturas de macromoléculas biológicas, contendo estruturas completas de proteínas, DNA e RNA, além das pequenas moléculas ligantes de proteínas (Figura 1.9). Devido a uma grande variação na qualidade dos dados depositados, desde 2008 o wwPDB tornou obrigatório o depósito da densidade eletrônica (Fo) juntamente com o depósito da estrutura da proteína resolvida (modelo atômico), com o intuito de dar suporte à avaliação independente das estruturas depositadas em relação aos dados experimentais utilizados para derivá-las [36].

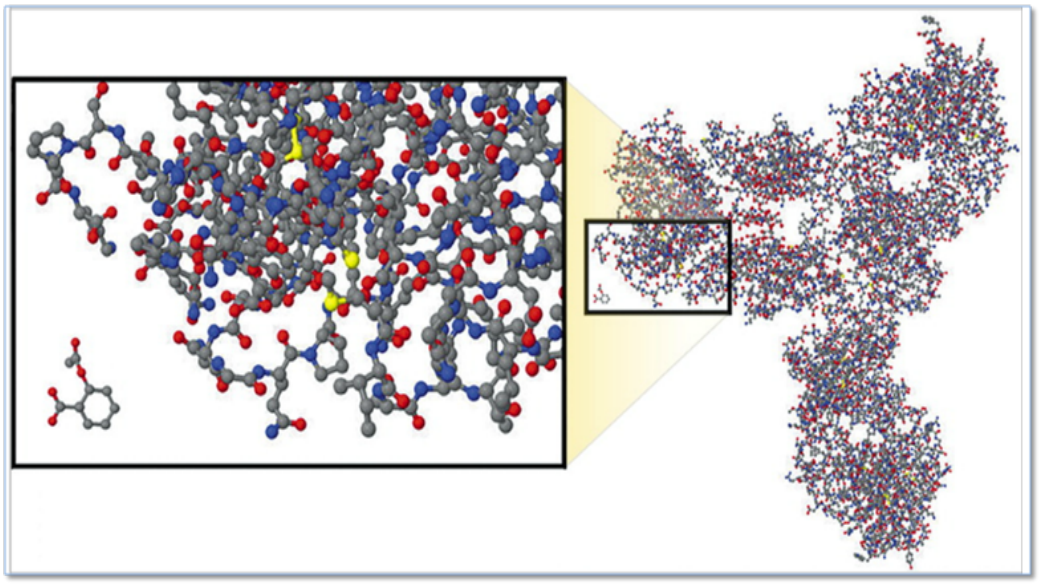

Figura 1.9: Comparação em proporção de um fármaco vs uma proteína. A imagem à direita ilustra a complexidade comparativa do fármaco da pequena molécula (aspirina) mais popular, que é a pequena estrutura molecular no canto inferior esquerdo da imagem em destaque e um anticorpo monoclonal (molécula grande) à direita. Fonte: https://www.azbio.org/getting-ready-for-biosimilars.

Apesar dos esforços do wwPDB em validar os dados de cristalografia sendo depositados, muitas estruturas de ligantes apresentam erros que inviabilizam seu uso direto por técnicas de mineração de dados sem que um filtro de qualidade seja devidamente aplicado [31]. Algumas investigações demonstraram que erros comuns nas estruturas do PDB podem ser atribuídos à negligência e falta de verificação rigorosa dos modelos em relação à densidade eletrônica, criação de modelos com ligantes sem evidências cristalográficas suficientes de que esses ligantes estão presentes, geração de números improváveis na validação, aplicação de simetria incorreta, apresentação ilógica dos resultados ou violação das regras da química e física [88]. Muitas vezes, esses erros são derivados da baixa resolução dos dados.

As restrições estereoquímicas na construção de modelos moleculares são normalmente usadas para descrever valores ideais e desvios-padrão estimados para comprimentos de ligação, ângulos de ligação, ângulos de torção, grupos planares e volumes quirais (objetos

não sobreponíveis à sua imagem especular - sem plano de simetria) [40]. Além disso, conflitos entre a estrutura da proteína e do ligante também devem ser avaliados na construção de modelos moleculares [59], i.e. inviabilidade da existência de átomos muito próximos. A geração de restrições de alta qualidade para moléculas é um obstáculo substancial para uma boa modelagem e deve ser tratada com o devido cuidado.

Existem várias medidas estatísticas para quantificar e exibir a correspondência entre um modelo atômico e a densidade eletrônica, sendo os principais o valor R do espaço real (RSR), o coeficiente de correlação com o espaço real (RSCC), as medidas de diferença da densidade (RSZD), a raiz da média do desvio ao quadrado (r.m.s.d.) das coordenadas do ligante no modelo inicial e após o refinamento e a força da densidade sob o modelo normalizado para ocupação (RSZO / OCC) [88, 85, 101]. Outros métodos e combinações de pontuações já foram propostos, como a razão entre o fator B do ligante e o dos átomos da cadeia lateral da proteína a seu redor [101], e são usados para avaliação da modelagem de ligantes [37].

O PDB fornece as entradas e as saídas para o problema de reconstrução de ligantes. Há uma grande quantidade de dados de treinamento disponíveis que foram o resultado de décadas de esforço da comunidade de cristalografia em determinar estruturas cristalográficas de macromoléculas, com depósitos crescentes desde o boom da genômica estrutural no início dos anos 2000 e atualmente com aproximadamente 10 mil novos depósitos de estrutura por ano [105] (Figura 1.10).

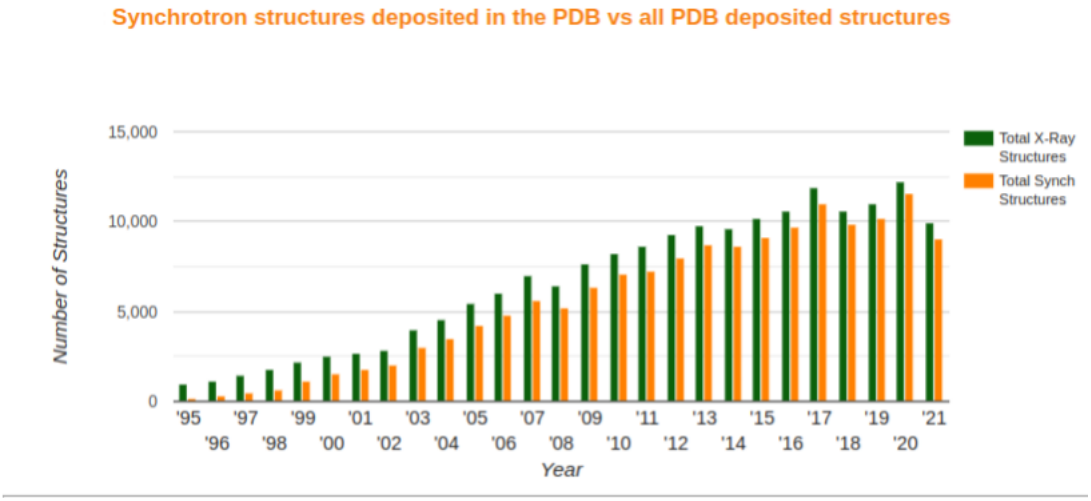

Figura 1.10: Estruturas de Raio X depositadas no PDB de 1995 até 2021. Essa distribuição evidencia a grande e crescente quantidade de dados presentes no PDB. A estimativa para 2022 passa de 10 mil depósitos. O número cumulativo de estruturas de proteína depositadas no PDB desde a sua criação já ultrapassa 150 mil. Fonte: https://biosync.rcsb.org/stats.do?stats_sec=MAIN&stats_focus_lvl=GLBL.

Porém, não existe uma rotulação definida para a imagem da densidade eletrônica de ligantes, o que se tem como saída é o modelo atômico em 3D proposto para os ligantes presentes na estrutura depositada, sendo este modelo atômico resultado da interpretação do "blob" do ligante. A tarefa de criar uma representação e modelagem para a densidade eletrônica a partir da estrutura química do ligante presente no modelo atômico também

foi um problema tratado neste projeto de mestrado. Diferentes rotulações foram testadas e modeladas para o problema, a fim de aprimorar o aprendizado do modelo preditivo sendo treinado.

1.5 Formulação do Problema

O problema da reconstrução da estrutura molecular de ligantes pode ser definido baseado na teoria de grafos para química [21], como um problema de geração de um grafo não direcionado denominado grafo molecular $G(V, E)$. Cada um dos átomos do ligante corresponde a um vértice $v \in V$ e cada ligação química presente na estrutura do ligante corresponde a uma aresta $e \in E$. A cada átomo v está associada uma posição x, y, z e a cada aresta e um tipo de ligação (simples, dupla, tripla ou aromática), o que define a conformação do ligante.

Cada tipo de átomo é limitado a fazer uma quantidade fixa de ligações químicas respeitando suas respectivas valências, i.e. cada tipo de átomo possui uma restrição quanto ao seu grau máximo. E todos os átomos estão sujeitos às restrições estereoquímicas e de energia da sua estrutura química e às restrições de conflito com a proteína (regiões inviáveis de existir um átomo devido à presença de um átomo da proteína).

Nessa definição, o objetivo do problema é construir o grafo molecular $G(V, E)$, representativo da estrutura química do ligante, que maximiza o seu encaixe na densidade eletrônica residual observada e é penalizado pela energia e geometria da molécula.

1.6 Trabalhos Relacionados

1.6.1 Soluções para Reconstrução de Ligantes Conhecidos

As soluções existentes para o problema de reconstrução molecular de ligantes conhecidos são baseadas no uso de ângulos de torção, matrizes de distância interatômicas ou na análise topológica da densidade eletrônica.

Dentre as soluções mais recentes para esse problema se destacam as soluções do Laboratório de Biologia Molecular Europeu, Arp/wArp [16]. A primeira solução desse grupo se baseou na geração de átomos testes (pseudo-atômicos) centrados em picos de máximo local da densidade eletrônica (pontos em que ρ é máximo em relação à sua vizinhança). A partir de características geométricas da molécula do ligante sendo construído, um modelo de erro é criado para os parâmetros posicionais dos átomos dos ligantes. Um algoritmo de busca foi desenvolvido para otimizar a função de pontuação criada baseada no modelo de erro e resulta em átomos do ligante sendo atribuídos a pontos do *grid* do mapa da densidade eletrônica [117]. Essa solução foi aprimorada para encontrar e classificar ligantes candidatos a partir da correspondência entre *grids* pseudo-atômicos esparsos gerados e descritores matemáticos da densidade eletrônica em relação a uma coleção de ligantes selecionados e seus variantes conformacionais [16]. A acurácia dessa solução foi testada para 82 ligantes distintos selecionados em até 200 conformações diferentes e resultou, após classificação baseada na correlação com a densidade, na molécula correta sendo colocada

no topo das predições em 32% dos casos.

A abordagem *LigEnergy* que estima as energias da interação proteína-ligante foi subsequentemente usada como um parâmetro adicional na classificação de ligantes usando a representação pseudo-atômica esparsa, e dentre os 100 casos considerados, 50 tiveram o ligante correto identificado no topo da lista de candidatos preditos [13].

A solução para este problema pelo sistema Phenix [99], mantido por diversos laboratórios de biologia estrutural computacional, identifica ligantes com base em duas características da densidade eletrônica: a correlação da densidade do ligante com cada um dos ligantes de um conjunto de teste após otimização do seu encaixe e a correlação de descritores de forma da densidade eletrônica residual com os descritores de forma de densidade dos modelos de cada ligante do conjunto de teste. Para um conjunto teste com os 200 ligantes mais frequentes do PDB, esse procedimento colocou o ligante correto no topo da lista de candidatos em 48% dos casos.

Mais recentemente, foi desenvolvida uma solução chamada *Check My Blob* [61], baseada em algoritmos de aprendizado de máquina, para identificar os 200 ligantes mais frequentes em mapas de densidade eletrônica a partir de descritores de forma, volume, ambiente químico e resolução do dado, capazes de generalizar partes dos ligantes. A acurácia desse método em um conjunto de dados próprio e nos conjuntos de dados das soluções do Arp/wArp e do Phenix variou de 56,3% a 72,5%.

Todas as soluções existentes para o problema de reconstrução da estrutura molecular de ligantes não apresentam um resultado muito bom quando a estrutura do ligante diverge daquela dos ligantes mais frequentes ou do conjunto teste de ligantes utilizado. Quando a estrutura do ligante, além de inovadora, também é complexa, a acurácia dessas soluções cai substancialmente e suas predições falham em gerar um modelo para o ligante que realmente se assemelhe à sua estrutura real. Além disso, não foi encontrado um trabalho que tenha comparado o resultado de todas essas soluções usando um mesmo conjunto de dados e as mesmas métricas de avaliação de desempenho, com exceção das acurácias reportadas pelo *Check My Blob* que apenas comparam os resultados da sua solução com as demais (Tabela 1.1). Também não foi encontrado uma comparação do desempenho de todas essas soluções para ligantes pouco frequentes e este resultado seria de grande proveito para grupos de pesquisa que trabalham com moléculas desconhecidas.

Tabela 1.1: Resultados de validação cruzada apresentados pelo *Check My Blob* (CMB). O conjunto de dados CL, utilizado na solução do Arp/wArp [16], possui contagens de ligantes únicos variando de 42622 a 16. Enquanto o conjunto de dados TAMC, utilizado na solução do Phenix [99], possui contagens de ligantes únicos variando de 36535 a 114.

Conjunto de Dados	**Algoritmo**	**Exemplos de teste**	**Acurácia (%)**
CL	Arp/wArp	121360	32
	CMB		72,5
TAMC	Phenix	161758	48,5
	CMB		56,3

1.6.2 Soluções para Encaixe de Ligantes em Mapas da Densidade Eletrônica

Sabendo que os ligantes complexos são formados por diversas subestruturas químicas mais simples (fragmentos), em arranjos variados, soluções de encaixe de ligantes para técnicas de *fragment screening* também foram estudas. Essas técnicas fazem o encaixe de um conjunto de fragmentos na densidade otimizando seus modelos atômicos, portanto podem auxiliar na fixação de uma parte substancial do ligante e facilitar a sua extensão até a estrutura completa do ligante. Quando a estrutura química do ligante é conhecida ou o ligante pertence a um conjunto conhecido de fragmentos (com mais de 4 átomos), o problema se torna encontrar as regiões na densidade eletrônica residual ("blobs") onde o ligante melhor se encaixa e definir sua melhor conformação segundo algum critério de otimização e classificação do encaixe da sua estrutura química na densidade observada.

Dentre as soluções existentes para o problema de encaixe de ligantes conhecidos, o grupo do Arp/wArp combina métodos baseados em grafos com "label swapping" e usa uma busca Metropolis para encontrar a melhor conformação para encaixar ligantes em cavidades da proteína [64]. A acurácia desse método variou de 53% a 75%. O grupo do Phenix utiliza um método que começa por encaixar um fragmento principal do ligante na densidade eletrônica e depois estende o restante da sua estrutura seguindo a densidade até um ajuste completo da molécula [100]. A acurácia do encaixe desse método variou de 58% a 73%.

O método de busca implementado pelo X-LIGAND [81] ajusta ligantes flexíveis em regiões do mapa em que eles podem ser inseridos variando o ângulo de torção de ligações químicas rotacionáveis e otimiza o ajuste usando um gradiente de refinamento. Para os sete casos de estudo, cinco foram encaixados corretamente.

A ferramenta Coot [39], muito utilizada para a visualização e refinamento dos modelos atômicos dos dados de cristalografia de proteínas, também oferece uma solução para encaixe de ligantes. O método implementado nessa ferramente começa com a geração de diversas conformações para o ligante a ser encaixado. Após a geração de conformações, o mapa da densidade residual é vasculhado em busca de aglomerações de pontos no *grid* (*clusters*) que possam conter um ligante. Esses *clusters* são comparados com a forma das conformações por meio da análise de componentes principais (PCA) e as melhores conformações são aceitas se elas passarem em certos filtros (definidos pelo usuário) para encaixe na densidade. Não foi encontrado um resultado de desempenho para essa ferramenta.

Diferente dessas soluções e visando o encaixe de ligantes em dados com baixa resolução, foi desenvolvido pelo grupo do Phenix um algoritmo que aproxima o eixo medial para simplificar o contorno da densidade eletrônica e encaixar ligantes na densidade. Esse procedimento captura a informação do eixo medial em um grafo que é então comparado contra um grafo do modelo molecular do ligante a ser encaixado [5]. Essa solução encaixou 22 dos 27 ligantes testados, com r.m.s.d. menor do que 2 Å de uma dada correspondência entre os átomos dos ligantes e o eixo medial.

Outra questão a ser considerada é a ocupância do ligante na estrutura cristalográfica. Por exemplo, uma das maiores dificuldade dos métodos de *fragment screening* é a baixa qualidade da densidade residual dos ligantes, o que é derivado de baixa ocupância. Isso

ocorre com frequência no procedimento de "soaking" com amostras desconhecidas. Os fragmentos usualmente não estão ligados a todas as proteínas do cristal (ocupância $\ll 1$) e como a determinação estrutural é resumida à ASU (é uma média de todas as ASUs no cristal), isso gera uma densidade fraca e ambígua do ligante, o que torna os modelos finais menos confiáveis e dependentes de decisões subjetivas do cientista responsável [88]. O *workflow* desenvolvido pelo grupo do XChem [62] é baseado no algoritmo PanDDA [84] para melhorar a identificação de ligantes em densidades residuais incompletas e subsequente uso da ferramenta Coot para reconstrução do ligante. O algoritmo PanDDA melhora a sensibilidade da análise e revela uma densidade residual mais clara ao subtrair da densidade seu "estado base" (densidade sem a presença do fragmento). Esse método permite a identificação da presença de ligantes mesmo em locais de ligação parcialmente ocupados e fornece medidas de confiança estatística para o sinal identificado.

1.6.3 Soluções de *Design* de Moléculas

Como forma de inspiração para a representação molecular e a rotulação dos dados de densidade eletrônica de ligantes, diversas soluções de *design* de moléculas (construção e descoberta de novas moléculas) foram buscadas. No *design* de moléculas o objetivo principal é a geração contínua de um grafo molecular ou SMILES (notação de linha para descrever estruturas químicas usando *strings* curtas) representando a estrutura química de uma nova molécula a partir das estruturas presentes em um banco de dados de moléculas.

Um primeiro passo crítico na descoberta de novas moléculas é gerar um conjunto de candidatos para estudo computacional ou síntese química e caracterização. Essa tarefa é muito desafiadora, porque o espaço de possíveis moléculas é enorme - o número de potenciais compostos semelhantes à fármacos foi estimado entre 10^{23} e 10^{60} [87], enquanto o número de todos os compostos que já foram sintetizados está na ordem de 10^8.

As principais soluções para esse problema são baseadas em autoencoders, algoritmos de busca com heurísticas de conhecimento de domínio e abordagens de aprendizado profundo. Dentre as soluções existentes para esse problema a *Junction Tree Variational Autoenconders* (JTVAE) [54] se destacou em relação às demais por ser a única solução encontrada que define um vocabulário de fragmentos a partir da decomposição de um banco de dados de moléculas. Essa solução define uma regra para decompor as moléculas em uma árvore de fragmentos (*fragment tree*) e usa o vocabulário de fragmentos obtido para fazer a construção de uma nova molécula em formato de árvore, onde cada nó representa um fragmento do vocabulário. No final da construção molecular a árvore de fragmentos é convertida para um grafo molecular representando a molécula gerada.

Para o problema de construção de modelos moleculares de macromoléculas (proteína, DNA e RNA) as soluções são baseadas na correspondência entre padrões da densidade eletrônica e a presença de um aminoácido (*building blocks*) na estrutura da proteína [66, 47, 19]. Essas soluções fazem o encaixe dos aminoácidos na densidade eletrônica observada e fortalecem a premissa de que existe um padrão na densidade devido a presença de certas subestruturas químicas nas moléculas do cristal. Porém, em comparação com o problema da reconstrução de ligantes, o número de possibilidades para a reconstrução de proteínas é bem menor. Só existem 20 aminoácidos que podem compor uma proteína e isso diminui

substancialmente a quantidade de padrões a serem definidos e detectados.

1.6.4 Soluções baseadas em Aprendizado de Máquina Profundo

O aprendizado profundo é uma tecnologia emergente que foi aplicada em muitas disciplinas da ciência e provou ser bem-sucedida nas tarefas de segmentação, classificação e reconhecimento de imagens [65]. As redes neurais convolucionais profundas (CNN, do inglês: *convolutional neural network*) compreendem uma subclasse de redes de aprendizagem profunda. Os filtros locais nas CNNs varrem o espaço de entrada e detectam padrões locais recorrentes que são úteis para o desempenho da classificação [65]. Ao empilhar várias camadas de CNN, as CNNs profundas compõem hierarquicamente recursos espaciais locais simples em características complexas.

As interações bioquímicas ocorrem localmente e podem ser agregadas no espaço para compor interações complexas e abstratas. O sucesso das CNNs na extração de recursos de imagens 2D sugere que o conceito de convolução pode ser estendido para 3D e já foi aplicado com sucesso a proteínas representadas como "imagens" em 3D [106, 102]. Essas soluções foram baseadas na arquitetura da 3D U-Net [118] para segmentação volumétrica e apresentaram resultados melhores do que aqueles obtidos por outras técnicas da bioquímica computacional (e.g. *docking* de proteínas). Nos últimos anos essa arquitetura também foi utilizada para validar e estimar a resolução de mapas de densidade da Microscopia Cryo-Eletrônica [8], e já havia sido utilizada para avaliar a qualidade de estruturas 3D de RNAs [67] e para predizer cavidades de ligação em proteínas [103, 53].

Recentemente e com arquitetura própria, um trabalho desenvolveu uma abordagem convolucional 3D baseada em rede neural profunda denominada seleção de *docking decoy* (pontuação de modelos de *docking*) com uma abordagem de *voxel* para redes neurais profundas (DOVE) para avaliar modelos de encaixe de proteínas [108]. Outro trabalho com arquitetura CNN 3D própria fez a detecção de estrutura secundária de proteínas em mapas da Microscopia Cryo-Eletrônica com resolução intermediária [71]. Também recentemente, o trabalho DEELIG utilizou uma abordagem baseada em aprendizado profundo apenas com as estruturas atômicas para predizer a afinidade de ligação entre proteína e ligante [3].

Como alternativa ao uso de CNNs, existem as arquiteturas emergentes de nuvens de pontos. A rede neural PointNet [89] trabalha com nuvens de pontos ("point clouds") representando a geometria 3D como entrada, evitando assim o problema de selecionar um tamanho rígido para cada *voxel* e para o *grid*. Foi demonstrado que o modelo da PointNet faz a classificação de objetos 3D com melhor desempenho que os modelos baseados em entradas 3D volumétricas e sua aplicação para imagens biológicas pode ser promissora [8].

Até o presente momento, não conseguimos encontrar um trabalho que tenha aplicado uma arquitetura de aprendizado profundo para classificação ou segmentação semântica de imagens da densidade eletrônica residual da cristalografia de proteínas com o objetivo de identificar ou reconstruir ligantes desconhecidos das proteínas (abordagem sugerida pela solução do *CMB* para identificar novas moléculas). Também não encontramos um banco de dados de *grids* de *voxels* 3D representando a densidade eletrônica residual de moléculas presentes em cristais de proteína. Os *grids* 3D da densidade eletrônica podem

ser gerados usando, por exemplo, a ferramenta *Gemmi* [111], disponibilizada em um pacote do Python, capaz de manipular a densidade eletrônica e convertê-la em um *grid* com determinado espaçamento. Porém, não encontramos a definição de uma rotulação para a densidade eletrônica que ofereça uma modelagem das moléculas presentes no modelo atômico. Essas tarefas foram abordadas neste projeto de mestrado e serão descritas com mais detalhes nas próximas seções.

1.7 Organização dos Próximos Capítulos

A metodologia deste trabalho de mestrado é apresentada no Capítulo 2 e foi particionada em cinco seções, sendo as duas primeiras seções sobre a hipótese e o objetivo deste trabalho. Na seção sobre o objetivo do trabalho são apresentadas as três etapas principais executadas na metodologia. Cada uma dessas três etapas é apresentada em uma seção separada. A seção de cada etapa começa descrevendo seus objetivos de forma modular e é particionada em subseções referentes aos módulos implementados para alcançá-los. Os conceitos de cada módulo são apresentados na sua respectiva subseção à medida que são necessários para o entendimento do seu funcionamento. Dessa forma, a leitura de cada módulo pode ser feita independentemente dos conceitos das demais subseções, sem ser necessária a leitura de todos os conceitos da metodologia para estudar apenas determinados módulos de cada etapa.

A divisão modular da metodologia segue a trajetória do desenvolvimento cronológico deste trabalho, então também conta o caminho percorrido para chegar nos resultados. As tentativas que não foram bem sucedidas são apresentadas ao longo do texto para que o leitor também possa aprender com os erros que foram cometidos.

O capítulo 3 apresenta os resultados e as discussões sobre os modelos de aprendizado profundo obtidos e sobre a aplicação desenvolvida, organizados em duas sessões separadas. Esse capítulo começa apresentando os modelos obtidos, primeiro são apresentados os modelos sem sucesso, seguidos dos melhores modelos obtidos e de análises sistemáticas das configurações utilizadas e, finalmente, o capítulo 3 termina com a avaliação da aplicação.

O capítulo 4 sintetiza as conclusões do trabalho e aponta possibilidades de trabalhos futuros.

Boa leitura!

Capítulo 2

Metodologia

2.1 Hipóteses

A hipótese deste projeto de mestrado é que as subestruturas químicas das moléculas presentes no cristal de proteína deixam padrões na distribuição da densidade eletrônica residual, que quando visualizados a certo contorno induzem formatos nesta imagem característicos das estruturas químicas presentes em cada região. Sendo assim, seria possível treinar um modelo de aprendizado profundo para fazer a segmentação semântica de imagens da densidade eletrônica residual e identificar as subestruturas que preenchem, explicam a nuvem eletrônica observada. E por fim, a segmentação semântica das partes que compõem uma molécula poderia servir de ponto de partida para guiar a reconstrução completa da sua estrutura química.

A premissa dessa hipótese é baseada nos resultados de soluções de modelagem molecular que utilizam descritores da topologia e padrões da densidade eletrônica residual para predizer as estruturas químicas presentes no cristal. Além disso, esse padrão é utilizado hoje, manualmente (no olho do cristalógrafo), para fazer a interpretação da densidade residual. As convoluções de uma rede neural possibilitam que informações da vizinhança de cada ponto da imagem sejam agregadas para gerar a predição de cada ponto. Dessa forma, o modelo preditivo poderá ser capaz de aprender o padrão induzido na distribuição da densidade em toda a região de cada subestrutura do ligante a ser aprendida. Espera-se que a consideração explícita da natureza 3D da imagem da densidade eletrônica melhore o desempenho do modelo a ser construído e que o modelo consiga lidar com os ruídos presentes nesses dados e possibilite a obtenção de uma predição mais precisa da estrutura de um ligante, mesmo que este apresente uma estrutura complexa e inovadora.

2.2 Objetivo

Para treinar um modelo de aprendizado profundo é necessário ter um conjunto de dados rotulados, e idealmente grande, diverso e balanceado, para o qual se tem a entrada e a saída esperada. O objetivo deste projeto é auxiliar na reconstrução da estrutura química de ligantes desconhecidos de PN a partir da imagem 3D da densidade eletrônica residual. Portanto, o conjunto de dados necessário para essa tarefa deve ser composto por imagens

3D da densidade eletrônica residual de ligantes rotuladas, e idealmente conter diferentes ligantes que cubram o espaço químico de moléculas orgânicas de PN com repetições em diferentes conformações e arranjos. Com esse conjunto, seria possível treinar um modelo de aprendizado profundo baseado em segmentação semântica para aprender os padrões presentes nessa rotulação e então ser utilizado para segmentar uma nova entrada. Parte-se da hipótese de que se as subestruturas dos ligantes forem bem entendidas pelo modelo, a composição das partes mais simples resultantes da predição desse modelo poderá ser inferida a posteriori para guiar a reconstrução completa da estrutura molecular de ligantes desconhecidos.

Os passos principais para alcançar o objetivo deste projeto de mestrado são:

1. Criação de um banco de dados de imagens 3D da densidade eletrônica residual de ligantes rotuladas;

2. Treinamento de um modelo de aprendizado profundo para segmentação semântica das imagens desse banco de dados;

3. Criação de uma aplicação para rotular uma nova entrada com o modelo obtido.

O workflow empregado neste projeto é apresentado na Figura 2.1. A partir de uma listagem do PDB, obteve-se os fatores de estrutura (Fo) e o modelo atômico de cada entrada via download das entradas. Com o modelo atômico, é possível extrair todos os ligantes presentes em cada entrada do PDB, selecionar os de interesse e criar uma listagem com todos os ligantes disponíveis. Após obtenção da lista de ligantes que foram selecionados, arquivos no formato SDF foram obtidos via download para a correta inferência das suas estruturas químicas (átomos e ligações) e posições atômicas (x, y, z). A partir das posições dos átomos de cada ligante foi possível delimitar uma caixa ao redor da estrutura do ligante que ao ser expandida cobre toda a imagem do ligante na densidade. Com o SMILES de cada ligante, criou-se vocabulários de fragmentos a partir da decomposição da sua estrutura química, sendo que este vocabulário foi utilizado para anotar cada átomo dos ligantes.

A partir dos dados obtidos das entradas do PDB, foi possível fazer um refinamento inicial para obter os mapas da densidade eletrônica residual de cada entrada. Foram extraídos *grids* 3D dos mapas da densidade eletrônica residual localizados dentro da caixa delimitadora de cada ligante. Diferentes contornos foram aplicados nos *grids* 3D dos ligantes para criar imagens 3D mais finas do formato dos ligantes. A rotulação dos ligantes foi extrapolada para a densidade eletrônica ao redor de cada um dos seus átomos e assim as imagens 3D criadas foram rotuladas. Com os bancos de dados criados, modelos preditivos de aprendizado profundo para imagens 3D foram treinados e avaliados com o objetivo de inferir subestruturas químicas a partir da distribuição da densidade eletrônica. Dependendo do comportamento dos modelos, ciclos de treinamento, avaliação e remodelagem da rotulação e das configurações foram realizados até que bons resultados foram alcançados. Com um bom modelo, é possível utilizá-lo para rotular uma nova entrada.

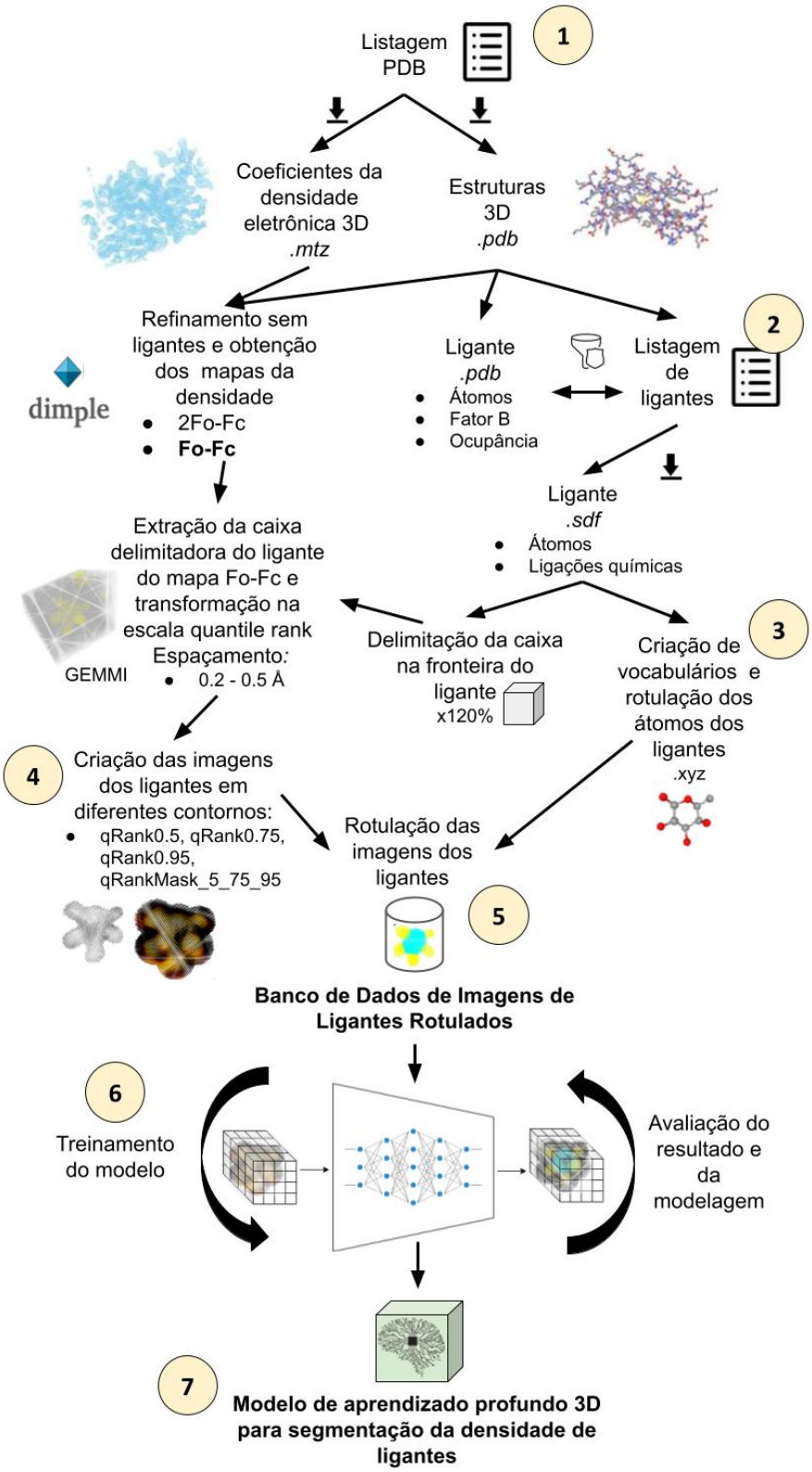

Figura 2.1: Proposta de workflow para obtenção do modelo preditivo de aprendizado profundo 3D para segmentação da imagem da densidade eletrônica de ligantes. As etapas principais do workflow foram numeradas e estão associadas no texto. As etapas 1 e 2 estão descritas na Seção 2.3.1, a etapa 3 na Seção 2.3.2, a etapa 4 na Seção 2.3.3, a etapa 5 na Seção 2.3.4, a etapa 6 na Seção 2.4 e a etapa 7 na Seção 3.1.3.

2.3 Criação de um Banco de Dados de Imagens 3D da Densidade Eletrônica Residual de Ligantes Rotuladas: Modelagem e Engenharia de Dados

A primeira etapa deste projeto de mestrado foi a criação do banco de dados de imagens 3D de ligantes rotuladas. Essa etapa consistiu na modelagem da imagem da densidade eletrônica e se dividiu em quatro grandes objetivos:

1. Criação de uma listagem confiável de ligantes para os quais existe densidade eletrônica residual;
2. Criação de propostas de rotulação dos ligantes;
3. Criação das imagens da densidade residual dos ligantes;
4. Extrapolação da rotulação dos ligantes para sua imagem da densidade eletrônica residual.

A criação de uma listagem de ligantes pode ser obtida a partir do PDB e também engloba definir as características que serão utilizadas para filtrar e selecionar as entradas que serão incluídas. A criação das imagens engloba definir o formato da imagem, sua representação e a escolha e atribuição de características a elas, extraídas do mapa de densidade eletrônica residual. A criação da rotulação engloba definir um vocabulário para as estruturas químicas dos ligantes formado por classes que estejam relacionadas com os padrões observados na densidade eletrônica e extrapolar essa rotulação da estrutura do ligante para a sua imagem de densidade residual.

2.3.1 Listagem de Entradas de Ligantes do PDB com Densidade Eletrônica Residual

O PDB contém os dados experimentais obtidos a partir da difração de raio X de cristais de macro moléculas (fatores de estrutura no formato .mtz) e os modelos atômicos (coordenadas no formato .pdb) em diferentes resoluções. Alguns dos modelos depositados também contêm a estrutura química de pequenas moléculas presentes no cristal, que podem ser ligantes, solvente, água, agentes cristalizantes, etc. Neste projeto estamos interessados em dados de cristalografia de proteínas de raio X, que contenham os coeficientes de difração experimentais utilizados para obter o mapa da densidade eletrônica e que possuam pequenas moléculas ligantes que sejam moléculas orgânicas que ocupam o espaço químico de PNs.

O PDB é um banco de dados construído pela comunidade científica. Até fevereiro de 2008 poucas métricas de qualidade eram verificadas para o depósito de novas entradas e não era obrigatório depositar os dados experimentais da difração de raio X, que comprovam a estrutura depositada. Logo, para obter uma listagem confiável do PDB é necessário pensar em garantias de qualidade das imagens ali presentes para incluir entradas que possuem uma correspondência entre a estrutura química e o mapa da densidade

eletrônica depositados. Além disso, a resolução da entrada influencia muito na qualidade da imagem da densidade eletrônica e na consistência dos padrões observados (ilustrado na Figura 1.8 da introdução). Resoluções muito ruins, acima de 3.5 Å, possuem muito ruído e comprometem a correspondência entre estruturas químicas e suas imagens na densidade eletrônica. Resoluções muito boas, abaixo de 1.5 Å, apresentam uma densidade muito melhor definida ao redor dos átomos, em contrapartida, com ruídos mais nítidos de diferentes conformações. Porém, resoluções muito boas não correspondem à realidade da maioria dos projetos que utilizam cristalografia, uma vez que a resolução depende da qualidade do cristal, que pode ser intrínseca a uma dada proteína.

O site do PDB [10] fornece uma ferramenta de busca avançada capaz de gerar relatórios com todas as entradas que obedecem uma seleção de filtros definidos pelo usuário. Para obter a listagem de todas as entradas do PDB (etapa 1 do workflow da Figura 2.1) de cristalografia de proteínas de raio X que contêm ligantes e dados experimentais (Fo), os seguintes filtros foram utilizados (acessado em julho de 2019): *has ligand = yes*; *Experimental type= x-ray & molecule type = protein*; *Experimental Method = x-ray & Has Experimental Data = True.*

Dadas essas informações, decidiu-se filtrar a listagem de ligantes presentes na lista de entradas do PDB para incluir apenas entradas depositadas após fevereiro de 2008 (2008-02-01), de ligantes livres (não fazem parte da estrutura da proteína), não covalentes e de moléculas orgânicas (formadas pelos átomos de carbono, hidrogênio, oxigênio, nitrogênio, fósforo, enxofre, iodo, flúor, selênio, cloro e bromo). A partir dessa listagem avaliamos a distribuição das entradas válidas de ligantes por faixa de resolução e o impacto do filtro de moléculas orgânicas (+NP) em cada faixa (Tabela 2.1).

Tabela 2.1: Sumário das contagens de entradas do PDB nas faixas de resolução escolhidas com filtro da data de depósito a partir de 2008 e adição do filtro de produtos naturais. As estatísticas descritivas são referentes à contagem de ligantes.

Faixa de Resolução	**[1.0,1.5)**		**[1.5,1.8)**		**[1.8,2.2)**		**[2.2,2.5]**	
Filtro	2008	+PN	2008	+PN	2008	+PN	2008	+PN
Número de Entradas do PDB	9957	9440	18728	17751	33693	31666	19207	18001
Número de ligantes únicos	3868	3690	6687	6432	10904	10592	7213	7009
Contagem de ligantes	67986	54429	142789	116575	293195	241375	177322	145490
Mínima	1	1	1	1	1	1	1	1
1st Quartil	1	1	1	1	1	1	1	1
Mediana	1	1	2	2	2	2	2	2
Média	17.1	14.8	20.8	18.1	26.4	22.8	24.2	20.8
3rd Quartil	3	3	3	3	4	3	4	4
Máxima	8313	8313	14666	14666	32566	32566	20647	20647
Desvio Padrão	219.6	212.4	349.1	338.5	561.6	542.3	409.3	391.5

A faixa de resolução entre 1.5 e 2.2 Å foi escolhida, minimizando assim variações muito grandes devidas à resolução dos dados. Essa faixa de resolução mantém uma alta contagem de ligantes e permite trabalhar com dados em uma resolução próxima aos dados mais comumente obtidos, inclusive os coletados no CNPEM. Foi obtida a listagem final de ligantes com 458058 entradas de ligantes, vindos de 55884 entradas do PDB, contendo 16180 ligantes distintos. As variáveis estatísticas descritivas da contagem de ligantes distintos evidencia o enorme desbalanço existente entre os tipos de ligantes presentes, com contagens que vão de 1 a 47453 ligantes por tipo, com um desvio padrão igual a 727.2.

Em seguida, para obtenção da listagem final de ligantes válidos, as entradas de ligantes com problemas na sua estrutura devido a depósitos errados no PDB foram excluídas por apresentarem erros na leitura dos seus arquivos SDF ou na construção do seu grafo molecular representativo. Também foram excluídas dessa listagem as entradas do PDB que apresentaram erro no seu refinamento para obtenção dos mapas da densidade eletrônica residual (mais detalhes na Seção 2.3.3). A listagem final de ligantes válidos (etapa 2 do workflow da Figura 2.1) foi obtida com 244081 entradas de ligantes de 36182 entradas do PDB. Essa listagem possui 12235 tipos de ligantes únicos, com repetições que variam de

1 a 33091 ocorrências por tipo de ligante com desvio padrão igual a 525.8. Essa queda de quase 50% das entradas da listagem final de ligantes evidencia a falta de validação nas entradas do PDB e a dificuldade de fazer mineração de dados com esse conjunto.

A listagem obtida contém todas as entradas de ligantes possíveis de serem utilizadas no treinamento do modelo de aprendizado profundo. Com essa listagem é possível saber o que pode ser incluído no conjunto de dados e o que não pode, logo, é possível pensar em uma rotulação para essas estruturas. Essa listagem final ainda foi revista para balancear as classes de cada proposta de rotulação que foram aprendidas pelos modelos preditivos. Como a rotulação foi baseada em subestruturas químicas mais simples e mais comuns aos ligantes, espera-se que as contagens de cada classe sejam menos desbalanceadas do que as contagens dos ligantes únicos. Com o intuito de evitar um mal desempenho do modelo inicial devido a diferenças entre as imagens de ligantes vindas de entradas em diferentes resoluções da densidade eletrônica residual, apenas as imagens em um intervalo menor de resolução, entre 1.5 Å e 1.8 Å, e depois entre 1.8 Å e 2.2 Å, foram utilizadas na validação da hipótese (contagem presente na Tabela 2.1).

Nesse momento também teria sido de grande importância obter uma avaliação do encaixe da estrutura dos ligantes nos seus mapas de densidade eletrônica residual. O encaixe de ligantes pode ser calculado utilizando métricas já bem descritas para essa tarefa (pontuadas na Seção 1.4) ou uma combinação delas. Essa avaliação poderia fornecer uma nota de qualidade mais confiável para cada entrada e seria de grande valia para incrementar a filtragem e avaliar a qualidade da listagem obtida. Mas essa tarefa se mostrou difícil de ser executada em larga escala e requeria um conhecimento maior dos dados e das ferramentas existentes. Por esse motivo, não foi possível obter as notas de encaixe nessa etapa e foi decidido tentar fazer essa avaliação em um momento mais avançado do projeto caso houvesse tempo.

A obtenção da listagem final de ligantes foi automatizada em um *script* parametrizado com os filtros descritos (faixa de resolução, data de depósito, ligante livre, mínima ocorrência do tipo de ligante e moléculas orgânicas) para facilitar futuras alterações e adequações. A entrada para esse processo é a lista completa de entradas do PDB com ligantes de experimentos de difração de raio X de cristais de proteína vinda da busca avançada do site do PDB. Essa listagem contêm os ligantes únicos presentes em cada entrada e é utilizada para fazer o download dos dados do PDB. A listagem completa de ligantes é obtida extraindo do modelo atômico de cada entrada do PDB todas as entradas de ligantes que estão na sua respectiva lista de ligantes únicos (algumas entradas do PDB contêm mais do que um exemplar de cada ligante único). Essa listagem completa é então utilizada para baixar os arquivos SDF de cada ligante, que contém suas posições atômicas (x, y, z) e as suas ligações químicas, e por isso possibilita que o grafo molecular representativo de cada ligante seja obtido com fidelidade ao que foi depositado e na conformação correta. Finalmente, o pacote RDKit [50] do Python é utilizado para validar os arquivos SDF baixados (testando se é possível criar um grafo molecular válido) e então a lista final de ligantes válidos é criada.

2.3.2 Rotulação da Estrutura dos Ligantes

A rotulação de conjuntos de dados pode ser vista como a criação de um vocabulário de classes com as quais é possível descrever de forma não ambígua todas as entradas do conjunto. Ou seja, é um vocabulário a partir do qual é possível voltar ao dado inicial ou a características utilizadas para descrever esse dado. A escolha de rotulações precisas é uma tarefa crítica para o desempenho de um modelo de aprendizado supervisionado e também muito propensa a erros. Para dados específicos, pouco utilizados pelas comunidades científicas de outras áreas, é difícil se conhecer uma rotulação que capture bem as características do dado necessárias para treinar um modelo de aprendizado supervisionado capaz de aprender os padrões ali presentes. Além disso, fazer a rotulação completa de um novo conjunto de dados é uma tarefa muito trabalhosa e que precisa ser automatizada para ser realizada rapidamente.

Não foi encontrada uma referência para rotulação de subestruturas na imagem de densidade eletrônica residual de ligantes, o que logo se mostrou não ser uma tarefa fácil. A escolha da rotulação da densidade dos ligantes implica nas classes que o modelo de aprendizado profundo terá que aprender e que poderá ser capaz de predizer em uma nova entrada. Portanto, a rotulação implica em definir uma modelagem capaz de capturar os padrões presentes na imagem da densidade do ligante e que permita treinar um modelo de aprendizado profundo que auxilie ao máximo a reconstrução da estrutura química do ligante. Não é possível saber qual a melhor rotulação sem antes treinar o modelo com cada proposta e depois avaliar como ele se comporta. Testar todas as possíveis propostas de rotulação é inviável, mas é possível inferir quais rotulações são inviáveis devido à complexidade do modelo que ela implica (quantidade e frequência das classes) ou devido a ambiguidades entre as classes (impossibilidade de rotulação única).

A hipótese deste projeto de mestrado é que subestruturas químicas dos ligantes implicam em padrões na imagem da densidade eletrônica residual. Dessa forma, a rotulação da estrutura química dos ligantes pode ser extrapolada para uma região ao seu redor e ser utilizada para rotular suas imagens na densidade eletrônica residual. Logo, rotulações de estruturas químicas podem ser utilizadas como pontos de partida para a rotulação da densidade eletrônica residual de ligantes.

Entretanto, não foi encontrada uma rotulação definida para estruturas químicas que modelem a distribuição da densidade eletrônica. O que se tem como alternativa é o uso de algoritmos de decomposição (quebra) de moléculas em subestruturas químicas, ou fragmentos. Soluções de aprendizado de máquina que modelam estruturas químicas de pequenas moléculas foram buscadas na literatura para encontrar inspirações para a modelagem dos ligantes. Na pesquisa de design de moléculas, existem muitas abordagens baseadas em aprendizado de máquina para construção de novas estruturas químicas de moléculas viáveis, e a solução JTVAE [54] se destacou. A solução JTVAE se baseia na criação de um vocabulário de fragmentos a partir da decomposição das estruturas químicas de um conjunto de dados de moléculas. Esse vocabulário é utilizado na representação de uma molécula em *fragment-tree* usando um grafo submolecular, onde cada um desses fragmentos é um nó da árvore. Os nós conectados na árvore compartilham pelo menos um átomo, e logo, isso pode gerar uma rotulação multiclasse para cada átomo envolvido em

mais do que uma ligação distinta. Nós auxiliares precisam ser inseridos na representação de árvore para remover quaisquer ciclos remanescentes no grafo submolecular criado com as rotulações. E então é feita a construção de novas estruturas químicas a partir da construção de grafos submoleculares conectando fragmentos desse vocabulário.

Para rotular a densidade, não é necessário representar o ligante em uma árvore e a adição de nós auxiliares não é de grande proveito para este objetivo. O pacote RDKit [50] do Python fornece funcionalidades para representar moléculas em grafos moleculares e para manipular os grafos moleculares criados. Funcionalidades como extração de todos os ciclos simples do grafo molecular, *matching* entre grafos moleculares e geração de SMILES a partir de subestruturas do grafo molecular são exemplos de rotinas que o RDKit oferece. Baseado no algoritmo do JTVAE o Algoritmo 1 foi implementado para decompor uma molécula em fragmentos simples (ciclos e arestas) e construir um vocabulário a partir dos fragmentos únicos criados para decompor todas as moléculas da listagem de ligantes válidos. Esse vocabulário poderia ser utilizado para rotular cada um dos átomos das moléculas quebradas, utilizando os SMILES dos seus respectivos fragmentos. É interessante manter a rotulação de ciclos, pois essas subestruturas são facilmente detectadas a olho nú na densidade eletrônica e obedecem restrições geométricas interessantes (i.e., rigidez, planaridade) para a distribuição da densidade eletrônica ao seu redor.

Algorithm 1 Algoritmo de Decomposição de Moléculas em Fragmentos Simples

```
procedure DecomposeMolSimple(G(V, E))  ▷ Decomposição do grafo molecular G em Ciclos simples e arestas
    F1 ← o conjunto de ciclos simples de G
    F2 ← o conjunto de arestas (u, v) ∈ E que não pertencem a um ciclo
    F ← F1 ∪ F2                                   ▷ Conjunto dos fragmentos de G
    S ← {}                                        ▷ Conjunto dos SMILES de F
    for frag in F do
        S ← S ∪ {SMILES gerado para o frag em G}
    return F, S      ▷ A lista de fragmentos simples de G e seus respectivos SMILES
```

Esse algoritmo foi utilizado na decomposição dos ligantes válidos presentes na listagem final entre 1.5 Å e 1.8 Å e criou um vocabulário com 406 fragmentos. Essa quantidade de classes evidencia uma inviabilidade de treinamento de um modelo de aprendizado profundo, pois seriam necessárias muitas entradas para que o modelo conseguisse aprender tantas classes. Além disso, os fragmentos retornados pelo JTVAE incluem pelo menos dois átomos e uma ligação ou são estruturas cíclicas maiores, sendo que dois fragmentos seguidos na estrutura de uma molécula podem ter pelo menos um átomo em comum. Isso implica em uma ambiguidade na rotulação das imagens dos ligantes, pois existiriam muitas sobreposições entre as classes e um modelo multi classe adiciona complexidade ao treinamento e requer uma grande diversidade no conjunto [112]. Apesar disso, o algoritmo do JTVAE mostrou um exemplo de manipulação de estruturas químicas utilizando a biblioteca RDKit e possibilitou pensar em alternativas viáveis. Um questionamento levantado nessa análise foi qual fragmentação cria subestruturas químicas cujo padrão é visível na densidade eletrônica, ou seja, qual fragmentação respeita a hipótese deste projeto.

Algumas simplificações no algoritmo de fragmentação do JTVAE foram testadas com

o objetivo de diminuir a quantidade de classes criadas e obter um vocabulário viável. Foram implementadas simplificações para ignorar o tipo de átomo (todos convertidos para um átomo genérico) e o tipo de ligação (todas convertidas para ligações simples). Com essas simplificações, foi possível obter vocabulários menores, ainda assim o número de estruturas cíclicas se manteve grande e muitas com pouca frequência. Essas simplificações ainda mantinham o problema de sobreposição de rotulações, evidenciando que a rotulação deveria partir dos átomos para ser única e não das ligações entre os átomos, dado que diferentes ligações compartilham os mesmos átomos. Além disso, as classes criadas a partir dessas simplificações carregavam pouca informação do padrão visível na densidade eletrônica e portanto se mostrou necessário pensar em outras alternativas de rotulação.

Especialistas da área de cristalografia foram consultados para auxiliar na proposta de rotulações viáveis e que pudessem ter relação com os padrões visíveis nas imagens 3D da densidade eletrônica. Uma rotulação pelo tipo do átomo (carbono, oxigênio, nitrogênio, etc.) não foi vista como muito promissora pelos especialistas pois a quantidade de ruído presente nos dados de cristalografia de proteínas nas resolução utilizadas usualmente não permite distinguir a olho nu o tipo de átomo presente em cada região da imagem. Entretanto, essa rotulação é a mais direta quando se pensa em rotular uma molécula e por isso ainda foi considerada como alternativa para ser testada ao longo do projeto. Sugeriu-se também o uso da hibridização SP dos átomos como alternativa de rotulação.

A hibridização SP define a conformação espacial 3D que cada átomo terá no seu arranjo molecular devida a quantidade de nuvens de elétrons ao seu redor, o que parecia ser ideal para a aplicação proposta. A hibridização SP de cada átomo é definida pelo seu número estérico. O número estérico é igual ao número de pares de elétrons ao redor do átomo, onde cada par de elétrons livres conta como um e toda ligação, independentemente do seu tipo, também conta como um par de elétrons. Logo, o número estérico é igual ao número de pares de elétrons livres ao redor do átomo mais o número de ligações químicas que ele faz. A ideia da hibridização SP é que pares de elétrons ao redor dos átomos se repelem, e logo, impõem conformações pré-definidas para os átomos devidas as restrições geométricas de distância e ângulos entre eles. Por exemplo, o átomo de carbono com 4 ligações simples, número estérico igual a 4 e hibridização ‘sp3’, vai assumir a conformação 3D de um tetraedro e por isso é chamado de carbono tetraédrico. A Figura 2.2 exemplifica diferentes conformações 3D devido às hibridização SP de cada átomo. Como a hibridização SP considera a conformação 3D dos átomos, ela se mostrou promissora para conseguir modelar os padrões presentes na densidade eletrônica devida a conformação espacial 3D das moléculas. Entretanto, para avaliar se uma proposta de rotulação consegue capturar os padrões presentes nos dados é necessário testá-la no treinamento de um modelo de aprendizado de máquina.

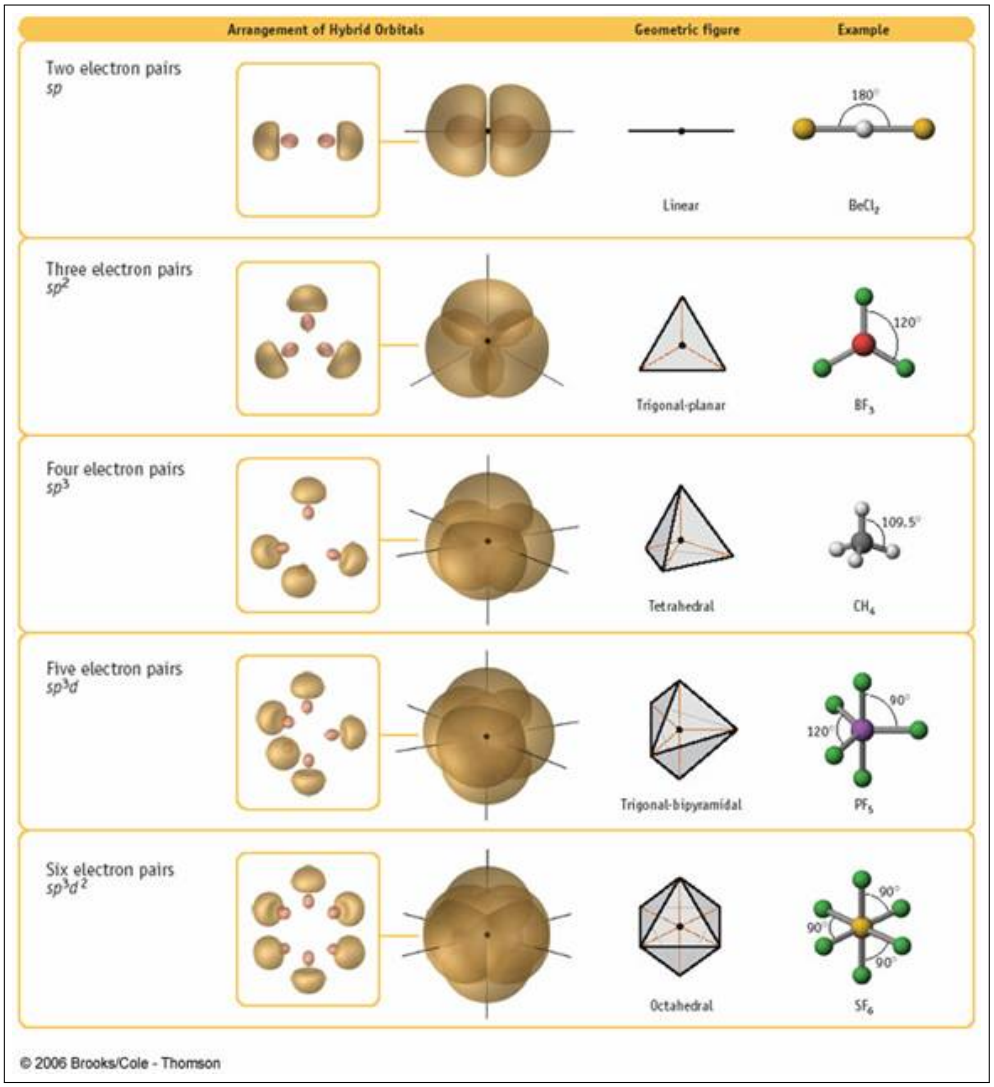

Figura 2.2: Diferentes conformações 3D das órbitas dos átomos devido a sua hibridização SP. Fonte: http://www.fccj.us/chm2045/StericNumber.htm

O Algoritmo 1 foi utilizado como referência para implementar um algoritmo de rotulação baseado na hibridização SP de cada átomo. O pseudo-código deste algoritmo é apresentado no Algoritmo 2.

Algorithm 2 Algoritmo de rotulação com hibridização SP

1: **procedure** ROTULACAOSP($G(V, E)$) ▷ Rotulação SP para todos os átomos em V
2: $SP \leftarrow \{\}$ ▷ Conjunto da hibridização SP dos átomos
3: **for** `átomo in V` **do**
4: $SP \leftarrow SP \cup \{$Hibridização SP calculada para o `átomo` em G$\}$
5: **return** V, SP ▷ A lista de átomos de G e suas respectivas hibridizações SP

Esse algoritmo foi utilizado para obter as classes necessárias para rotular os ligantes válidos da listagem final. Ele resultou em um vocabulário com 7 classes, contando a classe do ruído de fundo, chamado aqui de "Vocabulário de Classes SP", e logo, mostrou-se viável para ser testado. As classes são todas as hibridizações que aparecem nas estruturas químicas da listagem de ligantes: sp, $sp2$, $sp3$, $sp3d$, $sp3d2$ e $sp3d3$.

Um fato muito destacado pelos especialistas da área de cristalografia desde o início deste projeto é que as estruturas cíclicas, principalmente ciclos com até 7 átomos (i.e., pentano e hexano), deixam uma marca muito característica na densidade eletrônica. Essa marca é um anel, uma região com um buraco na densidade do ligante, que indica a presença de um ciclo ao seu redor. Os ciclos são estruturas químicas mais rígidas, com mobilidade restrita pelo arranjo geométrico circular dos átomos, e por isso costumam deixar sua marca bem definida na densidade eletrônica. Além disso, a aromaticidade dos ciclos, caracterizada por átomos sp2 (ligações simples e dupla alternadamente na sua estrutura), também pode ser percebida na densidade eletrônica. Isso acontece porque os ciclos aromáticos são estruturas ainda mais rígidas e que possuem uma conformação 3D planar muito característica. Como a densidade eletrônica é um consenso entre as diferentes conformações que a molécula pode assumir dentro do cristal de proteína, as estruturas mais rígidas têm uma densidade eletrônica melhor definida e menos ruidosa. É isso que acontece com as estruturas cíclicas e por isso incluir essas estruturas no vocabulário mostrou-se ser bastante promissor. Essa marca na imagem da densidade eletrônica é menos característica para ciclos com mais do que 7 átomos, os quais têm menos restrições geométricas, e logo, mais mobilidade.

A partir dessa ideia, o Algoritmo 2 de rotulação com hibridização SP foi modificado para acoplar nas classes SP o tamanho do ciclo em que cada átomo aparece. Dessa forma, a classe SP de cada átomo foi acoplada a um ciclo de certo tamanho em que ele aparece, definido como CX ou CAX, onde $3 <= X <= 7$, C é um ciclo não aromático e CA é um ciclo aromático. Ciclos com mais do que 7 átomos não são considerados na rotulação e seus átomos recebem apenas sua classe SP. O pseudocódigo obtido é apresentado no Algoritmo 3.

Algorithm 3 Algoritmo de rotulação com hibridização SP com Ciclos de tamanho 3 à 7 aromáticos ou não

procedure ROTULACAOSPCAX($G(V, E)$) ▷ Rotulação SP para todos os átomos em V acoplada com o tamanho e o tipo do ciclo em que eles aparecem
  $F1 \leftarrow$ o conjunto de ciclos simples de G até 7 átomos
  $SP\_ciclo \leftarrow \{\}$ ▷ Conjunto da hibridização SP dos átomos com informação do tipo e tamanho do ciclo em que eles aparecem
  for `átomo in V` **do**
    $atom\_ciclo \leftarrow$ {ciclo de `F1` em que o `átomo` aparece} ▷ {CX,CAX} se o `átomo` aparece em um ciclo de `F1` de tamanho X e CA se é aromático, se não {}
    $atom\_SP \leftarrow$ {Hibridização SP calculada para o `átomo` em G}
    $SP\_ciclo \leftarrow SP\_ciclo \cup$ {`atom_SP` + `atom_ciclo`}
  return V, SP_ciclo ▷ A lista de átomos de G e suas respectivas hibridizações SP com informação de ciclos

O Algoritmo 3 levou a uma ambiguidade na rotulação, pois existem estruturas químicas com ciclos acoplados denominadas policíclicas. Existem átomos nas estruturas de policíclicos que pertencem a mais do que um ciclo e esses ciclos podem ser de diferentes tamanhos. Logo, tais átomos poderiam receber mais do que uma classe seguindo este algoritmo. Para evitar uma rotulação múltipla, dada suas dificuldades [112], alternati-

vas foram estudadas para rotular esses casos. As possibilidades encontradas foram: (1) criar um novo vocabulário com classes policíclicas para átomos que aparecem em mais do que um ciclo, igual ao número de ciclos em que o átomo aparece (i.e. novas classes $sp3CC$, $sp3CCC$, $sp3CCCC$ e assim por diante); (2) criar um novo vocabulário com novas classes acoplando as classes do tipo de ciclo em que o átomo aparece (i.e. criar classes sp3C5C6 e sp3C6CA6 por exemplo); (3) criar uma regra de precedência para resolver os conflitos de átomos em estruturas policíclicas, sem alterar o vocabulário.

O Algoritmo 3 de rotulação SP com ciclos foi modificado para avaliar o impacto das três alternativas no vocabulário final. A primeira alternativa adicionou 6 novas classes ao vocabulário, o que ainda poderia ser uma tentativa válida quanto à complexidade do modelo. Entretanto, essa primeira alternativa cria uma quebra no padrão da imagem, pois o bloco de ciclos acoplados poderia receber diferentes rotulações em cada pedaço e o formato dos ciclos seria perdido na imagem. Além disso, são poucas estruturas que possuem essa formação policíclica, o que poderia ser um complicador para o modelo conseguir aprender bem essas classes. A segunda alternativa acabou criando muitas classes novas, resultantes de todas as combinações de ciclos que aparecem nas estruturas policíclicas da listagem final de ligantes. Logo, a segunda alternativa se mostrou inviável e foi descartada. A terceira alternativa não altera o vocabulário, e logo, poderia ser viável. De forma similar à primeira alternativa, a terceira também pode quebrar o padrão da imagem quando ciclos de tamanhos diferentes estão acoplados. Mas nessa terceira alternativa pelo menos um dos ciclos tem seu padrão mantido. Foi decidido testar apenas a terceira alternativa.

A regra de precedência implementada na terceira alternativa atribui maior prioridade para rotular os ciclos com as estruturas mais rígidas, para os quais se espera que a densidade eletrônica esteja melhor definida. Os ciclos de tamanho menor são mais rígidos do que o ciclos de tamanho maior, e os ciclos aromáticos são mais rígidos do que os ciclos não aromáticos do mesmo tamanho. Portanto, a precedência implementada dá prioridade para rotular os átomos de estruturas policíclicas com a classe do ciclo de menor tamanho em que ele aparece e caso ele apareça em ciclos do mesmo tamanho a prioridade fica para o ciclo aromático, caso exista conflito. A terceira alternativa criou o "Vocabulário de Classes SP com Ciclos CA34567" e o Algoritmo 4 descreve sua implementação em alto nível.

As duas propostas iniciais de rotulação apresentadas são tentativas de criar uma modelagem capaz de capturar toda a complexidade presente no dado e no problema sendo estudado. Porém, neste caso em que não foi encontrada uma referência de rotulação que funcione para o objetivo e para o tipo de imagem 3D sendo estudada, começar com propostas já muito complexas se mostrou uma tentativa bastante arriscada. Os treinamentos com essas rotulações e a frequência das classes de cada proposta nos conjuntos de treinamento utilizados serão mostrados com detalhes na próxima seção sobre o treinamento do modelo de aprendizado profundo. Mas o resultado desses treinamentos será brevemente adiantado nesta seção para apresentar todas as propostas de rotulação aplicadas e o aprendizado proporcionado pelas decisões tomadas.

Não foi possível obter um bom desempenho no modelo de aprendizado profundo para nenhuma dessas duas propostas, e apenas depois de muitas tentativas de treinamentos sem sucesso foi decidido simplificar as rotulações e testar o básico. As propostas mais simples

Algorithm 4 Algoritmo de rotulação SP com Ciclos de tamanho 3 à 7, aromáticos ou não, com regra de precedência

procedure ROTULACAOSPCAX($G(V, E)$) ▷ Rotulação SP com Ciclos para todos os átomos em V
  $F1 \leftarrow$ o conjunto de ciclos simples de G até 7 átomos
  $SPCAX\_atoms\_labels \leftarrow \{\}$ ▷ Conjunto da hibridização SP dos átomos com o ciclo de maior precedência em que eles aparecem
  for `átomo in V` **do**
    $atom\_label \leftarrow$ {Hibridização SP calculada para o `átomo` em G}
    if `átomo` in `F1` **then**
      $atom\_label \leftarrow atom\_label+$ {ciclo de `F1` com maior precedência em que o `átomo` aparece} ▷ {CX,CAX} se o `átomo` aparece em um ciclo de `F1` de tamanho X e CA se é aromático
      $SPCAX\_atoms\_labels \leftarrow SPCAX\_atoms\_labels \cup \{atom\_label\}$
  return $V, SPCAX\_atoms\_labels$ ▷ A lista de átomos de G e suas respectivas rotulações de hibridizações SP com ciclos

de rotulação que serão apresentadas a seguir possibilitaram obter bons desempenhos no modelo de aprendizado profundo e a partir desses resultados positivos foi possível chegar em propostas um pouco mais complexas. Testar primeiro propostas mais simples na aplicação de um novo método é um ponto de partida com mais garantias de que propostas mais complexas podem funcionar. Esta abordagem é também importante para testar uma premissa, ainda mais quando uma nova hipótese metodológica é lançada - como feito neste projeto. Testar primeiro as propostas complexas, sem saber se as simples funcionam, é um risco de não chegar em nenhuma solução.

Uma proposta simples para modelagem da imagem da densidade eletrônica de ligantes é a rotulação apenas da região do ligante, ou seja, é uma rotulação para separar na imagem da densidade a região onde o ligante se encontra e a região de ruído de fundo ("background"). Se o modelo for capaz de aprender a rotular a região do ligante, então se tem uma maior garantia de que ele seja capaz de aprender rotulações mais complexas que segmentam essa região. Se o modelo não consegue aprender uma rotulação simples, há menos chances dele ser capaz de aprender classes mais complexas, ou é um indicativo que a modelagem aplicada na rotulação simples deveria ser reformulada, ou que a premissa não é válida.

As rotulações mais complexas apresentadas correspondem a diferentes tipos de conformações geométricas de átomos e ciclos de diferentes tamanhos. Uma modelagem mais simples corresponde apenas em rotular átomos genéricos (independente do tipo de átomo e da sua conformação) fora de ciclos e ciclos genéricos (independente do tamanho). Começar do simples e adicionar complexidade aos poucos na modelagem, dependendo de como é a resposta do modelo e como ele se comporta em cada caso, mostrou-se uma estratégia com mais garantias de se chegar em modelos mais complexos e com bom desempenho.

Outras quatro propostas de rotulação simplificadas foram testadas. A proposta mais simples foi a rotulação apenas da região do ligante e da região de ruído de fundo. Com essa proposta foi obtido o "Vocabulário da Região do Ligante", com apenas duas clas-

ses. A segunda proposta foi a rotulação de átomos fora de ciclos (átomos genéricos, de qualquer tipo), de ciclos genéricos (de qualquer tamanho) e ruído de fundo. Com essa segunda proposta foi obtido o "Vocabulário de Átomos e Ciclos Genéricos", com três classes. A terceira proposta testada adicionou complexidade à classe de ciclos deste último vocabulário, com a rotulação de átomos fora de ciclos (genéricos), de ciclos de tamanho de 3 a 7, sendo os ciclos de tamanhos 5 e 6 aromáticos ou não, e ruído de fundo. Os ciclos aromáticos de tamanhos 3, 4 e 7 não foram discriminados dos ciclos não aromáticos devido a sua baixa ocorrência. Com essa proposta foi obtido o "Vocabulário de Átomos Genéricos e Ciclos C347CA56", que contém nove classes. De forma semelhante, foi criada a ultima proposta de rotulação com o "Vocabulário de Classes SP e Ciclos C347CA56" com as classes $sp3dX$, onde $X >= 1$, agrupadas e mapeadas para uma única classe genérica, chamada de $sp3dx$, devido a suas baixas ocorrências. Essas quatro propostas de vocabulário simplificado puderam ser testadas utilizando um mapeamento das classes do "Vocabulário de Classes SP com Ciclos CA34567" para os novos vocabulários propostos, sem ser necessário implementar outro algoritmo para rotular as imagens novamente. Esse mapeamento foi implementado na função de leitura das imagens na pipeline de treinamento e é aplicado antes das entradas serem passadas para o modelo. O mapeamento de classes durante o treinamento aumenta o tempo de processamento de cada entrada, mas permite que propostas derivadas de transformações em vocabulários implementados sejam avaliadas com mais facilidade. As classes do vocabulário "Vocabulário de Classes SP com Ciclos CA34567" e dos demais vocabulários resultantes dos 5 mapeamentos propostos são apresentados na Tabela 2.2.

Tabela 2.2: Vocabulário de classes SP com ciclos CA34567 e seus 5 mapeamentos.

Vocabulário de Classes SP com Ciclos CA34567	**Mapeamento 1 Vocabulário de Classes SP**	**Mapeamento 2 Vocabulário de Classes SP e Ciclos C347CA56**	**Mapeamento 3 Vocabulário da Região do Ligante**	**Mapeamento 4 Vocabulário de Átomos e Ciclos Genéricos**	**Mapeamento 5 Vocabulário de Átomos Genéricos e Ciclos C347CA56**
sp2CA5	sp2	CA5	átomo	C	CA5
sp3C3	sp3	C3	átomo	C	C3
sp2C5	sp2	C5	átomo	C	C5
sp3C4	sp3	C4	átomo	C	C4
sp3C5	sp3	C5	átomo	C	C5
sp2C7	sp2	C7	átomo	C	C7
sp2CA4	sp2	C4	átomo	C	C4
sp3C7	sp3	C7	átomo	C	C7
sp3C6	sp3	C6	átomo	C	C6
spCA6	sp	CA6	átomo	C	CA6
sp3CA7	sp3	C7	átomo	C	C7
sp3d	sp3dx	sp3dx	átomo	átomo	átomo
sp3d3	sp3dx	sp3dx	átomo	átomo	átomo
sp3CA6	sp3	CA6	átomo	C	CA6
sp2C3	sp2	C3	átomo	C	C3
sp2C4	sp2	C4	átomo	C	C4
sp2CA6	sp2	CA6	átomo	C	CA6
sp3d2	sp3dx	sp3dx	átomo	átomo	átomo
sp2C6	sp2	C6	átomo	C	C6
sp2CA7	sp2	C7	átomo	C	C7
sp3CA5	sp3	CA5	átomo	C	CA5
sp2	sp2	sp2	átomo	átomo	átomo
sp3	sp3	sp3	átomo	átomo	átomo
sp	sp	sp	átomo	átomo	á
background	background	background	background	background	background

O Algoritmo 4 de rotulação em classes SP com ciclos também foi modificado para criar o vocabulário baseado no tipo de átomo, denominado "Vocabulário de Tipos de Átomos com Ciclos CA34567". Esse algoritmo é mais simples, no lugar da hibridização SP de cada átomo ele retorna o tipo químico de cada átomo, que são aqueles tipos utilizados para filtrar moléculas orgânicas (filtro de PN), e concatena nessa rotulação o tamanho do ciclo de maior precedência em que ele aparece. Os tipos dos átomos se apresentam como as classes mais diretas quando se pensa em modelar moléculas. Porém, devido ao grande ruído presente nos dados de cristalografia de proteínas, nem sempre a resolução da imagem vai permitir distinguir entre os diferentes tipos de átomos em uma inspeção visual.

O "Vocabulário de Tipos de Átomos com Ciclos CA34567" resultou em 45 classes e por isso foram propostos dois mapeamentos simples removendo a informação de ciclos desse vocabulário. O primeiro mapeamento resultou no "Vocabulário de Tipos de Átomos" com 11 classes iguais aos tipos de átomos presentes no banco de dados mais o "background". O segundo mapeamento simplificou ainda mais o primeiro criando agrupamentos de tipos de átomos pouco frequentes e criou o "Vocabulário de Tipos de Átomos Agrupados" com

6 classes. As classes desses três vocabulários baseados no tipo de átomo e os respectivos mapeamentos são apresentados na Tabela 2.3.

Tabela 2.3: Vocabulário de tipos de átomos com ciclos C347CA56 e seus 2 mapeamentos.

Vocabulário de Tipos de Átomos com Ciclos CA34567	**Mapeamento 1 Vocabulário de Tipos de Átomos**	**Mapeamento 2 Vocabulário de Tipo de Átomos Agrupados**
NCA5	N	N
OC7	O	O
NC7	N	N
NC5	N	N
NCA6	N	N
Br	Br	Halo
CC7	C	C
SC7	S	PSe
CCA6	C	C
SC5	S	PSe
N	N	N
CC4	C	C
OCA7	O	O
CCA7	C	C
SeC5	Se	PSe
SC6	S	PSe
PC6	P	PSe
CCA4	C	C
CC3	C	C
C	C	C
SC4	S	PSe
O	O	O
OCA5	O	O
Cl	Cl	Halo
OCA6	O	O
CC6	C	C
CC5	C	C
SeCA5	Se	PSe
S	S	PSe
OC6	O	O
Se	Se	PSe
SCA6	S	PSe
SCA5	S	PSe
OC4	O	O
P	P	PSe
NC6	N	N
NC4	N	N
OC3	O	O
OC5	O	O
F	F	Halo
CCA5	C	C
PC5	P	PSe
NC3	N	N
I	I	Halo
background	background	background

Todos os 7 vocabulários resultantes (etapa 3 do workflow da Figura 2.1) dos mapeamentos propostos são apresentados na Tabela 2.4 com suas respectivas classes e contagens.

Esses vocabulários foram utilizados nos últimos treinamentos que apresentaram os melhores resultados.

Tabela 2.4: Propostas de rotulação da estrutura dos ligantes e suas respectivas classes

Vocabulário	Classes	Número de Classes
Classes SP	background, sp, sp2, sp3, sp3dx	5
Classes SP e Ciclos C347CA56	background, sp, sp2, sp3, sp3dx, C3, C4, C5, C6, C7, CA5, CA6	12
Região do Ligante	background, átomo	2
Átomos e Ciclos Genéricos	background, átomo, ciclo	3
Átomos Genéricos e Ciclos C347CA56	background, átomo, C5, CA5, C6, CA6, C3, C4, C7	9
Tipos de Átomos	background, C, O, N, P, S, F, I, Se, Cl, Br	11
Tipos de Átomos Agrupados	background, C, O, N, PSe, Halo	6

A rotulação automatizada é um processo muito propenso a erros, dado que variações no dado que desviam da estrutura esperada podem gerar comportamentos não desejáveis no algoritmo e criar ruídos de rotulação no conjunto de dados. Para aumentar a qualidade dos algoritmos de rotulação propostos, testes de caso de uso automatizados foram realizados com 8 estruturas de ligantes rotuladas manualmente. Os ligantes escolhidos possuem os seguintes códigos no PDB: 0YB, 1EJ, 58T, DJ4, I3C, MB5, MTE, Q0S. A escolha desses ligantes buscou cobrir praticamente todas as classes dos vocabulários propostos em diferentes arranjos químicos. Mais ligantes poderiam ser rotulados manualmente e ser automaticamente utilizados pelo *script* de testes, seguindo a padronização utilizada para testar estes 8 ligantes. A rotulação manual dos ligantes do conjunto de teste é uma tarefa tediosa e que demanda tempo, mas que envolve muito aprendizado e compreensão da metodologia interdisciplinar sendo proposta e das suas consequências no conjunto de dados criado. Além disso, testes baseados em engenharia reversa foram incluídos nos algoritmos de rotulação das imagens dos ligantes para aumentar a qualidade da rotulação. O SMILES dos ligantes foi utilizado como contra-prova para verificar a rotulação da estrutura do ligante a partir dos arquivos SDF e validar o conjunto de classes retornadas.

2.3.3 Criação das imagens 3D da densidade eletrônica residual dos ligantes

A imagem 3D da densidade residual de ligantes corresponde à nuvem eletrônica ao redor da estrutura do ligante (posição x, y, z dos seus átomos), evidenciada pelas regiões com intensidade acima de um certo contorno da densidade (Figura 2.3). Logo, criar uma imagem dos ligantes implica em obter o mapa 3D da densidade eletrônica residual, definir qual contorno será usado para se obter a nuvem eletrônica do ligante, em extraí-la do mapa da densidade eletrônica residual sem incluir ruídos adjacentes e em definir as características da densidade que serão incluídas na imagem (e.g. intensidade de cada ponto). E criar imagens para se aplicar uma arquitetura de aprendizado profundo envolve definir qual representação dos dados será utilizada, como atribuir nessa representação características presentes no dado experimental na forma de atributos e como rotular essas imagens.

A obtenção dos mapas 3D da densidade eletrônica residual de cada entrada do PDB presente na listagem final de ligantes foi realizada com o software Dimple [97], uma pipeline de cristalografia macromolecular para refinamento baseado nos programas da suíte CCP4

[110]. O refinamento com o Dimple foi executado para cada entrada do PDB utilizando seus respectivos arquivos .mtz e .pdb, com a opção de remoção de heteroátomos (remove os átomos de todos os ligante do arquivo .pdb) e com 2x de ciclos de refinamento (refinamento mais longo). As entradas do PDB que apresentaram erros no refinamento foram excluídas e aquelas que tiveram sucesso foram mantidas na listagem final de ligantes válidos. Com a opção de remoção de heteroátomos, a nuvem eletrônica dos ligantes pode ser encontrada nos mapas de densidade eletrônica residual criados.

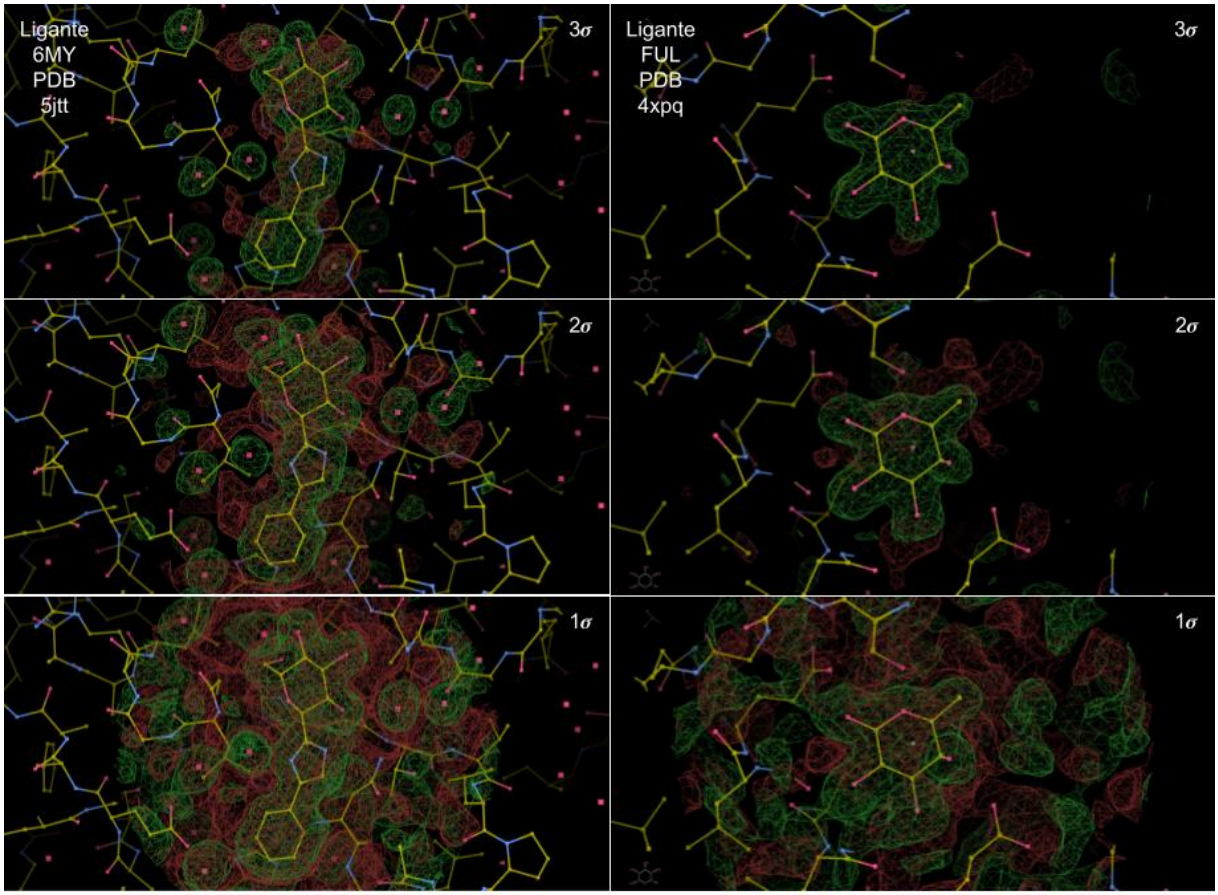

Figura 2.3: Imagens dos ligantes 6MY e FUL na densidade residual em três contornos: 1σ, 2σ e 3σ. Também são mostrados as estruturas das proteínas das duas entradas do PDB, 5JTT e 4XPQ, ambas com resolução de 1.85 Å. É possível perceber que, quanto menor o contorno σ, mais ruído é adicionado a imagem da densidade residual. Apesar da resolução das duas entradas ser a mesma, diferentes quantidades de ruído aparecem nas imagens dos dois ligantes nos mesmos níveis de contorno σ.

A representação da imagem afeta todas as questões que envolvem criar as imagens 3D dos ligantes e foi a primeira a ser definida. As representações de dados 3D existentes para se aplicar arquiteturas de aprendizado profundo podem ser divididas entre Dados Euclidianos Estruturados e Dados não Euclidianos [4], esquematizadas na Figura 2.4.

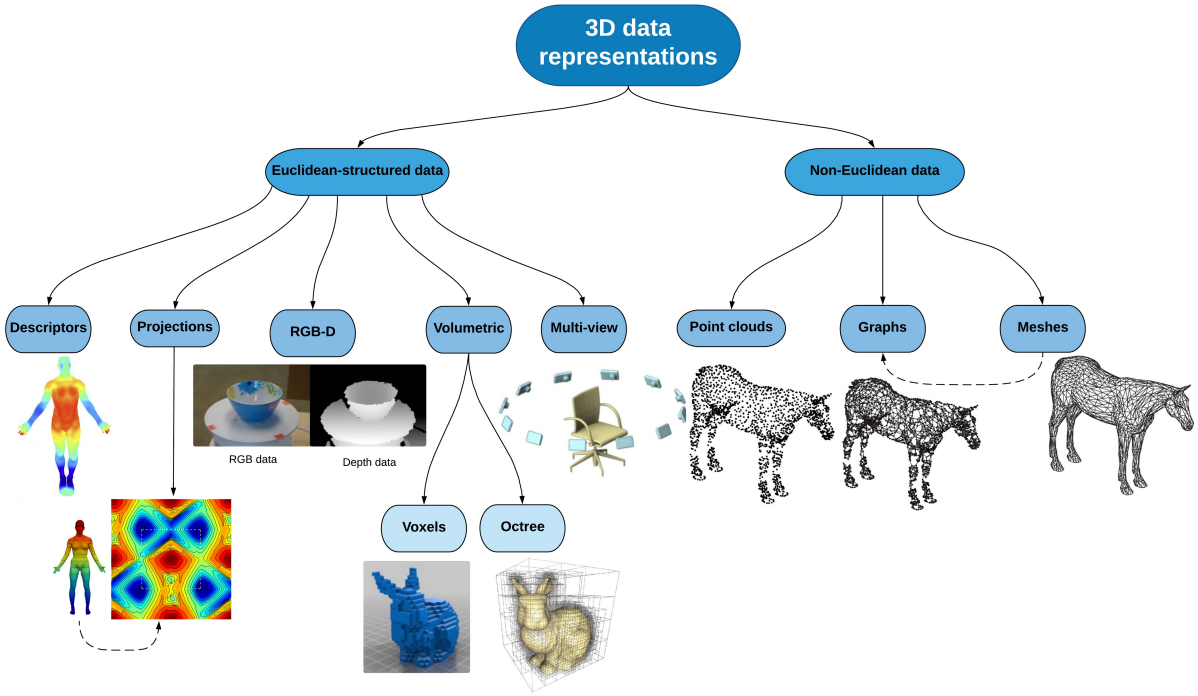

Figura 2.4: Várias representações para dados 3D: representações euclidianas (Descritores, Projeções, RGB-D, Volumétrica; voxels e octree e multi-view) e representações não-euclidianas (Nuvem de pontos, grafos e meshes). Fonte: [4].

Dados 3D Euclidianos Estruturados são aqueles que preservam as propriedades do dado estruturado em uma caixa rígida, que têm uma parametrização global e o mesmo sistema de coordenadas. Os principais tipos de dados 3D Euclidianos são descritores, projeções, RGB-D, volumétricas (voxels e octree) e multi-view. Os dados 3D não Euclidianos não necessariamente preservam essas propriedades e não seguem uma estrutura rígida, em alguns casos precisam de outras informações para se definir proximidade e vizinhança entre regiões. Os principais tipos de dados 3D não Euclidianos são nuvem de pontos, meshes 3D e grafos [4]. Nuvem de pontos e meshes 3D também podem ser considerados Euclidianos dependendo da escala do processamento (local ou global) e do sistema de coordenadas usado [4, 14].

Os ligantes de PN presentes na listagem final são moléculas orgânicas de diversos tamanhos, podendo ter de 1 (íons) a 140 átomos não hidrogênio, sendo a mediana igual a 6 átomos e a média igual a 11 átomos não hidrogênio. Algumas estruturas possuem cadeias longas e compridas enquanto outras ficam contidas em uma região menor. As diferentes entradas de ligantes do mesmo tipo podem aparecer em diferentes conformações 3D e podem ocupar diferentes espaços. Seria muito desafiador colocar todos os ligantes dentro de uma mesma caixa de tamanho fixo. Se a caixa fosse pequena, muitas entradas teriam que ser processadas utilizando uma caixa deslizante o que pode comprometer o aprendizado de padrões locais. E se a caixa fosse grande o suficiente para conter qualquer entrada, muitas entradas teriam uma quantidade enorme de espaço vazio e muita memória seria necessária para processá-las. Dadas essa variabilidade e pouca rigidez no tamanho e formato das entradas de ligantes, a facilidade e rapidez em se manipular nuvens de pontos e a existência de muitas arquiteturas de aprendizado profundo para nuvens de pontos 3D

[45], foi decidido representar as imagens 3D da densidade residual dos ligantes em nuvem de pontos.

A biblioteca Gemmi [111] do Python para biologia estrutural fornece o arcabouço de funções para manipular os mapas da densidade eletrônica em caixas 3D indexáveis, se comportando como vetores numéricos. A partir da interpolação trilinear dos 8 pontos mais próximos [1] da densidade eletrônica presentes no mapa residual é possível extrair caixas 3D de regiões específicas dos mapas da densidade eletrônica com diferentes espaçamentos entre os pontos utilizando funções do Gemmi. Os mapas da densidade eletrônica fornecem a característica de intensidade para cada ponto, e nesse primeiro momento foi escolhido utilizar apenas essa informação como atributo da imagem. Com isso é possível extrair uma imagem do mapa da densidade eletrônica residual em uma caixa cúbica e atribuir os valores de intensidade de cada ponto 3D da caixa a partir da interpolação dos valores do mapa.

A densidade eletrônica é usualmente visualizada utilizando a escala sigma, que permite distinguir características macromoleculares dentro de um mesmo conjunto de dados. A escala sigma é uma transformação linear apresentada nas Equações 2.1, 2.2 e 2.3 [104].

$$\rho_\sigma(x,y,z) = \frac{1}{\sigma_\rho}(\rho(x,y,z) - \overline{\rho}) \tag{2.1}$$

Com

$$\overline{\rho} = \frac{\sum_{N_{grid}} \rho(x,y,z)}{\sum_{N_{grid}} 1} = \frac{\sum_{N_{grid}} \rho(x,y,z)}{N_{grid}} \tag{2.2}$$

e

$$\sigma_\rho = \sqrt{\frac{1}{N_{grid}} \sum_{N_{grid}} (\rho(x,y,z) - \overline{\rho})^2} \tag{2.3}$$

onde $\rho(x,y,z)$ é o valor da densidade eletrônica nessa posição, N_{grid} é o número de pontos do grid na cela unitária, $\overline{\rho}$ é a média da densidade eletrônica na cela e σ_ρ é o seu desvio padrão. A escala sigma $\rho_\sigma(x,y,z)$ tem as propriedades da sua média ser 0 e seu desvio padrão ser igual a 1. Empiricamente, os cristalógrafos consideram valores de $\rho_\sigma(x,y,z) > 1$ como um 'nível de sinal' no qual os detalhes estruturais são analisados (valores notadamente acima do valor médio, ou seja, acima do valor para o ruído de fundo do solvente) e valores de $\rho_\sigma(x,y,z) > 3$ como um 'nível de sinal forte' [104].

Quando a escala sigma é utilizada em comparações visuais e numéricas ela pode levar a resultados errôneos [104]. Isso acontece porque a frequência de distribuição dos valores da densidade eletrônica de duas entradas distintas pode ser diferente e isso afeta a média e o desvio padrão das distribuições. Como consequência, o mesmo nível de contorno sigma ($p(x) > \sigma$) pode selecionar quantidades diferentes de pontos em mapas de entradas distintas, criando imagens diferentes em cada caso e pouco ou nada comparáveis [104]. Além disso, a escala sigma retorna valores em um intervalo aberto e portanto deveria ser truncada para ser melhor comparada e utilizada no treinamento de um modelo de aprendizado profundo. Uma alternativa que se mostrou viável foi a escala quantile rank, uma transformação não linear da densidade eletrônica baseada na posição do quantil do valor

de cada ponto do mapa ou de uma região delimitada. A escala quantile rank é apresentada nas Equações 2.4 e 2.5 e já foi utilizada em outras aplicações de cristalografia [104]. Em processamento de imagem essa operação é chamada de equalização de histograma [104, 48], a qual já normaliza os valores no intervalo de 0 a 1. Isso significa que para cada valor de corte μ se conta o número N_μ de pontos do grid tais que o valor da densidade eletrônica está abaixo dele, $\rho(x, y, z) < \mu$, e seu quantile rank η é calculado como mostra a Equação 2.4 [104].

$$\eta(\mu, \rho) = \frac{N_\mu}{N_{grid}}, \quad 0 \leq \eta(\mu, \rho) \leq 1. \tag{2.4}$$

$$Q(x, y, z) = \eta(\rho(x, y, z), \rho) \tag{2.5}$$

A escala quantile rank substitui o valor da densidade $\rho(x, y, z)$ em cada ponto pela sua posição na distribuição do quantil dado por $Q(x, y, z)$, como mostra a Equação 2.5. Essa escala não altera a forma da nuvem eletrônica, todos os pontos que têm o mesmo valor μ de intensidade possuem o mesmo valor nessa função. Além disso, diferentemente da escala sigma que deve ser aplicada globalmente em todo o mapa da densidade eletrônica, a escala quantile rank pode ser aplicada localmente dentro de uma caixa para comparação de uma mesma região. Isso permite agilizar os cálculos e excluir ruídos do mapa da densidade eletrônica de regiões distantes, uma vez que que a resolução do dado de cristalografia de proteínas varia localmente.

O trabalho de Adams et al. [104] também estimou uma correspondência entre contornos na escala sigma e na escala quantile rank. Eles observaram para diferentes entradas que contornos de 1 σ, 2 σ ou 3 σ correspondem a posições do quantil que variam aproximadamente entre 0.85, 0.95 e 0.98. Em cristalografia, normalmente é utilizado um contorno de 3σ para se inspecionar a densidade eletrônica residual de ligantes com pouco ruído. Contornos de 1 σ e 2 σ permitem visualizar mais informações presentes na nuvem eletrônica do ligante, mas também podem trazer mais ruído para a imagem (por exemplo, vindos de outras conformações da molécula, de moléculas próximas ou do solvente).

Foi decidido utilizar a escala quantile rank na criação da imagem dos ligantes. Os próximos passos que precisavam ser definidos eram: i) como extrair uma caixa ao redor da posição onde o ligante se encontra; e ii) quais contornos aplicar nessa caixa. Como não é possível saber antecipadamente o comportamento do modelo com imagens criadas de formas diferentes, se mostrou interessante testar diferentes contornos na criação das imagens dos ligantes. O esquema para obtenção da imagem dos ligantes para criação do banco de dados de imagens 3D de ligantes é mostrado na Figura 2.5 e cada etapa é descrita a seguir.

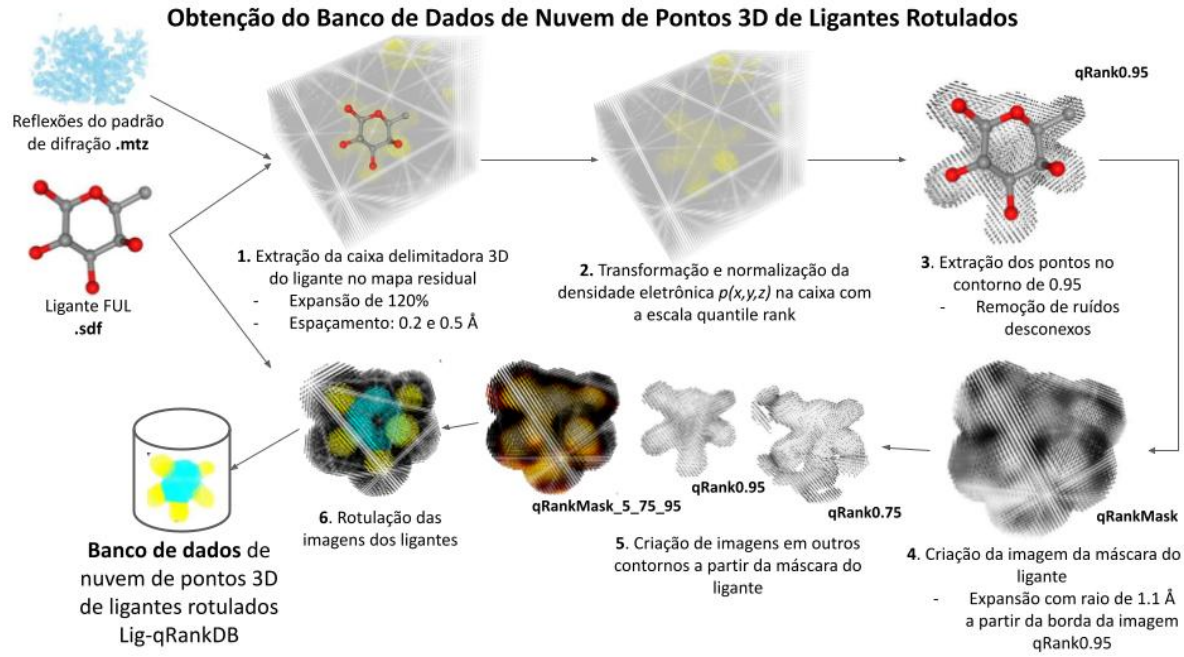

Figura 2.5: Esquema para obtenção do banco de dados de imagens 3D de densidade eletrônica residual de ligantes em nuvem de pontos 3D rotuladas a partir das suas estruturas químicas.

A partir das posições atômicas x, y, z da estrutura de um ligante depositado no PDB é possível obter uma caixa delimitadora na fronteira dessas posições que engloba toda a estrutura do ligante. A nuvem eletrônica dos ligantes ocupa uma região que contorna suas posições atômicas e que confere a forma da imagem do ligante. Para a caixa ao redor do ligante conter a forma completa da sua imagem na densidade eletrônica residual é necessário expandir a caixa delimitadora na fronteira das suas posições atômicas. Foi definido por inspeção experimental adicionar um *gap* de 2.1x2 Å, igual ao diâmetro do maior raio teórico (Tabela 2.5), e aplicar uma expansão de 120% nos limites dessa fronteira para se obter as dimensões da caixa delimitadora do ligante, centralizada na caixa delimitadora da fronteira do ligante. Dois espaçamentos para os pontos da caixa delimitadora do ligante foram escolhidos para serem testados, iguais a 0.2 e 0.5 Å, valores bem menores do que a distância de uma ligação química (uma ligação simples C-C mede em torno de 1.5 Å) e que permitem obter mais detalhes e precisão na predição do modelo.

Com as dimensões da caixa delimitadora do ligante é possível criar sua representação em nuvem de pontos e utilizar o pacote Gemmi para extrair os valores da densidade eletrônica em cada um dos pontos da caixa. A nuvem de pontos da caixa delimitadora do ligante recebe na cor de cada um de seus pontos o valor da interpolação trilinear da densidade eletrônica residual nessa posição. Em seguida, os valores da densidade da caixa delimitadora do ligante são transformados utilizando a escala quantile rank, a partir da ordenação dos valores da densidade dentro da caixa e substituição do valor de cada ponto pela sua posição na distribuição do quantil após ordenação, seguindo as Equações 2.5 e 2.4.

A partir da nuvem de pontos da caixa delimitadora do ligante foi possível inspecionar o resultado da aplicação de diferentes contornos da escala quantile rank nessa caixa para

obtenção das imagens do ligante. Foi possível perceber que contornos entre 0.85 e 0.98 criam imagens finas ao redor da estrutura do ligante, quanto mais próximo de 0.98 mais fragmentada pode ser a imagem e quanto mais próximo de 0.85 mais ruído de fundo pode ser incluído na imagem. Contornos próximos ou abaixo de 0.85 acabavam trazendo muito ruído para imagem final, que em alguns casos poderia se estender por toda caixa delimitadora por meio de pontes e caldas conectando ruídos. Esse ruído acaba por fazer com que a imagem final do ligante fique muito grande e necessite de muita memória disponível para processar mais do que uma imagem ao mesmo tempo.

A alternativa encontrada para padronizar a criação da imagem do ligante com diferentes contornos sem que muito ruído fosse incluído foi criar uma máscara da imagem do ligante expandida a partir de um contorno mais fino da nuvem de pontos da caixa delimitadora. A partir dessa máscara, se aplicou diferentes contornos dentro dela para obtenção das imagens finais do ligante. Foi definido criar a máscara a partir de uma imagem do ligante a um contorno de 0.95 na escala quantile rank e expandi-la a partir dos pontos da sua borda com um raio igual a 1.1 Å. O valor desse raio é igual a 65% do maior raio experimental dos átomos considerados para a resolução de 2.2 Å. Essa máscara é então extraída da nuvem de pontos da caixa delimitadora do ligante a partir da seleção dos pontos cobertos após expansão da imagem a um contorno de 0.95 e é chamada de imagem em nuvem de pontos da máscara do ligante. O pacote Open3D [116] do Python foi utilizado para criar a nuvem de pontos da caixa delimitadora e da máscara do ligante. Esse pacote possui uma implementação de KDTrees utilizando a biblioteca FLANN [76] para rápido acesso da vizinhança mais próxima dos pontos da nuvem que permite fazer buscas com ótimo desempenho e facilidade. O tempo médio de criação da caixa delimitadora do ligante em nuvem de pontos 3D foi de 0.33 segundos por ligante e o tempo médio de criação da imagem da máscara do ligante e das imagens em outros contornos foi de 0.39 segundos por ligante (tempo médio para um espaçamento dos pontos igual a 0.5 Å e o mesmo hardware utilizado nos resultados).

As imagens finais dos ligantes (etapa 4 do workflow da Figura 2.1) foram obtidas aplicando-se diferentes contornos na nuvem de pontos da máscara do ligante, chamada de "qRankMask". Foram criadas imagens dos ligantes com contornos em 0.5, 0.75, 0.85, 0.95 na escala quantile rank, chamadas de "qRank0.5", "qRank0.75" e assim por diante. A Figura 2.6 mostra as imagens de dois ligantes em três contornos diferentes, desde a caixa delimitadora, e a nuvem de pontos da máscara dos ligantes, e ilustra o impacto do valor do contorno na imagem final dos ligantes.

Outra imagem dos ligantes com 3 contornos diferentes em cada canal de cor também foi criada inspirada no trabalho de Pereira et al. [86]. Esse trabalho avaliou com sucesso uma abordagem nova de criação de imagens cerebrais para segmentação 2D de imagens de ressonância magnética (MRI) com CNN para diagnóstico da doença de Alzheimer. A metodologia usual desse problema de segmentação é utilizar a imagem da fatia central da MRI 3D para treinar o modelo de CNN. Dessa forma, só é apresentado ao modelo uma única imagem 2D de uma imagem 3D bem maior, a profundidade e a dimensão total do cérebro podem ser perdidas. Pereira et al. propuseram uma metodologia de colocar em cada canal de cor da imagem 2D fatias diferentes da MRI, permitindo assim que o modelo pudesse acessar mais informação sobre a profundidade 3D da imagem sem ser necessário

ter o custo de treinar um modelo de CNN 3D. De forma semelhante, os contornos na imagem da densidade eletrônica permitem acessar uma 4D da imagem que é a intensidade da densidade eletrônica em cada ponto. Porém, quando apenas um contorno é utilizado não é possível saber a profundidade desta quarta dimensão. Quando a imagem da nuvem de pontos da máscara do ligante é criada com 3 contornos diferentes em cada canal de cor, o modelo passa a conseguir acessar a profundidade da intensidade da densidade eletrônica residual nessa região e isso poderia auxiliar no seu desempenho. Uma imagem da máscara dos ligantes foi criada com os contornos 0.5, 0.75 e 0.95 na escala quantile rank em cada canal de cor da imagem, chamada de "qRankMask_5_75_95". Os pontos que não aparecem no respectivo contorno recebem o valor 0.0.

A Figura 2.6 mostra a imagem "qRankMask_5_75_95" para os mesmos 2 ligantes, 6MY e FUL. Nesta imagem os pontos que aparecem em todos os contornos ficam com uma coloração em tons de cinza e branco, os pontos que aparecem apenas nos contornos 0.75 e 0.5 ficam com a coloração em tons de amarelo, os pontos que aparecem apenas no contorno de 0.5 ficam com a coloração em tons de vermelho e os pontos da máscara que não aparecem em nenhum contorno ficam em preto.

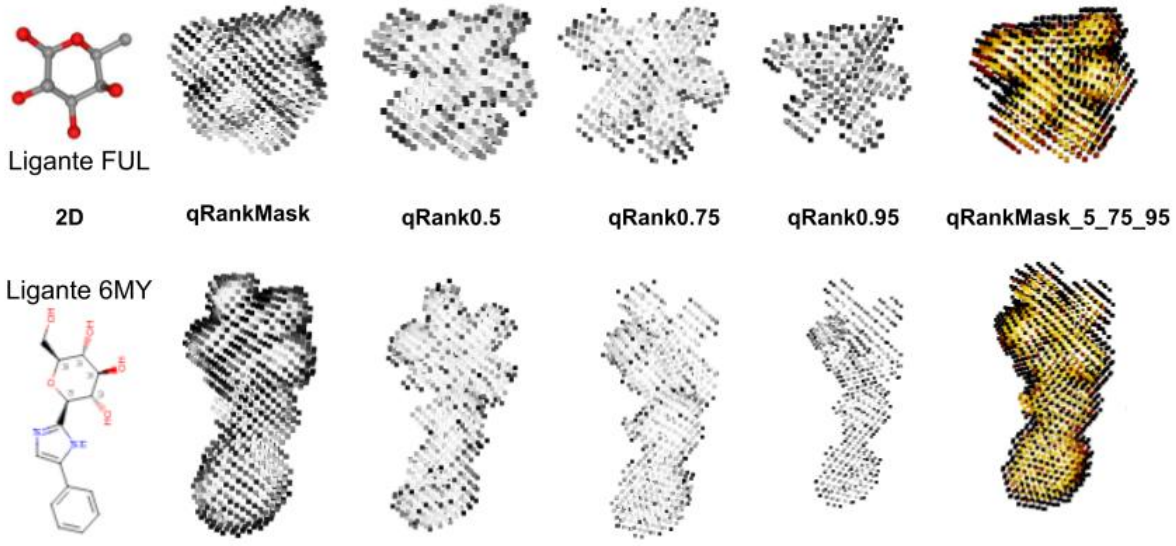

Figura 2.6: Imagens da densidade eletrônica de dois ligantes na escala quantile rank em nuvem de pontos 3D. São mostradas as imagens qRankMask, qRank0.5, qRank0.75, qRank0.95 e qRankMask_5_75_95 para os ligantes FUL e 6MY. A imagem da máscara do ligante (qRankMask) é utilizada para extrair as imagens nos contornos 0.5, 0.75 e 0.95 na escala quantile rank. A partir dos pontos das imagens que aparecem nesses três contornos é criada a imagem qRankMask_5_75_95.

Durante uma inspeção visual das imagens dos ligantes, percebeu-se que algumas entradas apresentavam grande quantidade de ruído de fundo devido a outras conformações da molécula. Esse ruído acabava quase duplicando o tamanho final da imagem, o que poderia comprometer o desempenho do modelo e limitar a quantidade de imagens que poderiam ser lidas ao mesmo tempo devido a limites de memória. Em um momento inicial desse projeto, foi implementado um filtro de remoção de pontos muito distantes dos átomos dos ligantes com o objetivo de remover esse ruído de outras conformações da imagem final dos

ligantes. Uma distância igual ao raio de expansão da máscara de 1.1 Å da posição dos átomos do ligante foi definida para fazer essa remoção de pontos ruidosos da imagem dos ligantes. Os primeiros treinamentos foram feitos com imagens criadas utilizando esse filtro de remoção de pontos distantes, mas foi constatado que essa metodologia não poderia ser reprodutível para dados novos de ligantes desconhecidos, para os quais não se sabe suas posições atômicas e nos quais o modelo final seria aplicado. Por esse motivo, esse filtro deixou de ser utilizado na criação das imagens dos ligantes para os últimos treinamentos e ficou a cargo do modelo aprender a remover os pontos de ruído de outras conformações. Sem o uso do filtro, menos imagens puderam ser lidas ao mesmo tempo devido a limites de memória. Esses limites serão apresentados na seção de resultados de treinamento do modelo de aprendizado profundo para segmentação da imagem 3D da densidade residual de ligantes.

Outras formas de criar a imagem dos ligantes podem ser testadas, esse projeto de mestrado fornece um arcabouço inicial de funções que podem ser modificadas e servirem de exemplo para aplicação de outras metodologias nessa etapa. Com a metodologia implementada foi possível criar um banco de dados de imagens de densidade residual dos ligantes com 4 tipos de imagens para cada ligante válido da listagem final, são eles: qRank0.5, qRank0.75, qRank0.85, qRank0.95 e qRankMask_5_75_95. Imagens em outros contornos também foram criadas, mas esses 4 tipos foram os utilizados nos treinamentos finais.

2.3.4 Rotulação das Imagens 3D dos Ligantes - Extrapolação da Rotulação da Estrutura dos Ligantes

A última etapa para obtenção do banco de dados de imagens 3D da densidade residual de ligantes rotuladas é a extrapolação da rotulação da estrutura do ligante para a sua imagem em nuvem de pontos. Essa extrapolação deve ser feita para atender os requisitos do modelo de segmentação semântica, que espera uma rotulação para cada ponto da imagem. Além disso, essa extrapolação tem que estar relacionada à natureza dos dados para ser possível capturar os padrões presentes na imagem. Uma modelagem muito utilizada para calcular o volume atômico de moléculas é tratar os átomos como esferas rígidas [58, 44]. Essas esferas possuem o raio igual ao raio atômico de van der Waals para cada tipo de átomo e servem de modelo para representar o volume da densidade que seria ocupado pela molécula. O trabalho de Batsanov de 2001 [9] sumariza os dados disponíveis sobre o raio atômico teórico de van der Waals em moléculas e cristais.

Foi decidido utilizar a modelagem de uma esfera atômica para extrapolar a rotulação dos átomos dos ligantes para a imagem da sua nuvem eletrônica, usando como raio os resultados de Batsanov para cada tipo de átomo (Tabela 2.5). O algoritmo para rotulação da imagem de nuvem de pontos dos ligantes atribui a rotulação de cada átomo para os pontos em uma esfera ao seu redor que estão a uma distância menor ou igual ao raio atômico teórico definido para o respectivo átomo. Os pontos que ficam na região de intersecção de duas ou mais esferas atômicas recebem a rotulação do átomo mais próximo. Esse procedimento foi implementado com as funcionalidades da biblioteca Open3D para rápido acesso da vizinhança de cada ponto e inicialmente utilizou os raios atômicos tabelados por Batsanov.

As imagens dos ligantes rotuladas nessa modelagem acabaram com rotulações que cobriam toda a imagem sem que a forma das subestruturas representadas pelas classes do vocabulário utilizado fossem mantidas. O que aconteceu é que a densidade eletrônica visível em contornos mais finos corresponde à região central do pico de intensidade de cada átomo, no modelo de esferas atômicas é como se apenas o centro da esfera fosse visível. Isso acontece pois a intensidade da densidade eletrônica se distribui teoricamente como uma gaussiana centralizada em cada átomo [44] e quando um contorno é aplicado na densidade eletrônica, apenas o pico central da gaussiana é visível [51]. A partir dessa constatação apenas 65% do raio teórico passou a ser utilizado para extrapolar a rotulação dos átomos dos ligantes para sua imagem da densidade residual em nuvem de pontos. Outras porcentagens do raio atômico não foram testadas.

Em setembro de 2020 foi publicado o trabalho do XGen [51] para encaixe de ligantes no espaço real de mapas de densidade eletrônica. Esse trabalho forneceu uma tabela com o raio experimental de raio X típico para elementos orgânicos em diferentes resoluções da densidade eletrônica. Os raios experimentais fornecidos por aquele trabalho passaram a ser utilizados neste projeto substituindo o uso dos raios atômicos teóricos de van der Waals (Tabela 2.5), mantendo-se o uso de 65% do tamanho do raio. A resolução da entrada do PDB passou a ser utilizada para selecionar o conjuntos de raios a serem usados, arredondando a resolução para a primeira casa decimal (valores tabelados para as resoluções 1.5, 1.6, 1.7, 1.8, 1.9, 2.0, 2.1 e 2.2 Å). Para os átomos de Boro (B) e Selênio (Se), que não aparecem na tabela do XGen, foram atribuídos os raios dos átomos de Enxofre (S) e de Bromo (Br), respectivamente. O trabalho do XGen também mostrou que a modelagem da densidade eletrônica baseada em uma função gaussiana esférica proposta por eles é bem semelhante a transformada de Fourier padrão. Isso justifica o uso de 65% do raio atômico experimental para rotulação, pois dessa forma apenas o "pico da gaussiana" da densidade eletrônica residual de cada átomo é rotulado na imagem dos ligantes.

Tabela 2.5: Raio atômico teórico e experimental de átomos orgânicos.

Átomo	**Raio Teórico de Van der Waals [9]**	**Raio Experimental XGen (1.5 Å) [51]**	**Raio Experimental XGen (2.2 Å) [51]**
B	1.8	1.34	1.68
C	1.7	1.40	1.72
N	1.6	1.38	1.70
P	1.95	1.33	1.68
O	1.55	1.36	1.68
S	1.8	1.34	1.68
Se	1.9	1.30	1.66
Cl	1.8	1.34	1.68
F	1.5	1.34	1.67
Br	1.9	1.30	1.66
I	2.1	1.30	1.65

Os pontos da imagem dos ligantes que não são cobertos pela esfera atômica com 65% do raio experimental do XGen recebem a rotulação de ruído de fundo ("background" do solvente). Aplicando o procedimento de rotulação nas imagens em nuvem de pontos dos ligantes válidos da listagem final, foi obtido, para cada vocabulário testado, um banco de dados de imagens 3D da densidade residual de ligantes rotuladas (etapa 5 do workflow da Figura 2.1).

Os bancos de dados de imagens 3D dos ligantes em nuvem de pontos serão chamados de $Lig - qRankDB - x\ - y$, onde x corresponde ao vocabulário utilizado sendo $x = SP$ ou $Atomo$ e y corresponde a faixa de resolução das entradas dos ligantes. Esse banco de dados é organizado em subpastas, onde cada subpasta corresponde a uma entrada do PDB e contém todas as imagens dos ligantes que aparecem na respectiva entrada.

2.4 Treinamento de modelos de aprendizado profundo para segmentação da nuvem de pontos 3D de ligantes

A segunda etapa deste projeto de mestrado foi o treinamento de modelos de aprendizado profundo para segmentação da nuvem de pontos 3D de ligantes (etapa 6 do workflow da Figura 2.1). Essa etapa consistiu nos seguintes passos:

1. Criação de um conjunto de treinamento a partir da listagem final de ligantes válidos
2. Escolha de uma arquitetura de aprendizado profundo
3. Criação de redes de aprendizado profundo utilizando a arquitetura escolhida

4. Criação de uma pipeline de treinamento

Criar um conjunto de treinamento da listagem final de ligantes engloba escolher filtros de qualidade para remover entradas possivelmente ruidosas ou erradas (i.e. a estrutura depositada não está de acordo com a densidade eletrônica residual), ajustar o balanceamento entre as classes do modelo nesse conjunto e separar o conjunto em entradas para treino, validação e teste do modelo (validação cruzada) respeitando uma distribuição equitativa de características dos ligantes entre cada partição. A escolha da arquitetura envolve entender qual opção pode ser melhor para o tipo de dado e o problema sendo resolvido. Criar as redes de aprendizado profundo envolve definir as características de estrutura da rede, como quantidade, ordem e tipo de camadas. Criar uma pipeline de treinamento envolve definir o workflow de treinamento e as parametrizações a serem testadas, como tamanho do *batch*, otimizador, função de perda e métrica de avaliação. O workflow de treinamento deve ter funções para ler e escrever os dados no formato esperado, permitir executar treinamentos seguindo critérios parametrizados e salvar a progressão do treino. Cada etapa será descrita em mais detalhes nas seções seguintes.

2.4.1 Criação de um conjunto de treinamento a partir da listagem final de ligantes válidos

Muitos estudos avaliaram o impacto de ruídos de rotulação em modelos de aprendizado profundo. Apesar desses modelos poderem ser robustos à presença de ruídos, muitos estudos ainda precisam ser feitos para se entender melhor suas consequências em conjuntos de dados diversos e evitar um aprendizado degradado [7, 6, 56]. A listagem final de ligantes válidos contém todas as entradas de ligantes que podem ser incluídas no conjunto de treinamento. Como os dados do PDB são fornecidos pela comunidade científica, eles podem conter erros de depósito devido a uma interpretação errada da densidade eletrônica observada [31]. Por isso, mostrou-se importante definir filtros de qualidade para as entradas dos ligantes com objetivo de remover ruído do conjunto de dados de treinamento. Esses filtros deveriam utilizar informações já presentes nos dados, pois a avaliação do encaixe dos ligantes não seria feita neste momento. Criaram-se variáveis a partir das informações presentes na listagem dos ligantes que compõem dois conjuntos de filtros de qualidade chamados de globais e locais, para serem utilizados na remoção de entradas de ligantes potencialmente ruidosas.

Os filtros de qualidade globais incluídos foram a razão entre o B fator do ligante e da proteína, o desvio padrão do B fator do ligante e a ocupância mínima do ligante. A razão entre o B fator do ligante e da proteína é uma métrica já reportada para se avaliar a qualidade de depósitos de ligantes quanto à conformidade entre as moléculas do cristal sendo que valores menores do que 2 são desejados [101, 37, 33]. Idealmente essa comparação deveria ser feita localmente, entre o ligantes e os átomos da proteína que estão ao seu redor [63]. Para facilitar os cálculos, essa comparação foi feita usando o B fator médio da proteína. O desvio padrão do B fator dos átomos do ligante é uma métrica com o mesmo intuito de avaliar a conformidade entre os átomos da molécula depositada [63] sendo que valores abaixo de 10 são desejados [101, 33]. A ocupância mínima dos

átomos do ligante também é uma métrica utilizada para averiguar a correspondência entre a estrutura e a densidade eletrônica [101, 33, 63] sendo que valores acima de 0.9 foram escolhidos. Esses filtros estão definidos como globais pois utilizam valores vindos do refinamento da entrada completa do PDB, e logo, relacionam cada ligante com a entrada do PDB onde ele foi depositado.

Os filtros de qualidade locais incluídos foram o tamanho da nuvem de pontos dos ligantes na imagem desejada e a porcentagem de pontos que seriam cobertos pela esfera atômica de cada átomo que estão de fato presentes nas imagens do ligante. Esses filtros relacionam atributos locais de cada entrada com o que se espera encontrar na imagem dos ligantes. O tamanho da nuvem de pontos dos ligantes é igual ao número de pontos de cada entrada e é desejado que todas as entradas tenham uma quantidade de pontos suficiente para capturar a imagem completa do ligante com o espaçamento utilizado. Nuvens de pontos muito pequenas podem indicar densidades eletrônicas residuais fragmentadas, e logo, baixa correspondência entre a imagem e a estrutura do ligante depositado. A porcentagem de pontos cobertos nas imagens dos ligantes também auxilia em averiguar essa correspondência.

No treinamento de um modelo de aprendizado de máquina, sabe-se que todo erro que entra no modelo pode induzir a saídas com erro na predição ("garbage in, garbage out"). Existem muitos tipos de ruído devido a uma rotulação errada, podendo ser classificados como ruído de rotulação uniforme, dependente da classe, concentrado localmente e dependente das features [6]. Conseguir identificar o tipo de erro de rotulação é bastante difícil, mas alguns dos seus impactos no treinamento são característicos e auxiliam pensar em alternativas para o problema [56, 6]. Os filtros de qualidade foram pensados na tentativa de remover entradas vindas de depósitos errados e que poderiam adicionar mais confusão ao modelo. Por outro lado, remover entradas diminui a diversidade e o tamanho do conjunto de treinamento, requisitos essenciais para se obter um bom desempenho em modelos de aprendizado de máquina. Além disso, estudos já mostraram que modelos de aprendizado profundo podem ser robustos a ruídos de rotulação quando o conjunto de treinamento é grande o suficiente [94], mas as avaliações existentes não cobrem diferentes tipos de dados e de ruídos. De toda forma, verificar o comportamento do modelo sem aplicar nenhum filtro de qualidade no conjunto de treinamento também se mostrou algo importante de avaliar. Na próxima seção, será apresentada uma função de perda utilizada para auxiliar o treinamento na presença de ruído, a qual demonstrou bons resultados.

O balanceamento da distribuição das classes do vocabulário utilizado no conjunto de treinamento é um ponto crítico para o desempenho geral de modelos de aprendizado de máquina. Quando existe um desbalanceamento entre as classes, ou seja, classes que aparecem muitas vezes em muitas entradas e outras classes que aparecem poucas vezes em poucas entradas, o modelo fica enviesado a aprender as classes mais frequentes e não consegue inferir as classes menos frequentes. Esse problema acontece porque a atualização do modelo fica dominada pela classe mais frequente [55]. O desbalanço pode ser intrínseco ao dado, quando a frequência das classes é consequência da natureza do dado (i.e. subestruturas químicas pouco frequentes em moléculas orgânicas), ou extrínseco ao dado, que se caracteriza por ter sido introduzido por fatores externos (i.e. depósitos do PDB com muita repetição de um mesmo ligante). Também é importante saber a quantidade

de entradas em que aparecem as classes pouco frequentes e se essa quantidade é suficiente para permitir treinar o modelo. Caso as classes minoritárias sejam raras ou pouco representadas é mais provável que o desempenho do modelo fique comprometido [55]. O problema de desbalanço entre as classes ainda apresenta muitas limitações quanto a sua avaliação para conjuntos de dados diferentes, e soluções para conjuntos muito grandes (*big data*) com classes raras [55, 15, 95]. À medida que o aprendizado de máquina profundo é cada vez mais aplicado em novos dados, esse problema se mostra mais relevante de ser estudado e novas soluções se mostram necessárias.

Para avaliar o conjunto de dados dos ligantes na faixa de resolução de 1.5 à 2.2 Å quanto ao balanceamento das classes foi necessário observar a distribuição de ocorrência das classes por átomo rotulado dos ligantes para todos os vocabulários sendo propostos. Para isso todos os ligantes da listagem final de ligantes válidos foram rotulados com os vocabulários propostos. As distribuições para os vocabulários mapeados a partir das classes SP são apresentadas na Figura 2.7. E para os vocabulários mapeados a partir das classes de tipos de átomos as distribuições são apresentadas na Figura 2.8 Além disso, para cada distribuição foi calculada a razão de desbalanceamento máximo entre as classes dada pela Equação 2.6 [55].

$$d_{max} = \frac{\max_i C_i}{\min_i C_i} \tag{2.6}$$

onde C_i é o conjunto de átomos rotulados com a classe i, e $\max_i C_i$ e $\min_i C_i$ são a máxima e a mínima ocorrência de átomos rotulados entre todas as classes i, respectivamente.

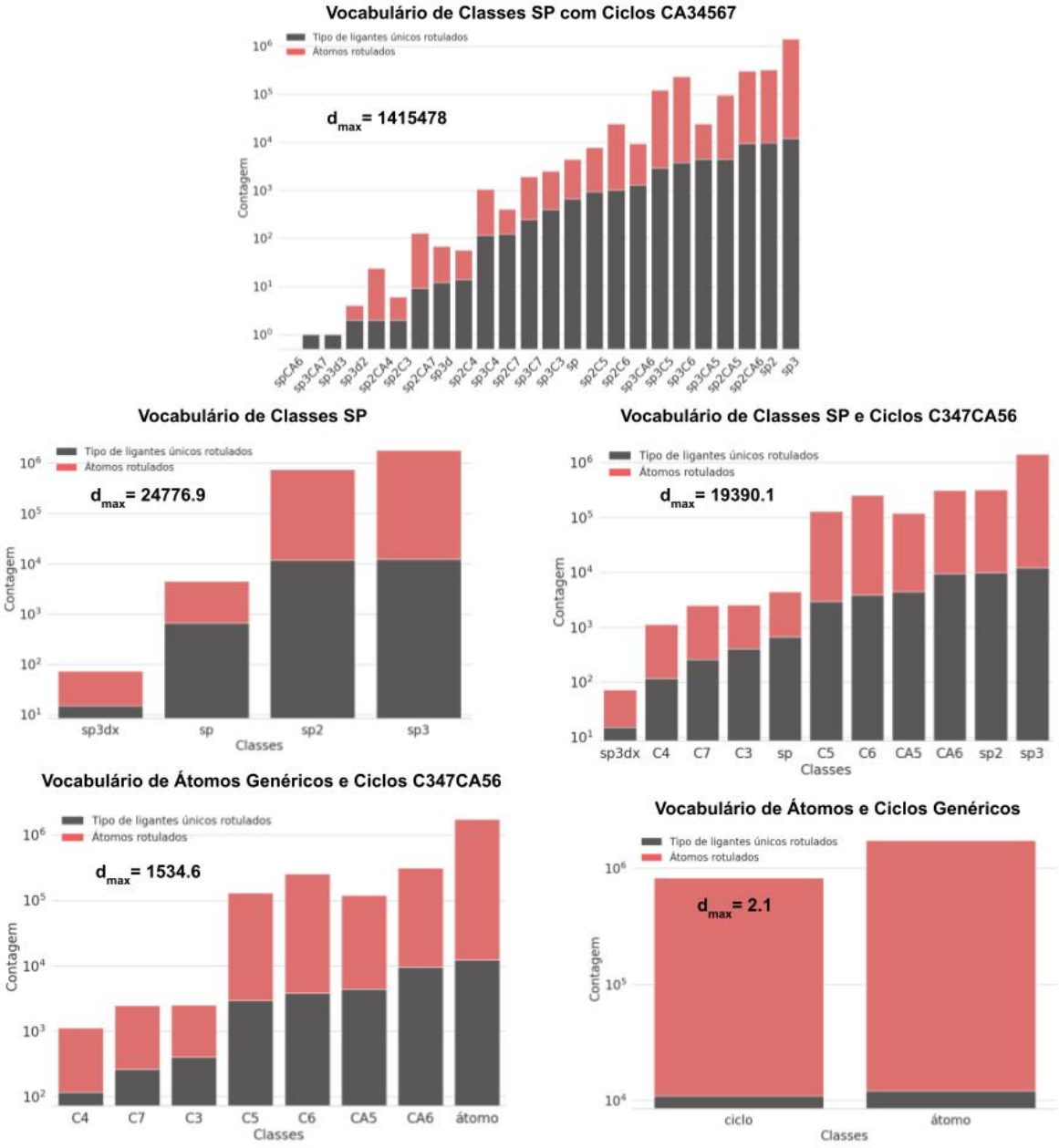

Figura 2.7: Distribuição de ocorrência por átomo rotulado das classes dos vocabulários baseados no SP em todas as entradas de ligantes disponíveis na faixa de resolução de 1.5 a 2.2 Å.

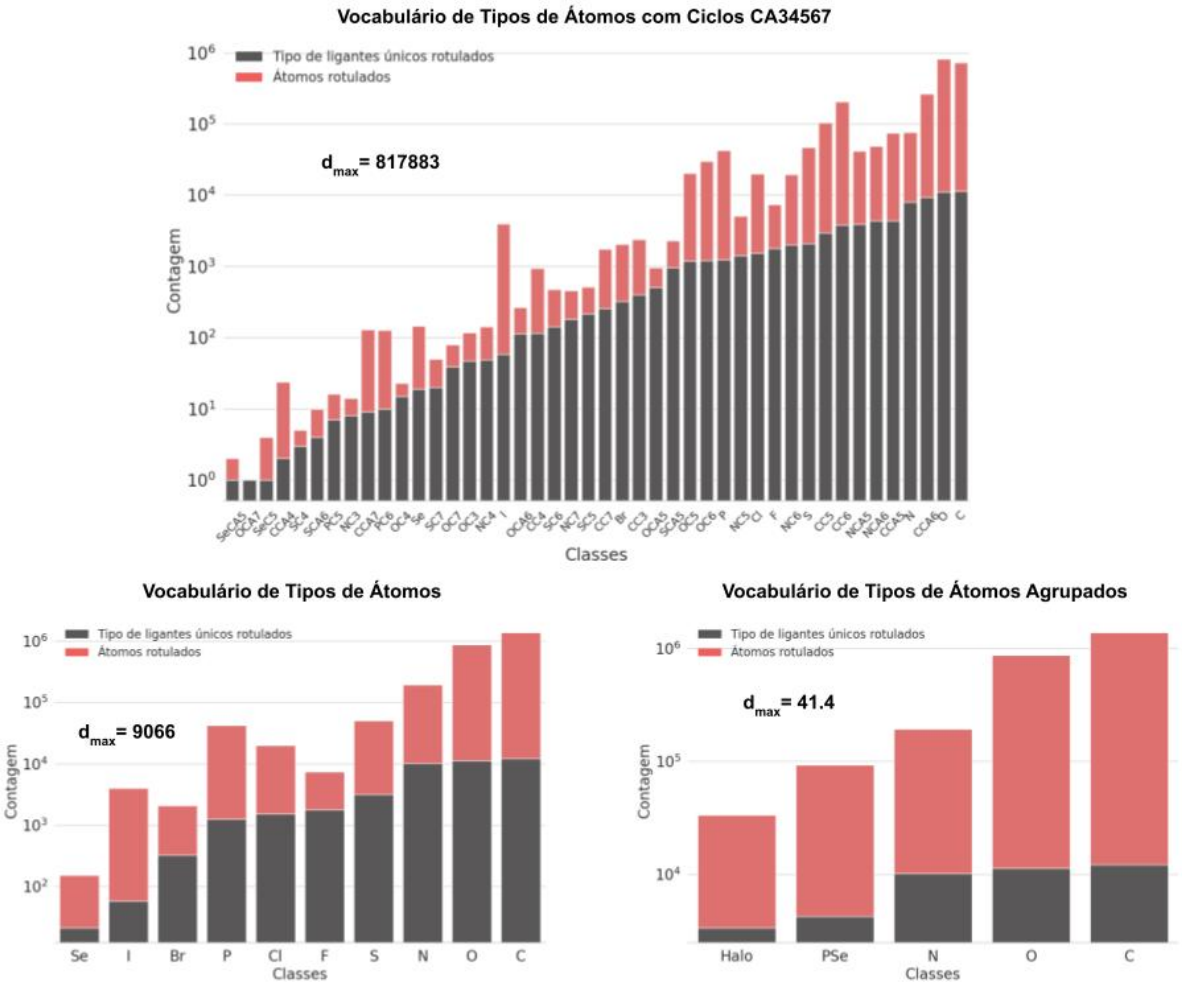

Figura 2.8: Distribuição de ocorrência por átomo rotulado das classes dos vocabulários baseados no tipo de átomo em todas as entradas de ligantes disponíveis na faixa de resolução de 1.5 a 2.2 Å.

As Figuras 2.7 e 2.8 exemplificam o problema de desbalanço entre as classes presente no conjunto de dados dos ligantes e alerta para a necessidade de alternativas para essa questão. Esse desbalanceamento tem origens tanto intrínsecas como extrínsecas ao dado, o que significa que mesmo com mais depósitos variados no PDB (extrínseco ao dado) o desbalanceamento ainda existirá para as classes que são minoritárias dentro do espaço químico de moléculas orgânicas (intrínseco ao dado). Portanto, pensar em alternativas para o desbalanço é essencial para se trabalhar com esses dados. Além disso, perceber o desbalanço é muito importante para entender as limitações que o modelo pode apresentar e escolher quais propostas testar.

Os métodos de aprendizado de máquina profundo para trabalhar com conjuntos de dados desbalanceados podem ser divididos em três categorias [55, 15]: ao nível dos dados, ao nível do algoritmo e híbridos. Os métodos ao nível dos dados alteram o conjunto de dados ou sua amostragem para mudar a distribuição das classes e diminuir o desbalanceamento. Os métodos ao nível do algoritmo não alteram o conjunto de dados e ajustam os algoritmos de treinamento e decisão para aumentar a importância das classes minoritárias. E os métodos híbridos combinam métodos das duas outras categorias para lidar

com o desbalanceamento.

Os principais métodos ao nível dos dados são o *oversampling* e o *undersampling*. O *oversampling* consiste na repetição de entradas com as classes menos frequentes até que a distribuição fique mais equilibrada. O *oversampling* se mostra inviável quando a classe minoritária é muito pequena, o que implica na criação de muitas entradas duplicadas. Isso pode fazer o modelo ficar muito difícil de ser treinado em tempo razoável ou causar *overfitting* devido à alta taxa de duplicação de entradas [55]. Uma técnica mais avançada que se mostrou muito promissora quanto ao *overfitting* foi o SMOTE (do inglês *Synthetic Minority Over-sampling Technique*), um método que produz entradas minoritárias artificiais a partir da interpolação de entradas reais muito próximas (parecidas) [55, 18, 15]. Porém, criar entradas sintéticas para os ligantes é um problema difícil, pois envolve criar moléculas com a estereoquímica viável e com uma imagem da densidade eletrônica residual que seja consistente com os dados experimentais reais. Se as entradas sintéticas não respeitarem essas restrições elas podem atrapalhar o desempenho do modelo para as entradas reais ou simplesmente serem duplicações de entradas reais. O *undersampling* consiste no descarte de entradas com as classes mais frequentes do conjunto de dados. Um lado negativo desse método é que entradas reais que poderiam auxiliar no desempenho do modelo são removidas do treinamento. Por outro lado, o *undersampling* pode ser utilizado para remover entradas ruidosas ou com características muito frequentes que poderiam enviesar o modelo.

No caso dos ligantes, além do desbalanceamento entre as classes também existe um desbalanço entre os tipos de ligantes presentes no conjunto de dados (mostrado na Tabela 2.1). Como é importante que exista no conjunto de treinamento uma diversidade de estruturas e arranjos para que o modelo consiga predizer uma entrada nova em diferentes conformações e estruturas, utilizar o *undersampling* para resolver os dois desbalanços se mostrou interessante. O *oversampling* foi descartado nesse momento devido à alta taxa de desbalanceamento no conjunto dos ligantes e tamanhos reduzidos das classes minoritárias, o que implicaria em muitas duplicações de entradas. Além disso, a dificuldade em se criar entradas sintéticas da densidade eletrônica residual descartou a tentativa de uso de técnicas baseadas no SMOTE neste trabalho.

No caso de modelos de segmentação semântica, a remoção de entradas por classe se torna uma questão mais complexa pois uma mesma entrada pode conter mais do que uma classe nos seus pontos. Dessa forma, uma entrada pode conter tanto classes muito frequentes quanto classes pouco frequentes. Com essa situação foi necessário pensar em formas de fazer o *undersampling* sem perder as entradas com as classes menos frequentes e sem incluir muitas repetições de um mesmo tipo de ligante.

A implementação do *undersampling* para o conjunto de dados dos ligantes é apresentada no Algoritmo 5. O Algoritmo 5 considera a ocorrência máxima por tipo de ligante e um intervalo de ocorrência mínima e máxima para cada classe que vai controlar a inclusão de exemplos no conjunto de treinamento. A ocorrência mínima de cada classe pode levar à exclusão de uma determinada classe do conjunto de dados caso não existam entradas suficientes para cobri-la. A ocorrência máxima pode levar à exclusão de muitas entradas até que o limite superior seja respeitado ou não exista mais entradas que possam ser removidas, sem alterar as classes menos frequentes cujas entradas são fixadas. A ocor-

rência máxima por tipo de ligante foi implementada com a remoção de entradas de tipos de ligante muito frequentes utilizando um algoritmo de anti-clusterização [83]. A anti-clusterização divide as entradas em grupos semelhantes segundo características definidas e reforça a diversidade dentro dos grupos. Dessa forma é possível remover entradas de um mesmo grupo de forma controlada para manter uma diversidade em relação a outras características dessas entradas de um mesmo tipo de ligante, como resolução e razão do B fator. O Algoritmo 5 também realiza um *undersampling* para remoção de ruído, todos os ligantes com menos do que 5 átomos diferentes de hidrogênio são removidos (estruturas muito simples) e todos os ligantes com a imagem qRank0.95 com menos do que 150 pontos são removidos (corte definido por inspeção visual dos dados abaixo desse valor).

Os principais métodos ao nível do algoritmo para trabalhar com conjuntos de dados desbalanceados consideram uma penalidade ou peso nas classes na função de perda, ou modificam o limite de decisão das classes durante o treinamento para reduzir o viés para as classes mais frequentes. Esses métodos atuam na função de perda do treinamento e podem ser mais simples de serem testados quando não se deseja otimizar os pesos ou limites automaticamente durante o treinamento. A dificuldade se mostra em definir a melhor matriz de pesos ou limites para as classes, dado o problema sendo tratado.

Utilizar uma combinação dos métodos existentes é uma alternativa muito recomendada e permite se beneficiar de seus pontos positivos em conjunto. Neste projeto, foram aplicados tanto o método de *undersampling* separado, como sua combinação com diferentes pesos na função de perda para ajustar o desbalanço remanescente. Uma função de perda mais robusta a ruídos de rotulação também foi testada. A aplicação dos métodos ao nível do algoritmo serão melhor descritos na próxima seção referente ao treinamento do modelo de segmentação semântica.

A última etapa para criação de um conjunto de treinamento é a reamostragem dos dados para validação cruzada (VC). A VC é uma técnica para avaliar a capacidade de generalização de um modelo preditivo e para evitar o *overfitting* [12]. Em aprendizado de máquina, a generalização se refere à capacidade de um algoritmo de ser eficaz em diferentes entradas. Um modelo com *overfitting* é caracterizado por estar adaptado e enviesado aos dados de treino, e logo, tem baixa capacidade de generalização para novos dados. Existem muitas técnicas de VC, todas elas compartilham o fato de particionar o conjunto de treinamento em dados para treino, teste e validação [12, 70]. Dessa forma, os dados para avaliar o modelo durante o treinamento (validação) e após o treinamento (teste) são diferentes daqueles usados para treinar o modelo (treino). Com esse particionamento é possível avaliar o desempenho do modelo em generalizar para dados novos. O método de VC mais usual e direto é o chamado *hold-out*, que consiste na separação aleatória do conjunto de treinamento em dados de treino e teste. Geralmente, os dados de teste contém entre 10% e 30% dos dados. Outro método de VC muito utilizado é o *k-fold cross-validation*, que consiste na aplicação do método "hold-out" k vezes, mas a reamostragem é feita de tal forma que nenhum conjunto de teste se sobrepõe. No *k-fold cross-validation*, o conjunto de treinamento é separado em k subconjuntos disjuntos de tamanho aproximadamente igual e um deles é utilizado para teste e os restantes $k-1$ para treino. A média do desempenho do modelo treinado em cada conjunto k é o desempenho da VC [12] (ilustrado na Figura 2.9). Usualmente, um valor de k em torno de 10

Algorithm 5 Algoritmo de undersampling para redução do desbalanço e remoção de ruído

procedure UNDERSAMPLINGLIGANTES(*listaLigantes*, *minAtomos*, *maxLigCode*, *minClassOcc*, *maxClassOcc*, *vocabulario*) ▷ A listagem de entradas de ligantes é fornecida no parâmetro `listaLigantes` e o algoritmo remove as entradas que não obedecem os filtros de número mínimo de átomos (`minAtomos`)), máximo número de ligantes por tipo (`maxLigCode`) e a quantidade de átomos por classe do vocabulário entre `minClassOcc` e `maxClassOcc`. Todos os ligantes com a imagem qRank0.95 com menos do que 150 pontos são removidos.
    `listaLigantes_filtrada` ← `listaLigantes` com as entradas com menos átomos do que `minAtomos` removidas
    `listaLigantes_filtrada` ← `listaLigantes_filtrada` com as entradas com tamanho da imagem qRank0.95 do ligante menor do que 150 pontos removidas
    `tiposLigante` ← lista de todos os tipos de ligantes presentes em `listaLigantes`
    `tiposLigante_unicos` ← lista dos tipos de ligantes únicos presentes em `tiposLigante`
    for `ligCode` in `tiposLigante_unicos` **do**
        if contagem de `ligCode` in `tiposLigante` > `maxLigCode` **then**
            `listaLigantes_filtrada` ← `listaLigantes_filtrada` com apenas `maxLigCode` entradas com o tipo de ligante igual a `ligCode` usando anti-clusterização ▷ Anti-clusterização baseada nas seguintes características das entradas: B fator, ocupância mínima do ligante, resolução e tamanho da imagem
    `listaLigantes_filtrada` ← `listaLigantes_filtrada` ordenada pelo tipo de ligante e o B fator
    `vocabulario` ← `vocabulario` ordenado em ordem crescente de ocorrência das classes
    for `vocab_classe` in `vocabulario` **do**
        if contagem de átomos da classe `vocab_classe` em `listaLigantes_filtrada` < `minClassOcc` **then**
            `listaLigantes_filtrada` ← `listaLigantes_filtrada` com todas as entradas em que aparece a classe `vocab_classe` removidas
        else
            if contagem de átomos da classe `vocab_classe` nas entradas de `listaLigantes_filtrada` > `maxClassOcc` **then**
                `listaLigantes_filtrada` ← `listaLigantes_filtrada` com as entradas em que aparecem a classe `vocab_classe` removidas aleatoriamente sem remover as entradas com as classes abaixo de `maxClassOcc` ▷ Remove as entradas com as classes majoritárias sem afetar as classes minoritárias aleatoriamente em relação a ordenação
    return `listaLigantes_filtrada` ▷ A listagem de entradas de ligantes filtrada após undersampling

é aplicado.

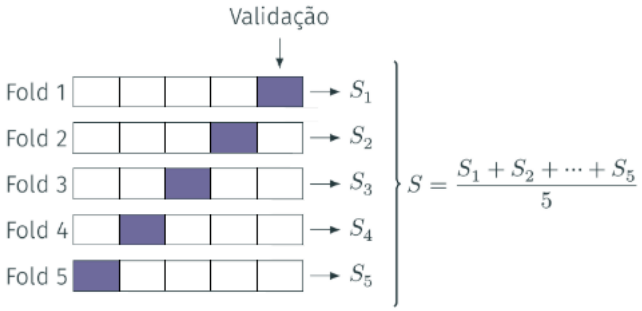

Figura 2.9: Esquema da validação cruzada *k-fold*. Cada retângulo indica um conjunto. O retângulo preenchido indica o conjunto que está sendo utilizado para validação ou teste do modelo. Os demais são utilizado para treino. Acurácia do modelo será a média da acurácia dos treinamentos com cada valor de k. Fonte: `https://medium.com/turing-talks/turing-talks-10-introdu%C3%A7%C3%A3o-%C3%A0-predi%C3%A7%C3%A3o-a75cd61c268d`

O método de *k-fold cross-validation* foi aplicado para reamostrar o conjunto de treinamento dos ligantes e fazer a VC do modelo de aprendizado profundo. Para melhor avaliar a generalização do modelo preditivo, é interessante se manter uma semelhança entre esses k subconjuntos para que cada um deles seja ao máximo possível representativo do todo, e logo, com muita diversidade de entradas. Normalmente essa separação é feita de forma aleatória, mas isso não garante que uma diversidade de características de entradas seja coberta em cada subconjunto. Para garantir a diversidade nos subconjuntos, foi realizada uma separação estratificada utilizando o algoritmo de anti-clusterização para particionar o conjunto de treinamento em k grupos semelhantes, mantendo dentro de cada grupo uma diversidade em relação às características selecionadas [83]. As características das entradas escolhidas para a estratificação foram: razão do B-fator, resolução, tipo de ligante, número de átomos por classe do vocabulário e tamanho da máscara da imagem do ligante. Além disso, cada grupo k foi separado em 2 grupos referentes ao conjunto de teste e de validação utilizando o algoritmo de anti-clusterização com as mesmas características selecionadas. Por questão do tempo para o treinamento de um modelo de aprendizado profundo, em alguns treinamentos apenas um subconjunto k foi selecionado para teste e validação e o método de VC aplicado foi o *hold-out*. O procedimento implementado para aplicar o método *k-fold cross-validation* estratificado em um conjunto de treinamento dos ligantes utilizando o algoritmo de anti-clusterização baseado nas características das entradas é descrito no Algoritmo 6.

A criação de um conjunto de treinamento para os ligantes segue os seguintes passos. Primeiro os filtros de qualidade globais e locais desejados são aplicados na listagem final de ligantes válidos, em seguida o Algoritmo 5 de *undersampling* é aplicado para diminuir o desbalanço do conjunto de treinamento e finalmente a listagem de ligantes resultante é particionada em k subconjuntos estratificados para validação cruzada, utilizando o Algoritmo 6.

Algorithm 6 Algoritmo de partição do conjunto de treinamento em "k-folds" estratificados utilizando anti-clusterização

1: **procedure** KFOLDANTICLUST($listaLigantes, kfold, vocabulario$) ▷ Implementação do método *k-fold cross-validation*. Particiona o conjunto de treinamento em $kfold$ grupos semelhantes; cada grupo k é separado em teste e validação mantendo a diversidade das entradas.

2: $lista\_caracteristicas\_ligantes$ ←{B fator, resolução, ocupância mínima do ligante, tipo de ligante, entrada do PDB, tamanho da imagem qRank0.95, ocorrência por átomos das classes do `vocabulario`}

3: $kfold\_ligantes$ ← particionamento em k grupos com objetivo de diversidade utilizando o algoritmo anti-clusterização baseado nas características das entradas dos ligantes de `listaLigantes` presente na lista `lista_caracteristicas_ligantes` ▷ Retorna uma lista com o índice do grupo de cada entrada variando de 1 até k

4: $test\_val\_k\_ligantes$ ← {'teste' repetido para cada entrada de ligante presente em `listaLigantes` }

5: **for** k in 1 to $kfold$ **do**

6: $test\_val\_k$ ← particionamento das entradas de `listaLigantes` com $kfold\_ligantes == k$ em 2 grupos semelhantes utilizando o algoritmo de anti-clusterização baseado nas características presentes em `lista_caracteristicas_ligantes`

7: $test\_val\_k\_ligantes[kfold\_ligantes == k \ \& \ test\_val\_k == 2]$ ← 'val' ▷ Adiciona ao conjunto de validação todos os ligantes do grupo k atribuídos ao segundo grupo em $test\_val\_k$

8: **return** $kfold\_ligantes, test\_val\_k\_ligantes$ ▷ A lista de k agrupamentos das entradas presentes em `listaLigantes` e a lista de separação em teste e validação igualmente repartida para cada grupo

2.4.2 Arquitetura da Rede de Aprendizado Profundo

A escolha da arquitetura de aprendizado profundo é uma tarefa difícil quando não se tem nenhum trabalho anterior sobre o problema e os dados. Idealmente, diferentes arquiteturas deveriam ser testadas para se avaliar qual apresenta o melhor desempenho para os dados sendo utilizados. As imagens dos ligantes do conjunto de treinamento estão representadas em nuvens de pontos 3D, muitas delas esparsas devido a diferentes conformações e com tamanhos variados. Características locais da dispersão dos pontos precisam ser entendidas pelo modelo para se capturar os padrões geométricos das classes propostas e características um pouco mais distantes auxiliam em capturar o formato completo do ligante.

A arquitetura pioneira em aplicar CNN diretamente em entradas de nuvem de pontos foi a PointNet [89]. Essa arquitetura foi baseada em perceptron multicamadas (MLP) para modelar cada ponto independentemente e apresentou resultados competitivos, mas não foi capaz de aproveitar ao máximo a estrutura local dos pontos devido à agregação global das *features* de todos os pontos [45]. Foi apenas com a sua sucessora, a PointNet++ [90], que padrões mais finos de regiões locais das imagens passaram a ser capturados, mas ainda se apresentava como uma arquitetura muito complicada e com grande tempo computacional [4]. A partir dessas soluções outras arquiteturas de redes baseadas em pontos começaram a emergir, com abordagens propostas variadas como kd-trees, tensores esparsos e grafos [4, 45]. Em geral, os métodos dessas abordagens podem ser divididos em métodos de MLP ponto-a-ponto, métodos de convolução de ponto, métodos baseados em RNN e métodos baseados em grafos [45]. Os métodos de MLP ponto-a-ponto usam vários MLPs para modelar cada ponto e agregam as *features* globalmente, logo, não conseguem capturar geometrias locais em nuvem de pontos e interações mútuas entre pontos sem o uso de métodos adicionais. Os métodos de convolução de ponto propõem operações de convolução eficazes para nuvem de pontos, conseguindo assim capturar informações locais e de vizinhança. Os métodos baseados em RNN utilizam redes neurais recorrentes (RNNs) para capturar recursos de contexto inerentes de nuvens de pontos. Já os métodos baseados em grafos definem métodos de conectividade dos pontos para aplicar convoluções baseadas na vizinhança dos pontos e capturar formas e estruturas geométricas subjacentes de nuvens de pontos 3D.

As arquiteturas de aprendizado profundo para nuvem de pontos 3D baseadas em convolução de ponto e em grafos se mostraram mais promissoras para os dados de densidade eletrônica dos ligantes por possibilitarem capturar melhor as formas geométricas locais e informação de vizinhança. Além disso, a arquitetura a ser escolhida deveria ter código aberto, um bom desempenho computacional para processar imagens grandes de ligantes e necessitar de pouca adequação do dado para ser aplicada. Com auxílio de *benchmarks* algumas arquiteturas puderam ser comparadas e a *Minkowski Engine* [20], que tinha a melhor performance na ScanNet [29] em 2020, foi selecionada para ser testada. A rede DGCNN, baseada em grafos dinâmicos também foi selecionada para ser testada, mas devido ao seu alto custo computacional e à restrição de tamanho fixo das imagens (necessitaria uma reamostragem dos pontos das imagens) acabou inviabilizando seu uso.

A arquitetura *Minkowski Engine* adota tensores esparsos e propõe uma convolução esparsa generalizada em 3D que engloba todas as convoluções discretas. A ordem da con-

volução em tensores esparsos não é sequencial. Para calcular eficientemente a convolução em um tensor esparso é necessário um mapeamento do mapa do *kernel*, que define como cada elemento diferente de zero em um tensor esparso de entrada é mapeado para o tensor esparso de saída. Sua funcionalidade foi disponibilizada em uma biblioteca de diferenciação automática para tensores esparsos (Figura 2.10) que fornece muitas funções de código aberto para redes neurais convolucionais de alta dimensão [20]. Seu foco é em dados espacialmente esparsos com técnicas de compressão de modelo para alta performance e baixo custo de memória.

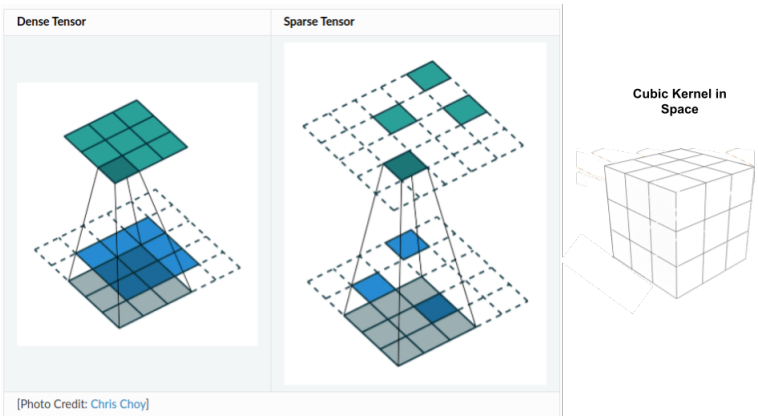

Figura 2.10: Visualização de uma convolução de imagem 2D simples em um tensor denso e em um tensor esparso (usado pela arquitetura *Minkowski Engine*) utilizando um *kernel* 3x3. À direita, um kernel cúbico 3x3x3 utilizado na arquitetura *Minkowski Engine*. Fonte das imagens adaptadas: [20].

2.4.3 Estrutura da Rede de Aprendizado Profundo

A arquitetura *Minkowski Engine* fornece algumas redes neurais de exemplo e a rede utilizada para segmentação semântica no *benchmark* da ScanNet é apresentada na Figura 2.11, chamada de rede "MinkUNet34C".

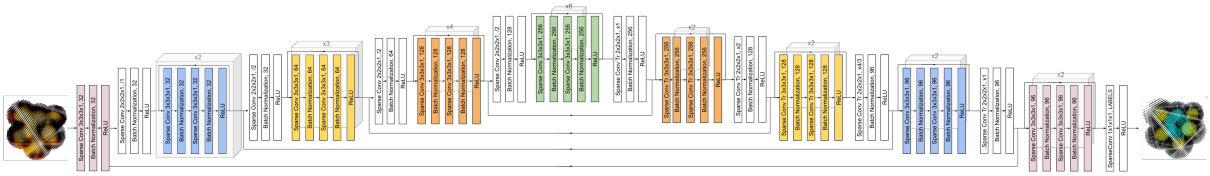

Figura 2.11: Estrutura da rede MinkUNet34C com 34 camadas. Esta rede possui 8 blocos com repetições iguais a 2, 3, 4, 6, 2, 2, 2 e 2, totalizando 23 camadas de blocos. Os blocos estão representados com um cubo ao fundo e na parte superior do cubo está a sua quantidade de repetições. O 'x' nas camadas de convolução indica um *kernel* cúbico.

A rede MinkUNet34C é baseada na rede 3D U-Net [118] com comunicação entre blocos de camadas intermediárias. Essa rede foi modificada neste trabalho para aceitar uma camada de normalização baseada na instância. Devido a restrições de memória é interessante aplicar a técnica de acúmulo de gradiente para simular um *batch* grande equivalente a leitura de muitas entradas ao mesmo tempo. Essa técnica faz o acúmulo de gradiente

do modelo a partir de iterações com *batchs* pequenos vindos da leitura de poucas entradas sequencialmente, até que o tamanho total do *batch* desejado seja alcançado. Quando se utiliza a técnica de acúmulo de gradiente, é importante utilizar a normalização baseada na instância ou em grupos para evitar uma normalização errada baseada no tamanho do *batch* de uma iteração ao invés do tamanho total do *batch*. A rede criada com a normalização baseada na instância foi chamada de "MinkUNet34CIN".

Outra modificação implementada nessa rede foi o uso da convolução dilatada ou convolução atrous [114, 35] 3D. A convolução dilatada permite expandir exponencialmente o campo receptivo da rede sem perda de resolução da imagem ou aumento de parâmetros na rede, e apresentou melhorias de desempenho relevantes [114]. A convolução dilatada funciona por meio da inserção de espaços vazios (zeros) entre cada voxel do *kernel* de convolução. A taxa de dilatação é controlada por um hiperparâmetro adicional d, onde $d-1$ espaços são inseridos entres os elementos do *kernel* de tal modo que $d = 1$ corresponde à convolução regular. A convolução dilatada é uma alternativa para capturar informações locais distantes sem o uso de muitas camadas de *poolings*, e logo, sem reduzir a resolução da imagem.

Um problema denominado *gridding* foi reportado para o uso de convoluções dilatadas sequências com a mesma taxa de dilatação [107]. O que acontece nessa convolução é que a introdução de zeros no *kernel* de convolução faz com que posições da imagem sejam puladas e não contribuam para o cálculo da convolução, e logo, suas informações são perdidas pela rede. O trabalho de Cottrell e Wang [107] propõe uma solução simples para esse problema denominada convolução dilatada híbrida. Ao invés de usar a mesma taxa de dilatação em todas as camadas de convolução dilatada, eles propuseram atribuir a taxa de dilatação seguindo um modo de onda dente de serra. Dessa forma, camadas sequenciais são agrupadas para formar a subida da onda com uma taxa de dilatação crescente, e o mesmo se repete para o próximo grupo de camadas. Ao fazer isso, a camada do topo consegue acessar a informação vinda de muitas posições da imagem, na mesma região da configuração original [107]. Esse problema e sua solução são ilustrados na Figura 2.12 com um exemplo em 2D.

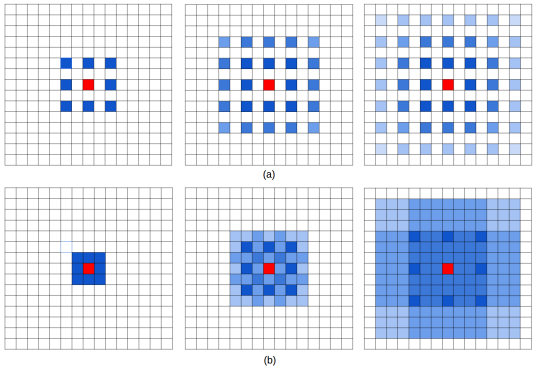

Figura 2.12: Ilustração do problema *gridding*. Da esquerda para a direita: os pixels marcados em azul contribuem para o calculo do pixel central em vermelho por meio de três camadas de convolução com um *kernel* de tamanho 3x3. (a) todas as camadas de convolução têm uma taxa de dilatação $d = 2$. O problema do *gridding* fica evidenciado pelos espaços em branco na vizinhança do pixel central. (b) camadas de convolução sequenciais têm taxas de dilação crescente iguais a $d = 1$, $d = 2$ e $d = 3$, respectivamente. Dessa forma, o problema do *gridding* pode ser contornado. Fonte: [107]

A rede construída com o uso de convolução dilatada híbrida foi chamada de "MinkUNet34C_CONVATROUS_HYBRID" e é apresentada na Figura 2.13, com as taxas de dilatação de cada camada definidas pela variável d. A versão desta rede com normalização baseada na instância é chamada de "MinkUNet34CIN_CONVATROUS_HYBRID".

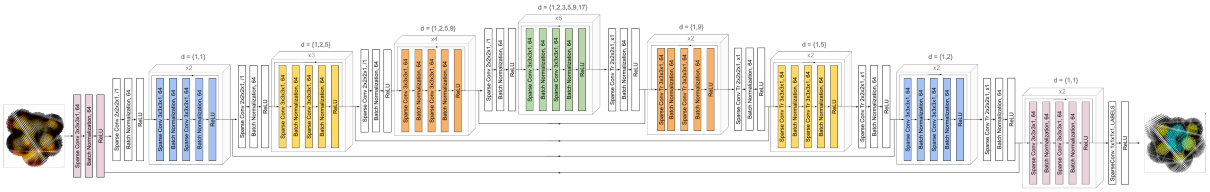

Figura 2.13: Estrutura da rede MinkUNet34C_CONVATROUS_HYBRID, semelhante a rede MinkUNet34C mas com diferentes taxas de dilatação em cada bloco e sem pooling. A taxa de dilatação está definida pela variável d colocada acima de cada bloco, os valores correspondem a dilatação de cada repetição do bloco.

2.4.4 Pipeline de Treinamento

O código para treinamento das redes neurais convolucionais da arquitetura Minkowski Engine utilizadas neste projeto foi modificado inicialmente da solução 4D-SpatioTemporal ConvNets [20]. Esse código foi adaptado para aceitar multi-GPU usando inicialmente a biblioteca DDP, "DistributedDataParallel" do Pytorch [68]. A pipeline de treinamento implementada era funcional, mas não tinha um código robusto, o que dificultava novas modificações e seu entendimento. Para simplificar o código e aumentar a garantia de qualidade na pipeline de treinamento, a biblioteca pytorch-lightning [41] foi utilizada para reformular o código e simplificar a implementação de novas modificações. Junto com a pipeline de treinamento, também foi implementada a rotina para controlar o acesso ao conjunto de dados dos ligantes e a leitura das imagens de entrada respeitando o agrupamento da validação cruzada.

A pipeline de treinamento implementada executa o treino do modelo até que a quantidade máxima de épocas ou de iterações seja atingida, seguindo os parâmetros fornecidos pelo usuário. Uma época em aprendizado de máquina significa passar o conjunto de treino completo pela rede, ou seja, corresponde a todas as iterações necessárias para ler o conjunto de dados de treino inteiro. A cada iteração uma quantidade de entradas igual ao tamanho total do *batch* é lida e enviada para a rede. O tamanho total do *batch* é igual ao tamanho do *batch* de treino multiplicado pelo número de GPUs (caso o treinamento seja feito em GPU) e pelo número de acúmulos de gradiente. A validação do modelo é executada ao final de cada época e dependendo dos parâmetros também pode ser executada depois de uma quantidade fixa de iterações. O teste do modelo é executado no final do treinamento com o último modelo obtido. A quantidade máxima de épocas é um hiperparâmetro muito importante pois controla quantas vezes o conjunto de treino será passado para a rede durante o processo de aprendizado, o que se relaciona diretamente com a convergência do modelo.

Devido a limites de memória do hardware utilizado, mostrou-se necessário o uso da técnica de acúmulo de gradiente para simular *batchs* maiores e possibilitar que mais classes fossem cobertas a cada iteração do treinamento e atualização do modelo. Junto com o uso do acúmulo de gradiente, as camadas de normalização da rede passaram a ser baseadas na instância. O tamanho total do *batch* é dado pelo número de GPUs multiplicado pelo tamanho do *batch* e pela quantidade de acúmulos de gradiente.

A métrica de avaliação utilizada na pipeline de treinamento desse projeto é a interseção sobre a união (IoU) [77], do inglês *Intersection over Union*, ilustrada na Figura 2.14. Essa métrica é adequada para avaliar o desempenho de modelos com desbalanceamento entre as classes, em casos em que a métrica de acurácia geral fica dominada pela classe majoritária (no caso da imagem da máscara do ligante a classe dominante é o *background*). A métrica de acurácia geral não é discriminativa, seu cálculo é igual ao total de pontos de acerto de todas as classes sobre o total de pontos da imagem, enquanto a IoU utiliza a intersecção dos pontos de acerto (predito igual ao esperado) de uma dada classe sobre a união dos pontos de acerto e erro (predito mais esperado) dessa classe, logo, avalia cada classe independentemente. A média da IoU (mIoU) de todas as classes é utilizada como métrica global do treinamento, como é feito em *benchmarks* de conjuntos de dados populares para segmentação semântica como ScanNet [29] e COCO [69]. A visualização da taxa de falso-positivos e falso-negativos por classe utilizando a métrica IoU é feita através de uma matriz de confusão, onde as linhas representam as classes esperadas e as colunas representam as classes preditas (exemplificada na Tabela 2.6). Essa tabela permite verificar a confusão entre as classes, ajuda a entender os erros que estão ocorrendo e os impactos de classes minoritárias ou difíceis de convergir [57]. A diagonal principal da matriz de confusão contém a IoU de cada classe. Neste projeto os valores da matriz de confusão são normalizado pela classe esperada (por linha), onde o total por classe é a soma das suas linhas e colunas (o total da classe Positivo da Tabela 2.6 para normalização da primeira linha é igual a $TP + FP + FN$).

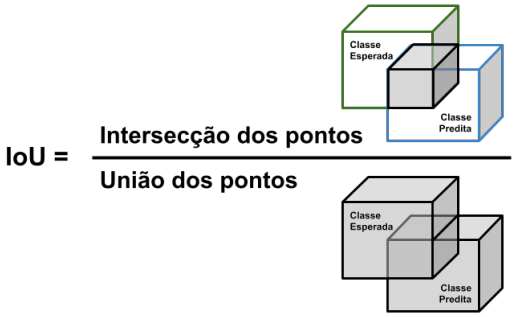

Figura 2.14: Ilustração da métrica de avaliação chamada interseção sobre a união (IoU). O cálculo da IoU é igual à intersecção entre os pontos preditos e os esperados (pontos de acerto) dividido pela união dos pontos preditos e esperados (pontos de acerto mais pontos de erro).

Tabela 2.6: Formato da matriz de confusão

		Classe Predita	
		Positivo	**Negativo**
Classe Esperada	**Positivo**	TP Verdadeiro Positivo	FN Falso Negativo
	Negativo	FP Falso Positivo	TP Verdadeiro Negativo

A função de perda utilizada para atualizar os pesos do modelo durante o treinamento foi inicialmente a Cross Entropy (CE) [113], já implementada na pipeline do trabalho 4D-SpatioTemporal ConvNets. Sua versão com pesos nas classes, chamada weighted Cross Entropy (wCE), também foi utilizada na tentativa de lidar com o desbalanço entre as classes. Outra função de perda utilizada após análise do comportamento da curva de aprendizado nos treinamentos realizados, que serão apresentadas na seção de resultados, foi a Symmetric cross entropy Learning (SL) [109]. A função de perda SL melhora a CE simetricamente, somando a esta uma contraparte robusta a ruídos na rotulação, a Reverse Cross Entropy (RCE) proposta neste trabalho [109]. A SL auxilia tanto no degradamento do aprendizado para classes difíceis de convergir quanto no *overfitting* da CE na presença de ruídos na rotulação. Ela foi avaliada em diferentes conjuntos de dados na presença de ruídos sintéticos e demonstrou bons resultados. A função de perda SL com pesos (wSL) também foi utilizada na tentativa de melhor lidar com o problema do desbalanço entre as classes.

Para permitir treinamentos mais longos sem *overfitting* e melhorar o aprendizado de classes raras e de classes difíceis de convergir, foi implementada durante o treinamento a rotação aleatória a uma taxa de $R\%$ das amostras de treino, onde R é fornecido como parâmetro. Essa técnica se aproxima de um *oversampling* em tempo real, com o benefício de adicionar diversidade ao treinamento.

Os hiperparâmetros de treinamento adicionais implementados na pipeline são a quantidade de acúmulo de gradiente, o número de GPUs (para treinamento com multi-GPU),

a função de perda com parâmetros e a taxa de rotação aleatória. Os parâmetros de treinamento em relação às imagens dos ligantes são o tipo da imagem (qRank0.75, qRankMask, etc) e o espaçamento da imagem (0.5 Å foi o melhor utilizado).

2.4.5 Hardware Utilizado

Grande parte dos experimentos e dos treinamentos feitos neste trabalho foram executados em um computador com as seguintes configurações: 1 CPU AMD Ryzen 9 3950X de 16 cores e 32 threads, 128Gb RAM e 2 GPUs GeForce RTX 2080 SUPER com 8Gb de RAM dedicada cada. Essa configuração permitiu utilizar um tamanho de *batch* igual a 8 imagens da máscara do ligante com espaçamento igual a 0.5 Å. Logo, o maior tamanho total de *batch* possível utilizando multi-GPU era igual a 16. Por isso, mostrou-se necessário o uso da técnica de acúmulo de gradiente para simular *batchs* maiores e possibilitar que mais classes fossem cobertas a cada iteração do treinamento e atualização do modelo.

No final deste projeto foi disponibilizado pelo grupo Tepui do CNPEM um cluster com mais capacidade de processamento e memória, o qual foi utilizado na validação cruzada "k-fold" do melhor modelo e nos modelos treinados com tamanhos de *batch* maiores do que 8 imagens. O cluster disponibilizado pelo grupo Tepui possui as seguintes configurações: CPU AMD EPYC 7742 com 64 cores e 80 threads disponibilizadas, 384Gb Ram e 3 GPUs NVIDIA HGX A100 com 40Gb cada. A única dificuldade no uso desse cluster é o fato dos processos serem interrompidos após 3 dias seguidos de alocação de recurso, o que acaba atrasando os resultados pela necessidade de reiniciar processos longos repetidas vezes (que é o caso dos treinamentos realizados).

2.5 *NP³ Blob Label* : Aplicação para uma Nova Entrada (.mtz + .pdb)

A terceira e última etapa deste projeto de mestrado foi a implementação de uma aplicação em linha de comando para utilizar os modelos de aprendizado profundo obtidos para a análise de uma nova entrada coletada.

Na pesquisa de descoberta de novos fármacos, quando é feita uma nova coleta de dados de um experimento de cristalografia de proteínas com a metodologia de "soaking" do cristal com uma amostra bioativa, espera-se encontrar no mapa da diferença desse cristal regiões de densidade extra (denominadas *blobs*) referentes à molécula bioativa (o ligante) presente na amostra bioativa e que se liga às proteínas do cristal. Esta densidade extra pode representar um PN ligante da proteína. Os *blobs* são regiões da densidade eletrônica residual que não foram modeladas, ou seja, que não são explicadas pelas estruturas da proteína ou solvente.

Após a coleta de dados, segue-se para ciclos de processamento e refinamento dos dados para se obter os primeiros mapas 3D da densidade eletrônica do cristal de proteína em estudo. A partir desses mapas 3D é possível seguir para mais ciclos de refinamento que podem melhorar a correspondência entre a estrutura química da proteína e a imagem ali presente, assim como a qualidade da imagem da densidade em si. Em seguida, inicia-se

a etapa de análise dos dados, onde é realizada a busca por *blobs* no mapa da diferença a fim de encontrar regiões onde a molécula bioativa se encontra (regiões que não são explicadas pela estrutura da proteína). Essa busca pode ser feita com uma varredura manual da imagem da densidade residual por meio de uma inspeção visual por regiões de alta densidade no mapa ou automaticamente, utilizando algoritmos que buscam por regiões no mapa da densidade acima de um dado valor de contorno e que obedecem certos critérios definidos para se caracterizar um *blob* (como volume, soma das intensidades dos pontos do *blob* e valor mínimo do pico de intensidade do *blob*) [38, 111, 110]. Finalmente, após os *blobs* serem identificados inicia-se a interpretação da densidade eletrônica extra na tentativa de elucidação da estrutura química do ligante que melhor se encaixa no *blob* encontrado.

Com o intuito de automatizar o processamento e análise de dados de cristalografia de proteínas com "soaking" de ligantes desconhecidos, os procedimentos desse experimento foram automatizados em um workflow com cinco etapas principais e essa aplicação foi chamada de *NP³ Blob Label*. O workflow do *NP³ Blob Label* começa obtendo os mapas 3D da densidade eletrônica residual e buscando por *blobs*, regiões onde pode existir um ligante. Em seguida, os *blobs* são convertidos para imagens em formato de nuvem de pontos 3D e essas imagens são rotuladas utilizando um modelo de aprendizado profundo de segmentação semântica 3D da densidade eletrônica residual. No fim a aplicação converte os resultados da predição do modelo em mapas no formato CCP4 para facilitar sua visualização no software Coot junto com os dados de entrada. Dessa forma, as partes que compõem a estrutura dos ligantes podem ser preditas pelo modelo e utilizadas como ponto de partida na reconstrução da estrutura química completa de ligantes.

O workflow da aplicação *NP³ Blob Label* para uma nova entrada coletada (arquivos .mtz e .pdb) consiste nos seguintes passos:

1. Refinamento e obtenção do mapa da densidade eletrônica residual (Fo-Fc)
2. Busca por *blobs* no mapa Fo-Fc
3. Criação de imagens em nuvem de pontos 3D dos *blobs* a partir do mapa Fo-Fc
4. Predição das imagens dos *blobs* usando um modelo de segmentação 3D da densidade residual de ligantes
5. Conversão do resultado da predição para mapas CCP4

A entrada para o *NP³ Blob Label* é um metadado descrevendo as coletas que devem ser processadas e analisadas, a pasta de dados onde se encontram as entradas das coletas (.pdb e .mtz ou refinamento previamente feito), um modelo de segmentação 3D da imagem da densidade residual de ligantes e parâmetros para a busca de *blobs*. A saída é criada em uma pasta dentro da pasta de dados, com subpastas para cada coleta. Dentro das pastas de saída de cada coleta se encontram um relatórios com todos os *blobs* encontrados e suas classes preditas, as imagens em nuvem de pontos criadas, as predições em mapas CCP4, um arquivo .pdb com átomos falsos localizados na posição de cada *blob* encontrado, para facilitar a navegação pelo resultado, e um *script* Coot para visualização automatizada do

resultado. Um esquema das etapas da aplicação é mostrado na Figura 2.15. Cada etapa da aplicação será descrita em mais detalhes nas próximas subseções.

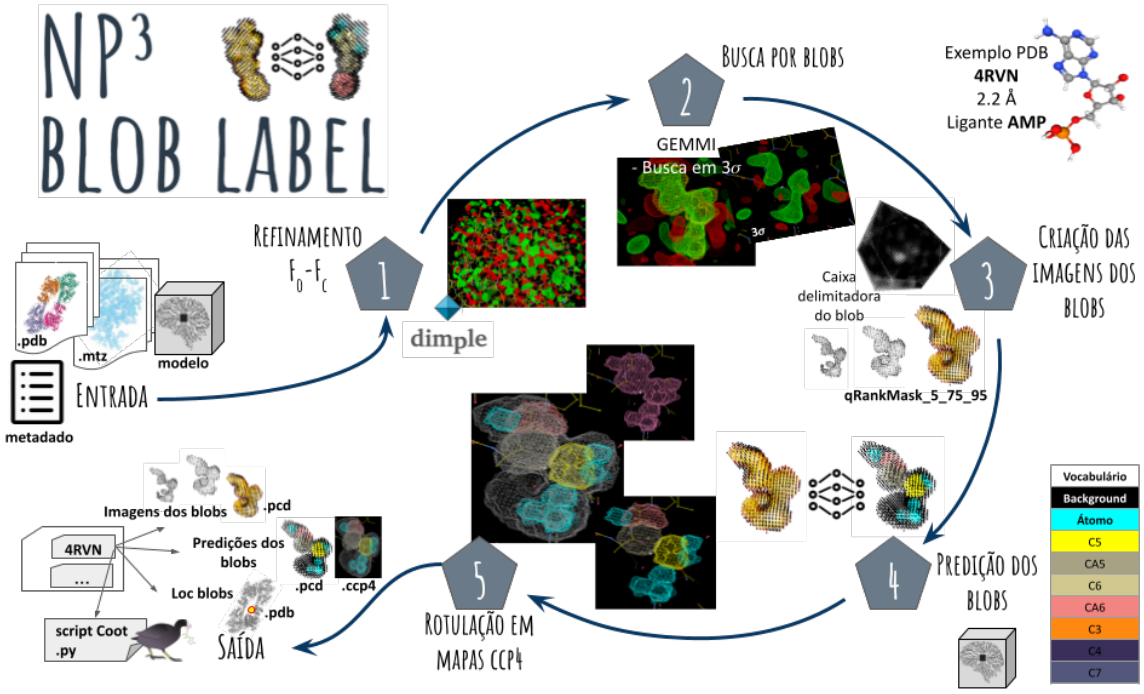

Figura 2.15: Esquema do workflow da aplicação *NP³ Blob Label*. As imagens ilustrando cada uma das 5 etapas da aplicação foram feitas utilizando a entrada do PDB 4RVN com 2.2 Å de resolução e foco no *blob* do ligante AMP.

2.5.1 Refinamento e Obtenção do Mapa 3D de Densidade Eletrônica Residual

A primeira etapa do workflow do *NP³ Blob Label* é a obtenção do mapa 3D da densidade eletrônica residual do cristal a partir do refinamento dos dados de entrada (coeficientes de difração experimental .mtz e estrutura da proteína .pdb) de cada coleta. O refinamento é feito utilizando o software Dimple [97], que consiste em um pipeline para refinamento macromolecular de cristalografia de proteínas baseado nos softwares da suíte CCP4 [110]. O Dimple é executado para cada coleta e retorna um modelo atômico refinado e mapas da densidade eletrônica residual que apontam para *blobs* não modelados da densidade eletrônica.

Nessa etapa, o usuário também tem a opção de fornecer uma pasta com os refinamentos das coletas previamente executados e contendo os mapas de densidade eletrônica obtidos.

2.5.2 Busca por *Blobs* no Mapa da Densidade Eletrônica Residual

A segunda etapa do workflow consiste em descobrir onde no mapa 3D da densidade eletrônica residual se encontra a molécula de interesse, sua localização, tamanho e intensidade. Ou seja, é a busca por *blobs* na densidade eletrônica residual.

O pacote Gemmi [111] do Python é utilizado para buscar os *blobs* relevantes que passam nos critérios definidos pelos parâmetros de entrada. A função do Gemmi utilizada nessa busca é baseada em um algoritmo de "Flood fill", utilizado para encontrar pontos conectados em uma região (área ou volume) que obedecem alguns critérios de características da densidade eletrônica residual. Os critérios dos pontos considerados são possuir intensidade acima de certo contorno sigma escolhido, soma das intensidade dos pontos conectados estar acima de um limite mínimo e o ponto mais intenso estar acima de um limite mínimo.

A busca é executada para cada coleta e retorna uma lista de *blobs*, e para cada *blob* é retornado seu volume acima do contorno, a soma da intensidade dos seus pontos, a sua posição central no mapa da densidade, a posição do seu pico de intensidade (ponto mais intenso) e o valor de intensidade do seu pico.

Um fluxo alternativo também foi implementado nessa etapa para possibilitar buscar por *blobs* em posições específicas fornecidas pelo usuário, neste caso o algoritmo só retorna a lista de *blobs* encontrados nas posições fornecidas.

Ao final dessa etapa é criado para cada coleta um arquivo pdb com a estrutura refinada mais átomos falsos inseridos na posição de cada *blob* encontrado. Esses átomos falsos são inseridos em uma cadeia falsa da proteína, criada e colocada no final da estrutura refinada.

2.5.3 Criação das imagens 3D dos *Blobs* a partir do Mapa da Densidade Eletrônica Residual

A terceira etapa do workflow consiste na criação das imagens dos *blobs* em nuvem de pontos 3D com os valores da densidade eletrônica residual devidamente processados.

Para cada *blob* de cada coleta é feito o seguinte procedimento. Primeiro é criada a imagem de uma caixa delimitadora do *blob*, a qual deve englobar toda a região do *blob* e ser utilizada para extrair sua imagem do mapa da densidade eletrônica residual. Essa caixa tem formato cúbico, é centralizada na posição central do *blob* e suas dimensões são calculadas a partir do volume do *blob*. O tamanho das laterais dessa caixa é calculado considerando o volume do *blob* igual ao volume de uma esfera e obtendo o diâmetro dessa esfera. O tamanho das laterais da caixa são definidos iguais a 300% desse diâmetro. Em seguida, é criada a imagem da caixa delimitadora do *blob* em nuvem de pontos a partir da interpolação do mapa da densidade eletrônica residual nas posições da caixa. A intensidade de cada ponto no mapa da densidade é atribuída como cor a cada ponto da imagem da caixa do *blob*. Depois os valores da imagem da caixa delimitadora são processados para transformar e normalizar o valor da densidade eletrônica residual dessa região usando a escala quantile rank. Finalmente, é criada a imagem da máscara do *blob*, seguindo a metodologia de criação da imagem da máscara dos ligantes, e são aplicados diferentes contornos na escala quantile rank para extrair imagens mais finas do *blob* e convertê-las para o formato esperado pelo modelo preditivo. Os contornos aplicados na imagem da máscara do *blob* são 0.75, 0.85 e 0.95 na escala quantile rank e também é criada a imagem da máscara do *blob* com contornos diferentes em cada canal de cor ("qRankMask_5_75_95").

2.5.4 Predição das Imagens 3D dos *Blobs* e Conversão para Mapas CCP4

A quarta etapa desse workflow consiste na rotulação da densidade eletrônica residual do *blob* e sugestão de estruturas químicas que expliquem a densidade extra sendo observada. Nesta etapa, o modelo de segmentação de imagens da densidade de ligantes fornecido como entrada é utilizado para rotular as imagens de cada *blob* e obter as predições do modelo para cada ponto.

Finalmente, na quinta etapa do workflow do *NP³ Blob Label*, a predição do modelo é convertida para um mapa CCP4 simulando uma densidade eletrônica para facilitar a visualização do resultado. Essa conversão obedece à seguinte lógica, para cada classe predita de cada coleta: atribui-se uma intensidade igual a 9 na posição de todos os pontos preditos dessa classe e uma intensidade igual a 6 a todos os pontos que estiverem a uma distância igual a 75% do espaçamento dos pontos da imagem. Logo, para cada ponto predito de uma determinada classe é atribuído no respectivo mapa CCP4 uma esfera de intensidade ao seu redor, centralizada na sua posição e com raio igual a 75% do espaçamento da imagem.

Um script em Python para o Coot é criado para cada coleta para facilitar a visualização do resultado. Esse script pode ser executado pelo Coot para abrir automaticamente a visualização de todos os mapas das classes preditas junto com os dados de entrada do refinamento da respectiva coleta. Esse script também carrega o pdb da estrutura com átomos falsos centralizados na posição de cada *blob* predito, inseridos em uma nova cadeia falsa da proteína. O usuário pode navegar pelos *blobs* e visualizar sua predição usando a navegação de átomos do Coot.

Capítulo 3

Resultados e Discussão

3.1 Modelos de Aprendizado Profundo para Segmentação Semântica da Imagem 3D da Densidade Eletrônica Residual de Ligantes

3.1.1 Primeiros Modelos com Vocabulário das Classes SP na Faixa de Resolução de 1.5 Å à 1.8 Å

O primeiro vocabulário utilizado neste trabalho para treinamento de um modelo de aprendizado profundo para segmentação da imagem 3D da densidade eletrônica residual de ligantes foi o "Vocabulário de Classes SP com Ciclos CA34567". Com o intuito de evitar um mau desempenho do modelo devido a diferenças entre as imagens de ligantes vindas de entradas em diferentes resoluções da densidade eletrônica, apenas as imagens em um intervalo entre 1.5 Å e 1.8 Å de resolução foram utilizadas nos primeiros treinamentos.

As imagens do conjunto de dados utilizadas nesses primeiros treinamentos foram criadas com um espaçamento dos pontos igual a 0.2 Å, utilizando os filtros de qualidade globais e o filtro de remoção dos pontos muito distantes dos átomos dos ligantes. Esse conjunto de dados é chamado de "Lig-qRankDB-SP-1.5-1.8" e possui 57360 imagens de ligantes, sem remoção de repetições de tipos de ligantes. As repetições variam de uma ocorrência para 2274 tipos de ligantes até 10538 ocorrências para o ligante ETHYLENE GLYCOL, com o código EDO. Isso evidencia o enviesamento do conjunto para poucos tipos de ligantes muito frequentes. As classes do "Vocabulário de Classes SP com Ciclos CA34567" utilizadas para rotular todas essas imagens são apresentadas no eixo X da Figura 3.1 e totalizam 20 classes, mais a classe de ruído de fundo ("background"). A distribuição da ocorrência das classes por átomo rotulado dos ligantes do conjunto "Lig-qRankDB-SP-1.5-1.8" é apresentada na Figura 3.1. Como a classe "background" não é utilizada para rotular os átomos dos ligantes, ela não aparece nessa distribuição.

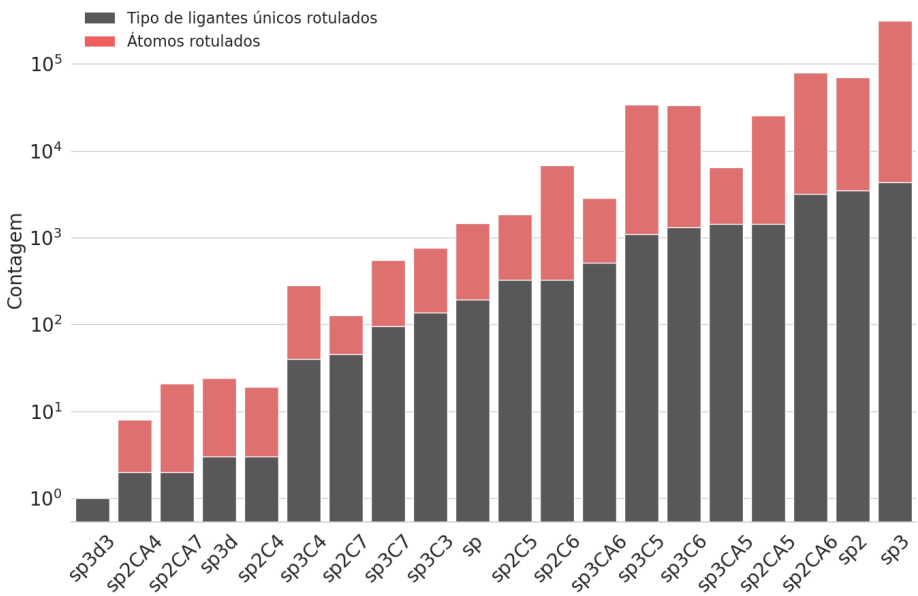

Figura 3.1: Distribuição da ocorrência das classes do "Vocabulário de Classes SP com Ciclos CA34567" por átomo rotulado dos ligantes presentes no conjunto de dados "Lig-qRankDB-SP-1.5-1.8", onde $d_{max} = 313850$.

A distribuição da Figura 3.1 mostra o grande desbalanceamento existente entre as classes do "Vocabulário de Classes SP com Ciclos CA34567" nesse conjunto de dados, com a classe 'sp3' majoritária aparecendo em 313850 átomos de 56426 entradas de ligantes e a classe 'sp3d3' minoritária aparecendo em apenas um átomo de uma única entrada de ligante. As classes que aparecem em menos do que 1000 átomos indicam classes que serão provavelmente difíceis de convergir. Para auxiliar na generalização do modelo e diminuir o enviesamento do conjunto para classes majoritárias e estruturas químicas frequentes, o Algoritmo 5 de *undersampling* foi aplicado. Os parâmetros utilizados no Algoritmo 5 foram a quantidade mínima de átomos (minAtomos) diferentes de hidrogênio igual a 5, a quantidade máxima de ligantes por tipo de ligante (maxLigCode) igual a 500, a quantidade mínima de ocorrência das classes em diferentes entradas igual a 1000 (minClassOcc) e a máxima ocorrência igual a 10000 (maxClassOcc). Ligantes com menos do que 5 átomos foram removidos por terem menos restrições geométricas, e logo, podem apresentar mais ruído devido a uma maior mobilidade dessas moléculas.

O resultado do Algoritmo 5 foi um novo conjunto de dados com 9317 entradas de ligantes, cobertos por 7 classes do "Vocabulário de Classes SP com Ciclos CA34567" mais a classe de "background". Esse conjunto de dados foi chamado aqui de "Lig-qRankDB-SP-1.5-1.8 8classes". A nova distribuição das classes nesse conjunto é apresentada na Figura 3.2. O desbalanço entre as classes no conjunto "Lig-qRankDB-SP-1.5-1.8 8classes" foi reduzido substancialmente, com a ocorrência das classes por entrada de ligante variando entre 3077 e 8898 entradas. Por outro lado, apenas as 8 classes mais abundantes passaram nos filtros, as demais foram removidas.

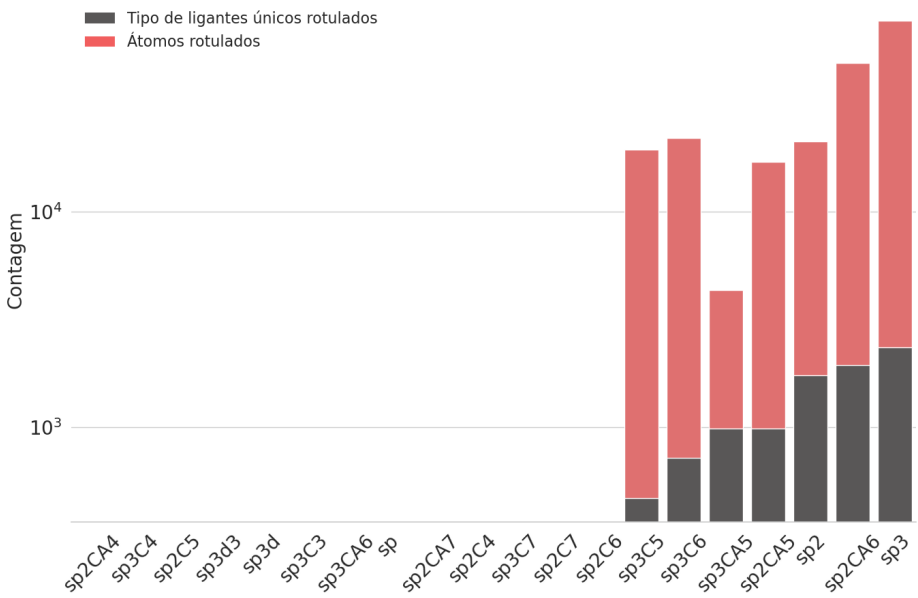

Figura 3.2: Distribuição da ocorrência das classes do "Vocabulário de Classes SP com Ciclos CA34567" por átomo dos ligantes presentes no conjunto de dados "Lig-qRankDB-SP-1.5-1.8 8classes", após aplicação do método de *undersampling*, onde $d_{max} = 17.9$.

O conjunto de dados "Lig-qRankDB-SP-1.5-1.8 8classes" foi particionado em $k = 6$ subconjuntos estratificados utilizando o Algoritmo 6 e o método de validação cruzada utilizado nos treinamentos foi o "hold-out", onde as entradas com $k = 1$ foram utilizadas para o conjunto de teste e de validação. O conjunto de treinamento obtido foi utilizado em três treinamentos. Os treinamentos serão identificados utilizando o nome do conjunto de dados mais um índice único para identificar cada treinamento separadamente, no seguinte formato: "Treinamento X - Lig-qRankDB-SP-1.5-1.8 8classes", onde $X = 1, 2, 3$. Esses treinamentos avaliam o uso de 2 imagens diferentes (Treinamentos 1 e 2) no desempenho do modelo e o impacto do uso de peso na função de perda (Treinamento 3) para lidar com o desbalanço remanescente e com as classes difíceis de convergir. As imagens utilizadas são a "qRankMask" nos Treinamentos 1 e 3, e a "qRank0.7" no Treinamento 2. O peso utilizado no Treinamento 3 foi igual à taxa de desbalanço de cada classe, exceto para a classe do "background" que recebeu um peso igual a 0.35, por ser a classe mais frequente na imagem da máscara do ligante. Todos os treinamentos foram executados por 60 épocas utilizando a rede MinkUNet34C, com um tamanho total de batch igual a 10, utilizando o otimizador Adam com parâmetros $beta1 = 0.9$ e $beta2 = 0.999$, uma taxa de aprendizado constante igual a 2^{-8} e com as funções de perda CE e sua versão com peso, a wCE. As configurações que diferenciam esses treinamentos são mostradas na Tabela 3.1. Os resultados finais de acurácia desses treinamentos em termos de mIoU e do valor final da função de perda são comparados na Tabela 3.1, destacado em negrito os melhores valores de mIoU. As curvas de aprendizado da validação desses três treinamentos são mostradas na Figura 3.3.

Tabela 3.1: Configuração, acurácia mIoU e valor da função de perda finais das etapas de treino, validação e teste dos três treinamentos com o conjunto Lig-qRankDB-SP-1.5-1.8 8classes

Nome	Tipo de Imagem	Função de Perda	Treino		Validação		Teste	
			Função de Perda (Loss)	**mIoU**	**Função de Perda (Loss)**	**mIoU**	**Função de Perda (Loss)**	**mIoU**
Treinamento 1	qRankMask	CE	0.93	**13.87**	0.96	**14.15**	0.95	**13.87**
Treinamento 2	qRank0.7	CE	1.58	10.8	1.62	11.31	1.65	10.87
Treinamento 3	qRankMask	wCE	1.84	12.94	1.61	12.57	1.79	12.43

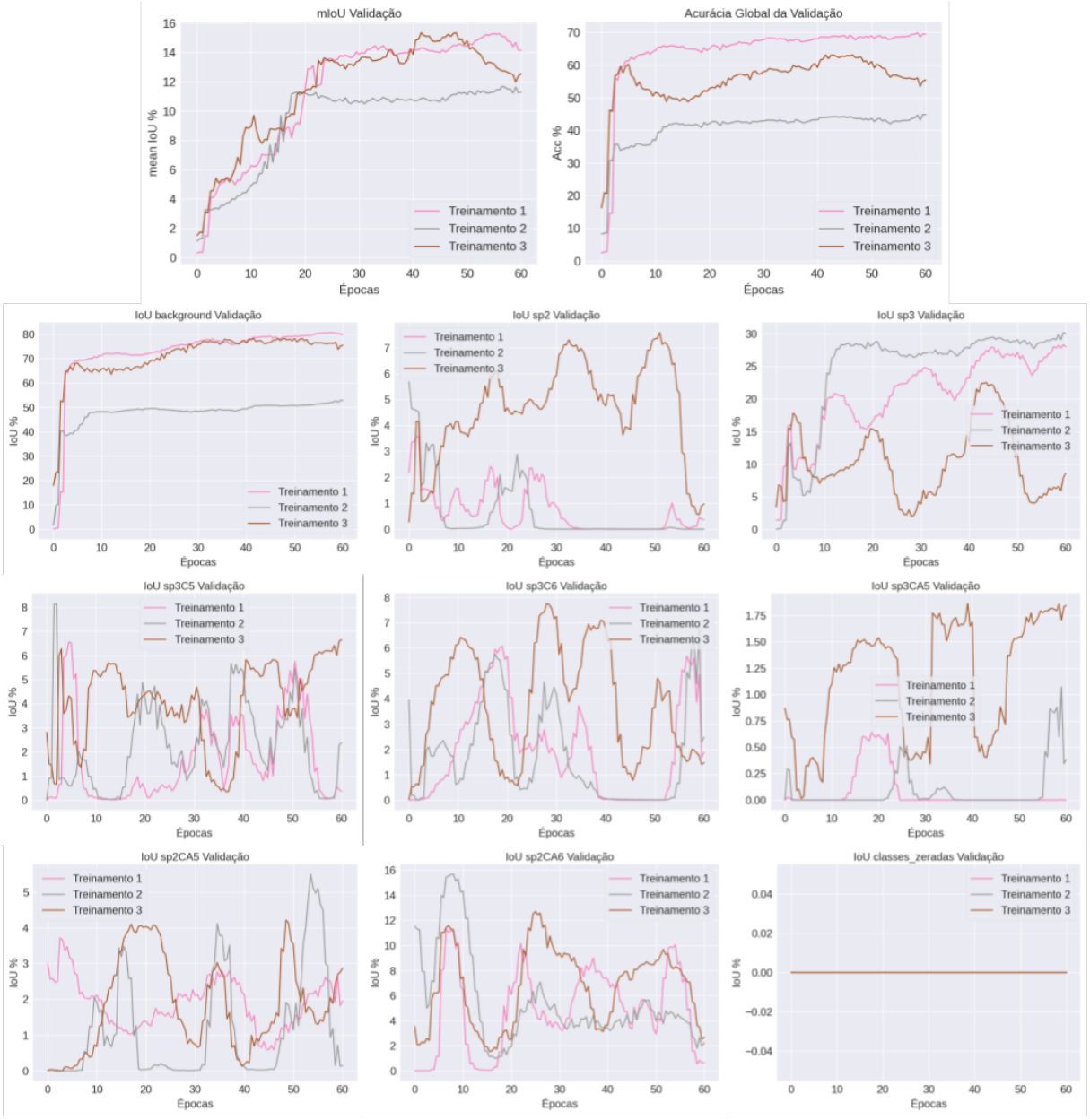

Figura 3.3: Curvas de aprendizado da validação dos três treinamentos com o conjunto de dados Lig-qRankDB-SP-1.5-1.8 8classes. Curvas da acurácia IoU da validação por classe, curvas de mIoU e da acurácia média global dos modelos. As classes que ficaram com a acurácia IoU zerada estão sendo mostradas com a mesma imagem para facilitar a visualização e foram chamadas de "classes_zeradas". As classes "classes_zeradas" são as seguintes: sp, sp2C4, sp2C5, sp2C6, sp2C7, sp2CA4, sp2CA7, sp3C3, sp3C4, sp3C7, sp3CA6, sp3d, sp3d3.

As trajetórias das curvas de validação dos três treinamentos apresentadas na Figura 3.3 ilustram um comportamento de um aprendizado degradado. Apenas a classe de ruído de fundo convergiu para um bom desempenho de mais do que 75% de acurácia (IoU) nos Treinamentos 1 e 3. A classe 'sp3' majoritária apresentou um sinal de convergência chegando a quase 30% de acurácia (IoU) nos Treinamentos 1 e 2, mas com pouca estabilidade. O uso de peso na função de perda do Treinamento 3 prejudicou a convergência da classe majoritária 'sp3', mas permitiu que as classes 'sp2' e 'sp3CA5' não ficassem zeradas. Todas as demais classes apresentaram uma amplitude muito grande na trajetória, sem melhoras significativas nessas 60 épocas e com valores de IoU em torno de 5% ou menor. Outras variações de configurações que alteram parâmetros da rede também foram testadas sem sucesso (treinamentos não apresentados).

Esse resultado mostra que o modelo não foi capaz de aprender a modelagem proposta, mas teve um ótimo desempenho em remover o ruído de fundo da imagem do ligante, com boa convergência a partir de 10 épocas. A métrica de acurácia média global (total de acertos dividido pelo total de pontos) acaba ficando enviesada pelas classes mais abundantes, que neste caso são as classes do ruído de fundo e 'sp3', e por isso apresenta valores muito melhores do que a métrica mIoU. Essa grande diferença entre as métricas ilustra a necessidade de uso da mIoU para melhor avaliar um modelo de segmentação semântica na presença de desbalanço entre as classes e alerta que o valor da acurácia média global pode levar a conclusões erradas. Nos próximos treinamentos, apenas a métrica global mIoU será apresentada.

Algumas possibilidades foram elencadas inicialmente para explicar esse aprendizado degradado: a complexidade da modelagem (classes difíceis de convergir); pouca quantidade de dados de entrada; um tamanho total de *batch* pequeno; e ruído na rotulação (entrada e saída erradas). Após buscar na literatura referências de causas para degradação de treinamentos de aprendizado profundo, algumas alternativas foram encontradas e a questão de ruído na rotulação teve destaque. Muitas referências sobre ruído na rotulação apresentam curvas de aprendizado com comportamento semelhante ao observado na Figura 3.3, com uma amplitude muito grande e baixa convergência [109, 115, 6]. A função de perda SL foi escolhida para ser testada para aliviar o impacto no treinamento devido a possíveis ruídos na rotulação e para auxiliar o aprendizado de classes difíceis de convergir. Além disso, o uso de um tamanho de *batch* maior pode auxiliar o desempenho de modelos mais complexos [43, 73] e também passou a ser avaliado. Devido a limites de hardware a técnica de acúmulo de gradiente foi implementada para possibilitar simular tamanhos de *batch* maiores e avaliar seu impacto no modelo final.

Devido ao baixo desempenho dessa modelagem, foi decidido testar essas alternativas em uma modelagem mais simples primeiro para validar a viabilidade da abordagem proposta. A partir de uma modelagem simples com sucesso é possível entender melhor os limites do modelo e adicionar mais complexidade aos poucos.

3.1.2 Modelos com Vocabulário Simplificado na Faixa de Resolução de 1.8 Å à 2.2 Å

A modelagem mais simples para o problema de segmentação semântica das imagens 3D da densidade eletrônica de ligantes se reduz a separar apenas a região onde os átomos do ligante estão do ruído de fundo. Ou seja, é a modelagem para segmentação da região do ligante representada pelo "Vocabulário da Região do Ligante", com apenas duas classes: átomo e ruído de fundo ("background"). Outra faixa de resolução dos dados foi definida para os treinamentos com esse vocabulário entre 1.8 Å e 2.2 Å, a qual possui mais dados de entrada.

O Algoritmo 5 de *undersampling* foi aplicado nesse conjunto de dados para limitar o número de repetições de tipos de ligantes a 500, manter apenas ligantes com mais do que 4 átomos diferentes de hidrogênio e remover entradas em que aparecem as classes minoritárias de ciclos de tamanho 3, 4 e 7. Escolheu-se remover as entradas com as classes cíclicas minoritárias para evitar que elas atrapalhassem a convergência do modelo e para remover possíveis causas de variações no desempenho do modelo.

As imagens dos ligantes foram criadas utilizando um espaçamento de 0.2 Å e com o filtro de distância até os átomos do ligante. Isso resultou no conjunto de dados denominado "Lig-qRankDB-SP-1.8-2.2-noCA347", sem filtro de qualidade e com 32761 entradas das quais 2048 foram utilizadas para teste, 2048 para validação e as 28665 restantes para treino. Os filtros de qualidade global também foram aplicados nesse conjunto e resultaram no conjunto de dados denominado "Lig-qRankDB-SP-1.8-2.2-noCA347 GlobalFilterQuali" com 24388 entradas das quais 1523 foram para teste, 1523 foram para validação e 21342 para treino.

Serão apresentados 4 treinamentos, dois deles foram feitos com o conjunto de dados "Lig-qRankDB-SP-1.8-2.2-noCA347" e são denominados "Treinamento 1" e "Treinamento 2". Os outros dois treinamentos foram feitos com o conjunto "Lig-qRankDB-SP-1.8-2.2-noCA347 FiltroQualiGlobal" e são denominados "Treinamento 3" e "Treinamento 4". Todos os treinamentos foram executados até 50 épocas utilizando o otimizador Adam com parâmetros $beta1 = 0.9$ e $beta2 = 0.999$, uma taxa de aprendizado constante igual a 2^{-8}, a rede MinkUNet34C e as imagens "qRankMask_5_75_95". Os Treinamentos 1 e 2 utilizaram a função de perda wSL e o Treinamento 2 avaliou o impacto de um tamanho de batch maior e igual a 64 utilizando a técnica de acúmulo de gradiente (8 acúmulos foram utilizados). Já os Treinamentos 3 e 4 avaliaram o uso da função de perda wSL em comparação com a CE sem peso, e a aplicação do filtro de qualidade global no conjunto de treinamento. A configuração desses quatro treinamentos é apresentada na Tabela 3.2. O peso da função de perda (loss) corresponde aos valores atribuídos às classes background e átomo, nessa ordem, separados por um hífen.

Tabela 3.2: Configurações dos Treinamentos de 1 a 4 com os Conjuntos de Dados "Lig-qRankDB-SP-1.8-2.2-noCA347" e "Lig-qRankDB-SP-1.8-2.2-noCA347 FiltroQualiGlobal" para o "Vocabulário da Região do Ligante".

Nome	**Quantidade de Entradas (treino|teste|validação)**	**Filtro de Qualidade Global**	**Função de Perda (Loss)**	**Peso da Loss (w)**	**Tamanho Total do Batch**
Treinamento 1	28665 | 2048 | 2048	Não	wSL	1-4.0	64
Treinamento 2	28665 | 2048 | 2048	Não	wSL	1-4.0	10
Treinamento 3	21342 | 1523 | 1523	Sim	CE	1	10
Treinamento 4	21342 | 1523 | 1523	Sim	wSL	1-2.5	10

O "Vocabulário da Região do Ligante" foi utilizado nesse conjunto por meio de um mapeamento simples das classes do "Vocabulário de Classes SP com Ciclos CA34567" para a classe "Átomo" e a classe do ruído de fundo foi mantida. Esse é o mapeamento 3 mostrado na Tabela 2.2 e resultou nos primeiros resultados com bom desempenho do modelo de aprendizado profundo sendo avaliado. As curvas de aprendizado desses 4 treinamentos são mostradas na Figura 3.4. As curvas de treino dos Treinamentos de 1 a 4 foram nomeadas como T1, T2, T3 e T4, respectivamente, e as curvas de validação foram nomeadas da mesma maneira como V1, V2, V3 e V4. O tamanho total do *batch* utilizado em cada treinamento também foi incluído no nome das curvas com a variável *b*, junto com a função de perda utilizada que é informada após a marcação 'loss' seguida dos valores atribuídos como peso após a marcação 'w' quando presente, sendo o primeiro valor do peso referente à classe do "background" e o segundo valor referente à classe do "átomo".

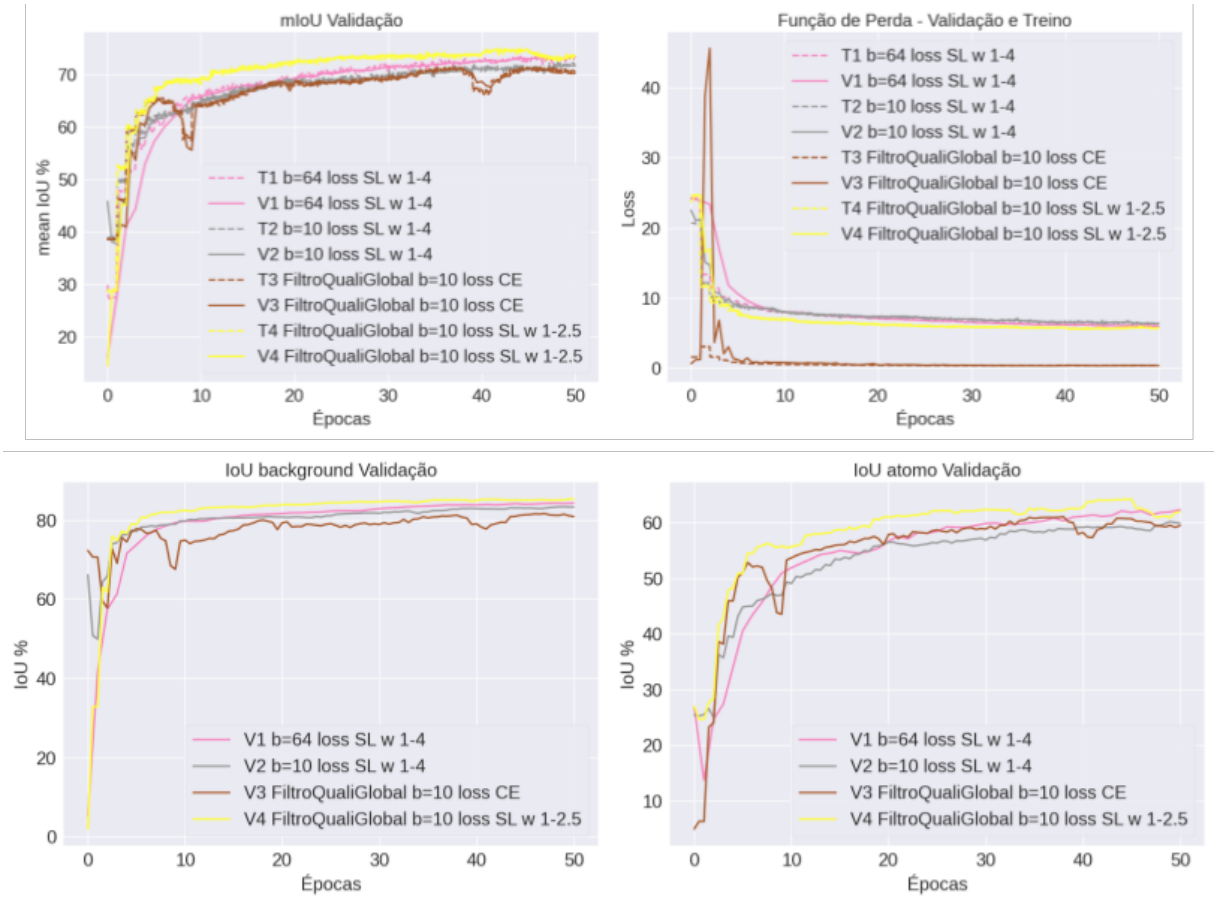

Figura 3.4: Curvas de aprendizado do treino da validação dos quatro treinamentos com os conjuntos de dados "Lig-qRankDB-SP-1.8-2.2-noCA347" e "Lig-qRankDB-SP-1.8-2.2-noCA347 FiltroQualiGlobal" para o "Vocabulário da Região do Ligante". São apresentadas as curvas de acurácia IoU da validação por classe e as curvas de mIoU e das funções de perda da validação (linha contínua) e do treino (linha tracejada).

Os resultados de treino, validação e teste desses quatro treinamentos são mostrados na Tabela 3.3 para comparação da acurácia média em termos de mIoU (melhor valor destacado em negrito) e do valor da função de perda finais. A matriz de confusão do Treinamento 4, que teve o melhor desempenho, é mostrada na Tabela 3.4.

Tabela 3.3: Resultado de acurácia e da função de perda para os Treinamentos de 1 a 4 com os Conjuntos de Dados "Lig-qRankDB-SP-1.8-2.2-noCA347" e "Lig-qRankDB-SP-1.8-2.2-noCA347 FiltroQualiGlobal" para o "Vocabulário da Região do Ligante".

Nome	Treino		Validação		Teste	
	Função de Perda (Loss)	**mIoU**	**Função de Perda (Loss)**	**mIoU**	**Função de Perda (Loss)**	**mIoU**
Treinamento 1	6.06	72.7	5.98	73.3	6.12	73.1
Treinamento 2	6.54	71.32	6.4	71.61	6.45	71.41
Treinamento 3	0.4	70.62	0.38	70.19	0.39	69.99
Treinamento 4	5.85	**73.06**	5.69	**73.61**	5.56	**74.21**

Tabela 3.4: Matriz de confusão do Teste do Treinamento 4 com o Conjunto de Dados "LigqRankDB-SP-1.8-2.2-noCA347 FiltroQualiGlobal" e "Vocabulário da Região do Ligante".

	Background	**Átomo**
Background	**85.54**	4.82
Átomo	24.75	**62.88**

Os treinamentos com o vocabulário simplificado tiveram ótimos resultados, com um mIoU no teste acima ou muito próximo de 70%. Existe uma grande dificuldade na avaliação desses resultados devido a falta de referência justa para se avaliar um corte de acurácia aceitável para o problema. Nesse cenário, quanto maior for a acurácia melhor será o resultado e valores acima de 50% destacam a capacidade do modelo em distinguir entre as classes. Soma-se a isso o fato de existir erros na borda da predição dos ligantes (melhor explicado abaixo), o que não compromete as classes preditas corretamente, mas diminui a acurácia da solução em algumas entradas. Por inspeção visual, algumas entradas com acurácias em torno de 60% apresentaram predições corretas e viáveis de serem utilizadas, mas não foi possível definir uma tolerância fixa desejada. De toda forma, com esse resultado ficou comprovada a viabilidade da abordagem: é possível treinar um modelo de aprendizado profundo para segmentação semântica da imagem da densidade residual de ligantes em nuvem de pontos 3D! A partir dessa validação, outras modelagens mais complexas puderam ser testadas com mais confiança.

Na matriz de confusão do Treinamento 4 apresentada na Tabela 3.4 é notável que o maior erro do modelo está em confundir os pontos que deveriam ser da classe átomo (região do ligante) com a classe do ruído de fundo. Apenas com a informação dessa tabela não é possível saber se o ruído de fundo está sendo colocado no meio da região do ligante ou na borda dessa região, mais próximo aos pontos rotulados como ruído de fundo verdadeiro. O erro na borda da região dos ligantes pode ser devido a diferenças entre a conformação predita e a esperada ou a diferenças entre o raio atômico predito e o esperado. A visualização da predição de algumas imagens desse conjunto de teste fortaleceu a hipótese de erro na borda da região do ligante (não mostrada para esse resultado), o que será abordado com mais detalhe nos resultados dos últimos treinamentos. Densidades fragmentadas que indicam baixa correspondência entre a imagem do ligante e a estrutura depositada e adicionam mais erro na rotulação podem estar contribuindo para essa confusão. Novamente os modelos obtidos tiveram um ótimo desempenho em separar o ruído de fundo da região do ligante e fortaleceram a hipótese desse trabalho.

O uso da função de perda wSL no Treinamento 4 ajudou a estabilizar a curva de aprendizado das duas classes, é possível observar uma menor amplitude na sua trajetória e uma convergência um pouco melhor e mais rápida em comparação com o Treinamento 3, feito com a função de perda CE. O uso de um conjunto de dados com mais entradas nos Treinamentos 1 e 2, sem o uso do filtro de qualidade global, afetou pouco a curva de aprendizado. Isso indica que o modelo foi capaz de lidar com os ruídos na rotulação associados aos critérios do filtro de qualidade global com mais dados fornecidos como entrada. Além disso, um peso maior e igual a 4 foi atribuído à classe de Átomo nesses dois treinamentos, mas não resultou em um ganho significativo. O uso de um tamanho total de *batch* maior no Treinamento 1 trouxe uma pequena melhora nas curvas de aprendizado.

É possível perceber uma inclinação maior na acurácia desse treinamento fazendo seu resultado final encontrar com o resultado do Treinamento 4. Seguindo essa tendência, o Treinamento 1 poderia passar o Treinamento 4 se esses fossem executados por mais épocas.

A próxima modelagem investigada adicionou um pouco de complexidade nas classes com o uso do "Vocabulário de Átomos e Ciclos Genéricos" e avaliou a capacidade do modelo em predizer subestruturas químicas cíclicas genéricas, independentemente do seu tamanho e tipo de átomo, e átomos fora de ciclos. Três treinamentos com esse vocabulário serão apresentados. O Treinamento 1 foi feito com o conjunto de dados "Lig-qRankDB-SP-1.8-2.2-noCA347" e o Treinamento 2 foi feito com esse mesmo conjunto mais a aplicação de um filtro de qualidade local para remover entradas de ligantes com imagens no contorno qRank0.9 com menos do que 47% dos pontos esperados para cobrir a esfera atômica de todos os átomos dos ligantes. O conjunto de dados do Treinamento 2 é chamado de "Lig-qRankDB-SP-1.8-2.2-noCA347 FiltroQualiLocal" e possui 24212 entradas de treino, 1716 de validação e 1716 de teste. Esse novo conjunto de dados visava remover um pouco do ruído na rotulação devido a depósitos errados no PDB e possíveis erros de rotulação (e.g. densidades fragmentadas). O Treinamento 3 foi feito com o conjunto de dados "Lig-qRankDB-SP-1.8-2.2-noCA347 FiltroQualiGlobal", que inclui o filtro de qualidade global.

Todos os treinamentos foram executados por 50 épocas, utilizando o otimizador Adam com parâmetros $beta1 = 0.9$ e $beta2 = 0.999$, uma taxa de aprendizado constante igual a 2^{-8} e a rede MinkUNet34C. Os Treinamentos 1 e 2 foram executados com a função de perda wSL e um tamanho total de *batch* igual a 64, com acúmulo de gradiente. O Treinamento 1 avaliou a imagem qRank0.75 e os Treinamentos 2 e 3 foram feitos com a imagem qRankMask_5_75_95. O Treinamento 3 avaliou o impacto da função de perda CE para essa modelagem com 3 classes e utilizou um tamanho total de *batch* igual a 10, sem acúmulo de gradiente. As configurações desses três treinamentos estão descritas na Tabela 3.5, onde o peso da função de perda corresponde, nessa ordem, aos pesos atribuídos as classes de background, ciclo e átomo. Suas curvas de aprendizado do treino e da validação são apresentadas na Figura 3.5. A matriz de confusão do Treinamento 1 é apresentada na Tabela 3.6.

Tabela 3.5: Configurações dos Treinamentos de 1 a 3 com o Conjunto de Dados "Lig-qRankDB-SP-1.8-2.2-noCA347" com e sem Filtros de Qualidade para o "Vocabulário de Átomos e Ciclos Genéricos".

Nome	**Quantidade de Entradas (treino|teste|validação)**	**Filtro de Qualidade**	**Tipo de Imagem**	**Função de Perda (Loss)**	**Peso da Loss (w)**	**Total Batch Size**
Treinamento 1	28665 | 2048 | 2048	Não	qRank0.75	SL	1-2.0-2.0	64
Treinamento 2	24212 | 1716 | 1716	Local	qRankMask_5_75_95	SL	1-2.0-2.5	64
Treinamento 3	21342 | 1523 | 1523	Global	qRankMask_5_75_95	CE	1	10

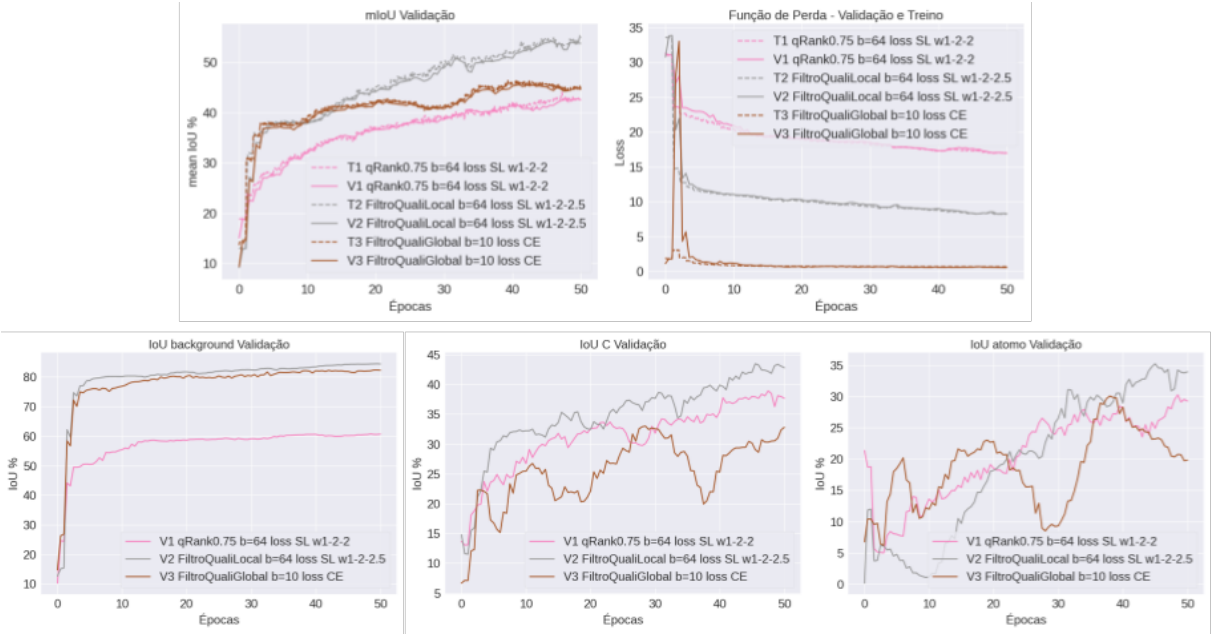

Figura 3.5: Curvas de aprendizado do treino e da validação dos três treinamentos com os conjuntos de dados "Lig-qRankDB-SP-1.8-2.2-noCA347", "Lig-qRankDB-SP-1.8-2.2-noCA347 FiltroQualiLocal" e "Lig-qRankDB-SP-1.8-2.2-noCA347 FiltroQualiGlobal" para o "Vocabulário de Átomos e Ciclos Genéricos". São apresentadas as curvas da acurácia IoU da validação por classe e em termos de mIoU para treino e validação e as curvas das funções de perda do treino e da validação.

Tabela 3.6: Matriz de confusão do Teste do Treinamento 1 com o Conjunto de dados "Lig-qRankDB-SP-1.8-2.2-noCA347" e "Vocabulário de Átomos e Ciclos Genéricos".

	Background	**C**	**Átomo**
Background	**60.52**	5.68	5.65
C	26.34	**35.82**	16.44
Átomo	30.33	10.06	**31.67**

Os treinamentos na modelagem com o "Vocabulário de Átomos e Ciclos Genéricos" tiveram bons resultados, eles foram parados mais cedo em 50 épocas com boa inclinação nas curvas de aprendizado. Isso indica que eles poderiam atingir acurácias próximas as dos treinamentos anteriores apesar de apresentarem uma convergência mais lenta. O melhor desempenho em termos de mIoU na validação foi de 53.71% para o Treinamento 2. Apesar desses três treinamentos terem sido feitos com entradas e configurações um pouco diferentes, é possível perceber que novamente o uso da função de perda SL auxiliou na estabilidade do aprendizado e na convergência do modelo (Treinamentos 1 e 2). Além disso, é possível observar uma queda na acurácia da classe de átomo em todos os treinamentos em comparação com os treinamentos anteriores, indo de 62.9% para 35.6% (Treinamento 2) nos melhores resultados. Em todos os treinamentos, a classe de átomo teve uma acurácia menor do a classe de ciclo, o que evidencia uma maior facilidade do modelo em apreender os padrões deixados por ciclos nas imagens da densidade eletrônica dos ligantes. A diferença de 10 épocas entre esses treinamentos e os anteriores não explica a grande diferença entre as acurácias da classe de átomo, essa diferença de acurácia

indica que o modelo teve mais dificuldade em aprender essa modelagem com distinção de subestruturas cíclicas.

O Treinamento 1 utilizou uma imagem diferente da máscara do ligante, apenas com o contorno de 0.75 na escala quantile rank e teve uma pior acurácia na classe do ruído de fundo igual a 60.5%. De fato o ruído de fundo é menos presente nas imagens nesse contorno em comparação com a imagem da máscara do ligante e isso pode ter afetado a convergência dessa classe. A matriz de confusão do Treinamento 1 apresentada na Tabela 3.6 mostra que o erro entre a classe esperada de átomo com o ruído de fundo neste treinamento é bem maior do que nos modelos da região do ligante. Isso fortalece a hipótese desse erro estar na borda da região do ligante, pois nessa imagem a um contorno de 0.75 apenas o ruído próximo à borda é mantido na imagem e este apresentou uma menor acurácia. A imagem da máscara do ligante com diferentes contornos em cada canal de cor utilizada no Treinamento 2 teve o melhor desempenho, com um incremento em torno de 5% de acurácia na validação IoU das classes de átomo e ciclo em comparação com o Treinamento 1 e com bom sinal de ascensão na curva de aprendizado.

Esses treinamentos foram parados mais cedo para que outras configurações e modelagens fossem avaliadas. Todos esses treinamentos apresentaram boa inclinação de subida nas curvas de aprendizado e sem sinal de estagnação ou *overfitting*, o que indica que eles poderiam continuar por mais épocas. Uma comparação sistemática do impacto de diferentes configurações no desempenho do modelo será apresentada nos próximos treinamentos, feitos com o conjunto de dados final e seguindo a mesma configuração para melhor avaliar o impacto de cada opção.

3.1.3 Modelos Finais de Segmentação Semântica da Densidade Eletrônica de Ligantes

Muitos treinamentos com o "Vocabulário de Átomos e Ciclos Genéricos" se sucederam após os bons resultados e atingiram acurácias em termos de mIoU acima de 60% após mais do que 50 épocas. Nesses novos treinamentos a pipeline de treinamento com a biblioteca pytorch-lightning foi implementada, outras configurações de rede foram avaliadas e um conjunto de dados cobrindo toda a faixa de resolução de 1.5 à 2.2 Å passou a ser utilizado sem perda de acurácia. Dentre as configurações de rede avaliadas estão o uso da convolução dilatada ou ATROUS híbrida, uso do otimizador SGD, uso de aumentação de dados para rotação aleatória nos 3 eixos de uma porcentagem das imagens durante o treinamento (aumento de diversidade do conjunto de treinamento), uso de tamanhos de *batch* diferentes com e sem acúmulo de gradiente e uso de um espaçamento dos pontos maior e igual a 0.5 Å.

Uma vez identificado que o filtro de remoção de pontos distantes dos átomos dos ligantes não é reprodutível para uma entrada nova, as imagens dos ligantes passaram a ser criadas sem o uso desse filtro. Assim, ficou a cargo do modelo aprender a remover os ruídos devidos a outras conformações das moléculas. O desempenho geral do modelo foi pouco impactado pela remoção deste filtro, foi observada uma perda de acurácia de no máximo 5% (resultado não será mostrado). Além da acurácia, sem o uso do filtro de remoção de pontos distantes, as imagens dos ligantes ficaram um pouco maiores e um

tamanho de *batch* menor precisou ser usado por causa dos limites de hardware. Com o uso de multi-GPU e da técnica de acúmulo de gradiente o tamanho total do *batch* não foi impactado. Também foi constatado que o espaçamento dos pontos igual a 0.2 Å criava imagens muito grandes e pesadas, que inviabilizam o seu uso em uma aplicação em larga escala para um computador pessoal. Além disso, fazia com que os treinamentos durassem muito tempo. Imagens com menos resolução na nuvem de pontos, com um espaçamento igual a 0.5 Å, foram testadas com sucesso, apresentando acurácias muito próximas às obtidas com as imagens em alta resolução na nuvem de pontos e passaram a ser utilizadas. O tempo de treinamento com o aumento do espaçamento das imagens diminuiu pela metade e permitiu utilizar tamanhos de *batch* maiores. Os treinamentos com as imagens com espaçamento de 0.2 Å e tamanho total de *batch* igual a 10 demoravam aproximadamente 8 dias para atingir 100 épocas e com as imagens com espaçamento de 0.5 Å esse tempo passou para um pouco menos do que 4 dias, e manteve-se para um tamanho total de *batch* igual a 16. No cluster do grupo Tepui do CNPEM esse tempo caiu pela metade, mas foi maior para os treinamento com tamanho de *batch* maior do que 16.

Os treinamentos finais que serão mostrados foram padronizados para usar o mesmo conjunto de dados, sem adição de variação na quantidade de entradas e parâmetros controlados para obtenção dos modelos finais (etapa 7 do workflow da Figura 2.1). Eles foram organizados em três conjuntos de treinamentos: o primeiro conjunto de treinamento irá apresentar o melhor modelo na configuração final estabelecida até o momento para o "Vocabulário de Átomos Genéricos e Ciclos C347CA56" junto com sua validação cruzada "k-fold"; o segundo irá apresentar uma análise sistemática do impacto das configurações principais no desempenho final desse modelo; e o terceiro conjunto de treinamento irá apresentar o desempenho da configuração final para os modelos treinados com os outros vocabulários propostos.

O conjunto de dados dos treinamentos finais contêm entradas de ligantes na faixa de resolução entre 1.5 e 2.2 Å, sem o filtro de distância até os átomos dos ligantes, com espaçamento dos pontos igual a 0.5 Å e sem filtros de qualidade. O Algoritmo 5 de *undersampling* foi aplicado com parâmetros iguais a mínimo de 3 átomos diferentes de hidrogênio, no máximo 1000 repetições por tipo de ligante e sem limites para a quantidade de ocorrência das classes. Esse conjunto de dados é chamado de "Lig-qRankDB-1.5-2.2" e possui no total 78911 entradas de ligantes. Ele foi particionado em $k = 13$ grupos estratificados com o Algoritmo 6 e resultou em 72829 entradas para treino, 3035 para validação e 3035 para teste.

Modelo Final Vocabulário de Átomos Genéricos e Ciclos C347CA56

O melhor resultado deste trabalho foi obtido com a modelagem do "Vocabulário de Átomos Genéricos e Ciclos C347CA56". Esse vocabulário foi utilizado para rotular as imagens do conjunto "Lig-qRankDB-1.5-2.2" e resultou no conjunto de treinamento "Lig-qRankDB-SP-1.5-2.2 C347CA56" com 9 classes. A distribuição das classes desse vocabulário no conjunto de treinamento "Lig-qRankDB-SP-1.5-2.2 C347CA56" é apresentada na Figura 3.6. Esse conjunto foi pensado para predizer classes de subestruturas químicas cíclicas

simplificadas (sem distinção entre tipos de átomos e ligações), de átomos fora de ciclos e ruído de fundo. As classes aprendidas pelo modelo são ciclos com tamanho de 3 a 7 átomos, sendo os ciclos de tamanho 5 e 6 aromáticos ou não, e átomos fora de ciclos: C3, C4, C5, CA5, C6, CA6, C7, Átomo e Ruído de Fundo.

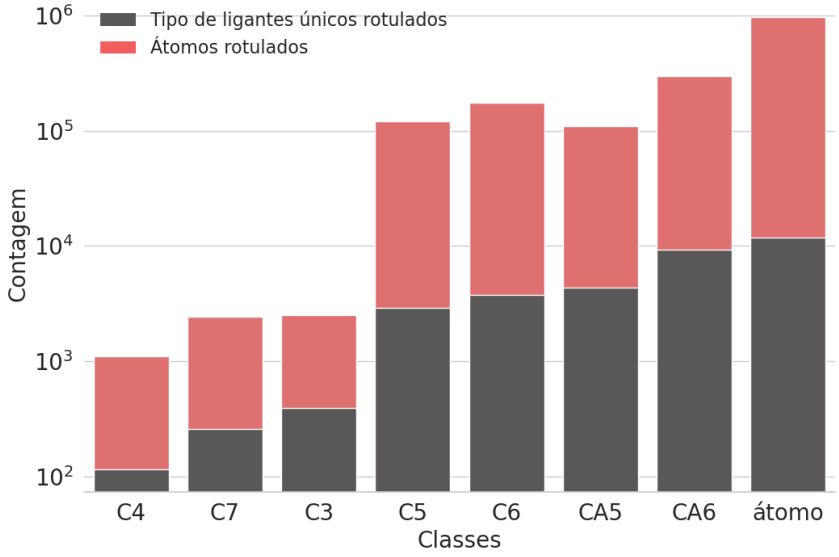

Figura 3.6: Distribuição da ocorrência das classes do "Vocabulário de Átomos Genéricos e Ciclos C347CA56" por átomo dos ligantes presentes no conjunto de treinamento "Lig-qRankDB-SP-1.5-2.2 C347CA56", onde $d_{max} = 865.4$.

A Figura 3.6 mostra que as classes minoritárias para esse conjunto são em ordem crescente C4, C7 e C3. As demais classes são majoritárias e apresentam pouco desbalanço entre si, com exceção da classe de Átomos fora de ciclos que têm quase uma ordem de grandeza mais átomos rotulados (classe mais abundante).

As configurações utilizadas para obter o melhor modelo com o conjunto de treinamento "Lig-qRankDB-SP-1.5-2.2 C347CA56" foram: a imagem qRankMask_5_75_95; um tamanho total de *batch* igual a 16 entradas (sem acúmulo de gradiente); a rede profunda MinkUNet34C_CONVATROUS_HYBRID; o otimizador SGD com parâmetros $momentum = 0.9$ e $dampening = 0.1$; uma taxa de aprendizado constante igual a 2^{-8}; uma taxa de rotação das imagens de treino de $R = 50\%$; e a função de perda wSL. O peso na função wSL foi igual a 1 para a classe de ruído de fundo, 500 para as classes C3, C4 e C7 minoritárias e 5 para as demais classes.

Todos os treinamentos da validação cruzada "k-fold" foram feitos com essa mesma configuração mudando apenas as entradas utilizadas para treino, validação e teste. Esses treinamentos foram chamados de Treinamento 1 até Treinamento 13, correspondentes a $k = 1$ até $k = 13$, e foram executados até 200 épocas. As configurações utilizadas nesses treinamentos estão detalhadas na Tabela 3.7. Os resultados de acurácia mIoU e da função de perda do treino, validação e teste final de todos os treinamentos da validação cruzada são apresentados na Tabela 3.8 para comparação. A matriz de confusão do teste do

Treinamento 1, que será utilizado como referência nas próximas análises, é apresentada na Tabela 3.9.

Tabela 3.7: Configurações dos treinamentos da validação cruzada "k-fold" para o conjunto "Lig-qRankDB-SP-1.5-2.2 C347CA56".

Configuração	Valor
Rede	MinkUNet34C_CONVATROUS_HYBRID
Tipo de Imagem	qRankMask_5_75_95
Otimizador	SGD
Parâmetros do Otimizador	$momentum = 0.9$ e $dampening = 0.1$
Taxa de Aprendizado	2^{-8}
Função de Perda (Loss)	wSL
Peso da Loss	1-500-500-500-5-5-5-5-5
Taxa de Rotação	50%
Tamanho Total do *Batch*	16
Número de Acúmulo de Gradiente	1
Tipo de Normalização	BN

Tabela 3.8: Resultado de acurácia do treino, validação e teste do conjunto de treinamentos da validação cruzada "k-fold" para o conjunto "Lig-qRankDB-SP-1.5-2.2 C347CA56"

	Treino		Validação		Teste		Teste IoU								
Nome	**Loss**	**mIoU**	**Loss**	**mIoU**	**Loss**	**mIoU**	**Background**	**Átomo**	**C3**	**C4**	**C7**	**C5**	**CA5**	**C6**	**CA6**
Treinamento 1	5.95	51.7	5.97	51.8	6.02	50.3	86.4	59.5	15.1	23.5	26.3	64.0	64.3	51.0	62.7
Treinamento 2	5.96	52.2	5.84	54.9	5.87	53.2	86.6	59.9	20.9	39.4	26.6	65.5	64.4	52.8	62.9
Treinamento 3	5.96	52.4	5.83	52.2	5.94	49.6	86.6	59.0	12.9	9.2	38.1	64.0	61.6	**53.9**	61.0
Treinamento 4	5.96	52.2	5.86	49.6	5.95	49.9	86.4	59.0	20.6	20.2	18.6	**66.1**	65.3	50.6	62.6
Treinamento 5	5.99	52.5	5.86	49.0	6.06	48.3	86.2	58.9	21.6	9.8	16.4	64.1	64.9	50.7	62.1
Treinamento 6	5.88	53.2	5.77	51.3	5.86	49.1	86.5	60.0	16.1	0.0	32.6	65.2	65.6	51.8	**63.8**
Treinamento 7	5.96	52.1	5.80	50.1	5.99	49.9	86.4	58.8	19.8	8.0	39.3	62.5	62.5	49.8	61.7
Treinamento 8	5.86	53.4	5.71	52.8	5.81	49.8	**86.8**	60.0	21.6	10.6	26.0	64.1	63.7	52.5	63.2
Treinamento 9	5.87	52.7	5.86	48.6	5.86	48.5	86.7	**60.1**	15.9	8.3	23.9	64.7	64.2	50.6	62.2
Treinamento 10	5.97	52.2	5.99	50.9	5.84	50.2	86.6	59.4	23.1	7.2	29.5	66.0	64.0	52.2	63.5
Treinamento 11	5.98	52.1	5.84	49.4	5.73	50.5	86.4	59.3	21.0	16.9	29.1	64.0	64.1	51.8	62.3
Treinamento 12	5.98	51.7	5.90	52.2	5.90	50.6	86.5	58.7	22.8	23.4	22.1	63.9	64.4	50.8	62.4
Treinamento 13	5.98	51.8	5.85	52.0	5.79	**56.6**	86.5	58.9	**32.3**	**33.5**	**51.5**	64.3	**65.9**	53.3	63.5
VC k-fold	**5.95**	**52.33**	**5.85**	**51.13**	**5.89**	**50.50**	**86.51**	**59.36**	**20.26**	**16.27**	**29.15**	**64.50**	**64.24**	**51.68**	**62.60**

Tabela 3.9: Matriz de confusão do Teste do Treinamento 1 com o Conjunto de treinamento "Lig-qRankDB-SP-1.5-2.2 C347CA56" e $k = 1$

	Background	**Átomo**	**C5**	**CA5**	**C6**	**CA6**	**C3**	**C4**	**C7**
Background	**86.37**	3.56	0.26	0.22	0.47	0.74	0.00	0.00	0.01
Átomo	21.89	**59.50**	0.27	0.13	0.52	0.60	0.00	0.00	0.03
C5	13.48	3.80	**64.04**	2.27	2.21	1.60	0.00	0.01	0.00
CA5	11.50	3.48	0.91	**64.29**	0.43	4.30	0.00	0.00	0.00
C6	18.59	5.63	0.39	0.36	**51.04**	7.16	0.00	0.00	0.01
CA6	13.20	3.82	0.22	1.09	1.20	**62.74**	0.00	0.00	0.01
C3	37.09	41.30	0.00	0.00	1.71	0.06	**15.10**	0.00	0.00
C4	30.91	10.45	28.03	0.26	3.48	0.12	0.00	**23.51**	0.00
C7	18.85	7.09	0.00	0.31	18.21	9.03	0.00	0.00	**26.29**

A Tabela 3.7 mostra que todos os treinamentos da validação cruzada tiveram um desempenho muito semelhante e a acurácia da validação cruzada "k-fold" para esse conjunto

foi de 50.50%. O modelo obtido com o Treinamento 13, referente a $k = 13$, teve o melhor resultado para segmentação semântica das classes do "Vocabulário de Átomos Genéricos e Ciclos C347CA56" com acurácia mIoU no teste igual a 56.6%. Essa consistência no resultado da validação cruzada evidencia que o Algoritmo 6 fez uma partição equitativa das entradas quanto a suas características entre os 13 subconjuntos.

Todos esses treinamentos tiveram curvas de aprendizado comportadas para as classes majoritárias e com boa convergência a partir de 10 épocas, mas ainda degradado para as classes minoritárias (C3, C4 e C7) que apenas começam a aumentar sua acurácia a partir de 50 épocas (a curva do Treinamento 1 será apresentada nos próximos conjuntos de treinamento).

A matriz de confusão na Tabela 3.9 mostra que as classes de ciclos aromáticos e ciclo de tamanho 5 foram as que apresentaram as melhores acurácias, com IoU igual a 64% para a classe C5, 64.3% para a classe CA5 e 62.7% para a classe CA6. Esse resultado fortalece a hipótese de que a estrutura planar dos ciclos aromáticos auxiliaria no seu aprendizado pelo modelo. Além disso, a confusão de 4.3% da classe CA5 com a classe CA6 também indica que a planaridade dessas estruturas é entendida pelo modelo. Nessa mesma linha de raciocínio, a rigidez de ciclos menores leva a menos mobilidade nos seus átomos, e logo, a uma imagem na densidade com mais qualidade e menos ruído. Esse fato levou à hipótese de que a rigidez de ciclos menores auxiliaria nos seus aprendizados e é o que acontece para a classe C5, o menor ciclo majoritário. Esse comportamento não é observado para os ciclos menores de tamanho 3 e 4, a baixa abundância dessas classes pode explicar seu lento e baixo aprendizado. A classe C6 teve a menor acurácia entre as classes cíclicas com alta abundância, provavelmente isso se deve à grande confusão entre a classe C7 e a classe C6 igual a 18.2% do total de pontos da classe C7. Além disso, os ciclos de tamanho 6 não aromáticos apresentam uma maior mobilidade, assim como os ciclos de tamanho 7, e isso pode ter impactado negativamente a acurácia dessas classes. A classe minoritária C3, apesar de ser a mais abundante entre os ciclos minoritários, também é a que possui a menor acurácia. Essa classe possui a maior confusão com a classe dos Átomos fora de ciclo e isso diminui sua acurácia em termos de IoU. Essa confusão provavelmente se deve ao pequeno volume desse ciclo e sua imagem correspondente na densidade ser pouco característica em comparação com a imagem de Átomos fora de ciclos. A classe C4 é a mais minoritária e ainda assim teve resultados melhores que a classe C3. A confusão entre a classe C4 e a C5 é de 28%, esse erro também acontece entre a classe C7 e a C6 e CA6 iguais a 18.2% e 9%, respectivamente. A proximidade do volume e formato dessas classes na densidade eletrônica pode explicar essa confusão. Ainda assim, a classe C7 é a classe minoritária com a melhor acurácia igual a 26.3%.

Todas as classes possuem uma confusão maior do que 11.5% com a classe do ruído de fundo, essa característica chama a atenção para um provável erro na borda da região do ligante ou devido a diferenças entre a conformação predita e a esperada. A baixa confusão entre o ruído de fundo e as demais classes fortalece essa hipótese, pois todos os pontos que deveriam ser ruído de fundo estão sendo preditos como tal, e pontos de outras classes estão sendo confundidos com ruído de fundo. Ou seja, o modelo está predizendo a classe de ruído de fundo onde deveriam estar outras classes mas não o contrário, e o mais provável é que isso aconteça com os pontos menos intensos da borda da esfera

dos átomos. Um raio de esfera atômica muito grande também poderia levar a uma alta confusão entre as demais classes e o ruído de fundo. Quatro exemplos do conjunto de teste foram selecionados para mostrar a distribuição do erro na predição de suas imagens e são apresentados na Figura 3.7.

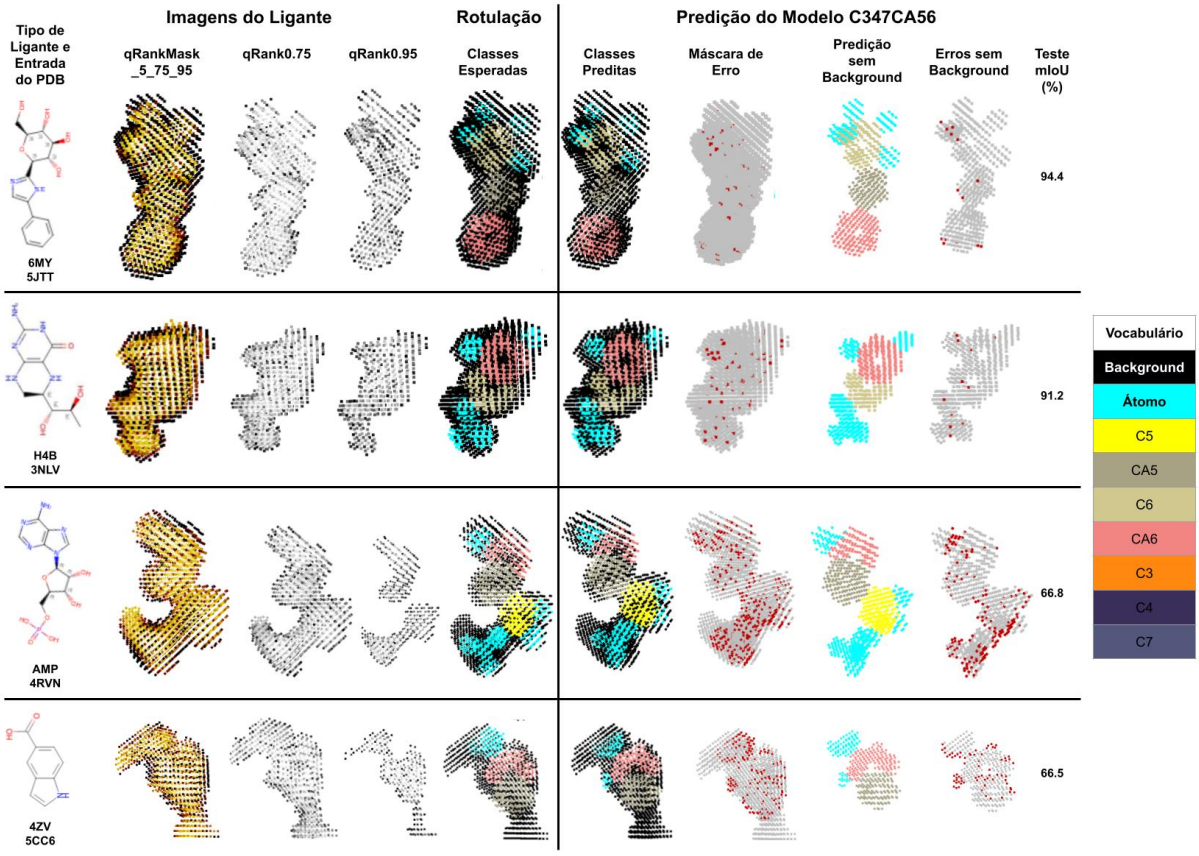

Figura 3.7: Quatro exemplos de predições do conjunto de teste "Lig-qRankDB-SP-1.5-2.2" para o "Vocabulário de Átomos Genéricos e Ciclos C347CA56" feitas com o modelo obtido no Treinamento 1. Os seguintes tipos de ligantes foram selecionados: 6MY, H4B, AMP, e 4ZT. Estes ligantes pertencem as seguintes entradas do PDB, na mesma ordem: 5JTT, 3NLV, 4RVN, e 5CC6. Cujas resoluções são, nessa ordem: 1.85 Å, 2.1 Å, 2.2 Å e 2.1 Å. Para cada ligante é apresentado sua estrutura 2D, suas imagens qRankMask_5_75_95, qRank0.75 e qRank0.95, sua imagem qRankMask_5_75_95 rotulada, e sua predição com o modelo obtido no Treinamento 1. Para cada predição também é apresentada a máscara de erro da imagem do ligante, com os pontos que receberam uma predição errada coloridos em vermelho e os demais em cinza, e a imagem da predição e da máscara de erro de cada ligante sem os pontos preditos como Background. A acurácia média mIoU do teste do Treinamentos 1 para cada ligante é apresentada na última coluna da imagem.

Os quatro ligantes do conjunto de teste mostrados na Figura 3.7 obtiveram predições muito próximas ao esperado. Os ligantes 6MY e H4B, com acurácia mIoU acima de 90%, apresentaram erros de predição apenas na borda da região do ligante e a comparação visual com as classes esperadas indica que esses erros são devido a uma conformação predita um pouco diferente da esperada. Os ligantes AMP e 4ZV, com acurácias muito próximas uma da outra e abaixo de 70%, apresentaram em suas imagens mais ruído de fundo próximo as regiões flexíveis dessas moléculas, provavelmente devido a outras

conformações ocupadas no cristal. No ligante AMP, podemos observar uma imagem mais borrada próxima ao átomo de fósforo (P), devido provavelmente a flexibilidade dessa região. E no ligante 4ZV é possível observar uma imagem borrada próxima aos átomos de oxigênio, região flexível, e também abaixo da região rotulada do ligante, indicando um provável movimento de translação da molécula inteira ou ruído de uma molécula adjacente. No ligante 4ZV, um átomo foi predito errado onde era esperado ruído de fundo, e os demais erros estão na borda da região predita. No ligante AMP, todos os erros parecem ser devido a uma conformação predita diferente da esperada, o que fica bem evidente para os átomos ligados ao ciclo de tamanho cinco (C5). Esses quatro exemplos de predições mostrados corroboram com a suposição de erros na borda da região do ligante e também mostram que acurácias próximas a 65% apresentaram resultados muito bons. Além disso, os dois exemplos com as menores acurácias também mostraram a capacidade do modelo em remover ruído de fundo.

Os erros de predições mostrados nos exemplos da Figura 3.7 indicam a necessidade de um pós-processamento no resultado da predição do modelo para remover pequenos erros locais cujo volume está fora do esperado, como o átomo extra predito para o ligante 4ZV. Esse pós-processamento poderia, por exemplo, identificar agrupamentos de pontos da mesma classe e pelo volume de cada agrupamento identificar possíveis classes inviáveis. Para os agrupamentos com classes inviáveis, o pós-processamento poderia atribuir aos pontos desses agrupamentos a classe do ponto mais próximo pertencente a um agrupamento viável.

O resultado de acurácia do teste do Treinamento 1 também foi utilizado para avaliar a relação entre algumas características das entradas, como as variáveis utilizadas nos filtros de qualidade, e suas acurácias médias em termos de mIoU. A Figura 3.8 mostra sete gráficos da acurácia média em termos de mIoU de cada entrada do conjunto de teste pelas seguintes características das entradas: resolução, tamanho da imagem qRank0.95 (filtro de qualidade local), razão entre o B fator do ligante e da proteína (filtro de qualidade global), desvio padrão do B fator do ligante (filtro de qualidade global), ocupância mínima do ligante (filtro de qualidade global), B fator do ligante e B fator da proteína. Nesses gráficos foi realizada uma regressão linear simples e a curva ajustada junto com o coeficiente de correlação de Pearson (variável r) também aparecem nas imagens. A variável r ao quadrado é igual ao coeficiente de determinação, que é uma medida de ajuste de um modelo estatístico linear generalizado aos valores observados de uma variável aleatória. Essa analise permitiu verificar se alguma das variáveis selecionadas poderia auxiliar na remoção de entradas ruidosas que induzem um baixo desempenho do modelo.

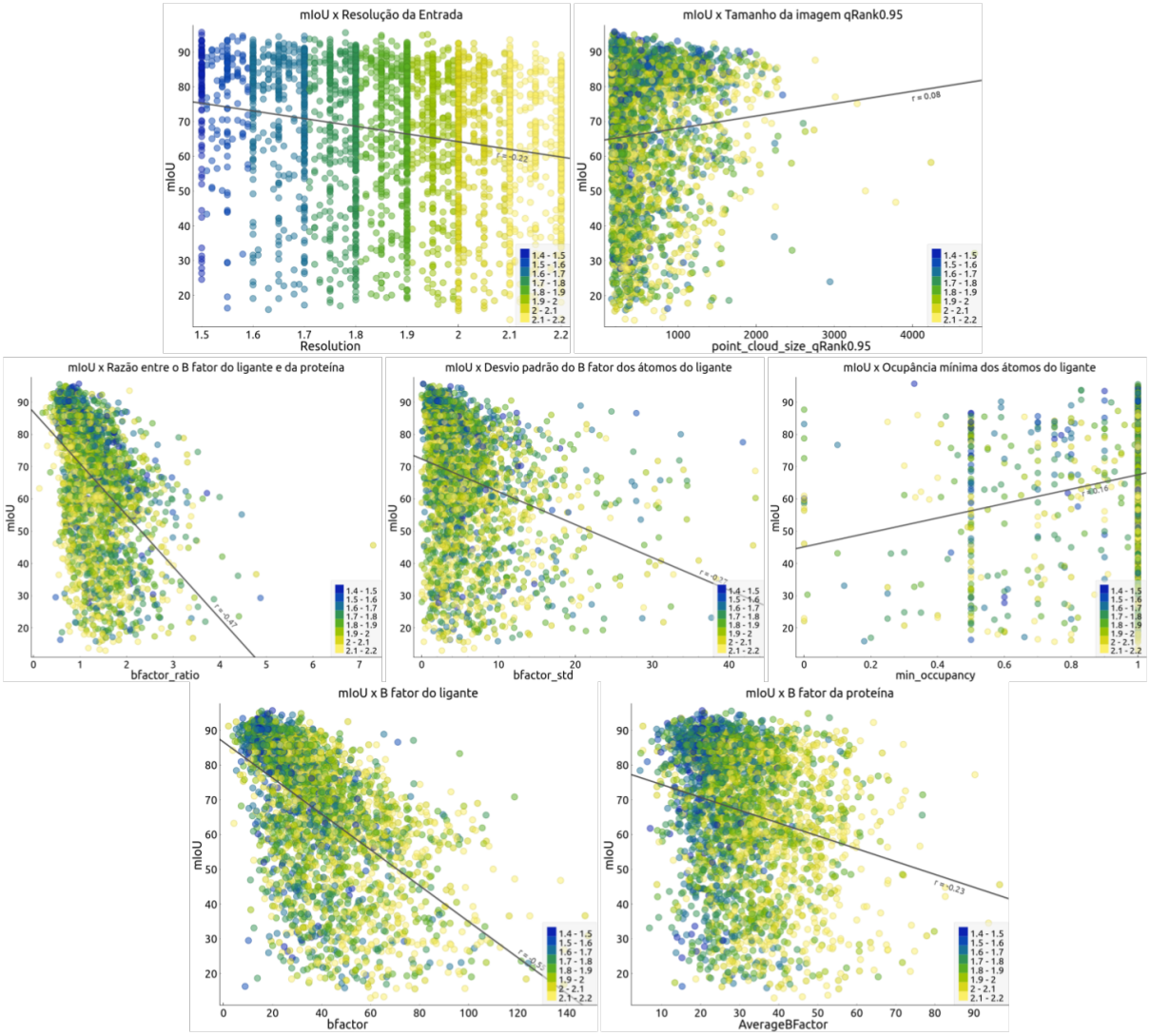

Figura 3.8: A acurácia média das entradas do conjunto de teste Lig-qRankDB-SP-1.5-2.2 C347CA56 do Treinamento 1 são mostradas versus sete características das entradas. Todos os gráficos foram coloridos de acordo com a resolução de cada entrada em intervalos de 0.1 Å. Foi realizada uma regressão linear simples em todos os gráficos e o coeficiente de correlação de Pearson, variável r, é mostrado junto com a linha ajustada pela regressão. A variável r ao quadrado é igual ao coeficiente de determinação.

Nos gráficos mostrados na Figura 3.8 nenhuma das características das entradas dos ligantes apresentaram uma boa correlação ou um coeficiente de determinação maior do que 30%, ou seja, nenhuma dessas variáveis explicam a variância da acurácia do modelo. Esse resultado mostra que os filtros de qualidade utilizados não afetam substancialmente a acurácia final do modelo, e logo, não utilizá-los pode ser benéfico para aumentar o conjunto de treinamento e a diversidade de entradas. O gráfico da resolução das entradas ajuda a perceber a relação entre a qualidade do dado e a acurácia do modelo. Esse gráfico mostra que as resoluções abaixo de 1.7 Å apresentaram acurácias médias maiores e com menos variação, enquanto as resoluções acima desse valor apresentaram mais acurácias abaixo de 60%. Essa análise mostra que a acurácia média é afetada negativamente a medida que a resolução aumenta, mas com muita variação. Isso corrobora com o fato dos

ruídos nos dados de cristalografia serem locais, e logo, entradas com uma resolução ruim ainda podem ter uma boa acurácia se a densidade residual do ligante estiver bem definida e pouco ruidosa (mas nem sempre esse é o caso).

O modelo obtido com o Treinamento 13, referente a $k = 13$, foi considerado o melhor resultado deste trabalho por ser capaz de segmentar subestruturas químicas maiores e com mais informação estrutural em comparação com as outras modelagens propostas e com acurácias IoU por classe próximas ou maiores do que 60%. Uma melhor escolha dos pesos das classes na função de perda poderia contribuir para a convergência das classes com acurácia abaixo de 60% e melhora da acurácia mIoU no teste deste modelo.

Análises Sistemáticas da Melhor Configuração para o Conjunto Lig-qRankDB-SP-1.5-2.2 C347CA56

Com o intuito de avaliar melhor o impacto de cada configuração no desempenho da validação cruzada "k-fold" do Treinamento 1 com o conjunto de treinamento "Lig-qRankDB-SP-1.5-2.2 C347CA56", foram realizadas três análises sistemáticas para verificar o impacto na acurácia final do modelo devido a: diferentes tamanhos de *batch* com e sem acúmulo de gradiente; diferentes tipos de imagem; e a cada hiperparâmetro da rede na configuração final. Esses 3 conjuntos de treinamento da análise sistemática foram feitos com $k = 1$, antes do resultado da validação cruzada "k-fold", e serão apresentados separadamente, na seguinte ordem. Primeiro serão apresentados 10 treinamentos realizados mantendo-se a configuração do Treinamento 1 e alterando-se apenas o tamanho do *batch* e o número de acúmulos de gradiente. Depois serão apresentados 4 treinamentos alterando apenas o tipo da imagem utilizada. E por último serão apresentados 6 treinamentos mantendo o tamanho do *batch* e o tipo de imagem do Treinamento 1 - Lig-qRankDB-SP-1.5-2.2 C347CA56 e alterando um hiperparâmetro ou configuração por vez. Em todos esses conjuntos de treinamentos o Treinamento 1 da validação cruzada *k-fold* será mantido como o Treinamento 1 de cada conjunto da análise sistemática.

A configuração de todos os 10 treinamentos para análise do melhor tamanho de *batch* e número de acúmulos de gradiente é apresentada na Tabela 3.10. O tipo de camada de normalização também foi modificado, quando há uso da técnica de acúmulo de gradiente a normalização baseada na instância (IN, do inglês *instance normalization*) é utilizada, caso contrário a normalização baseada no batch (BN, do inglês *batch normalization*) é usada. Apenas no Treinamento 3 foi usada a IN sem acúmulo de gradiente e no Treinamento 4 foi usada a BN com acúmulo de gradiente. O Treinamento 1 é o modelo com o melhor resultado até o momento na configuração final, com $k = 1$, tamanho de *batch* igual a 16 e BN (resultado da Tabela 3.9).

As curvas de aprendizado da validação desse primeiro conjunto de treinamentos da análise sistemática são apresentadas na Figura 3.9. As curvas foram nomeadas com a letra 'V' seguida do índice do treinamento, a variável b definindo o tamanho total do *batch* seguido do tipo de normalização usada ('IN' ou 'BN') e a variável *acum* definindo o número de acúmulos de gradiente, caso este tenha sido usado no treinamento (i.e. $acum > 1$). As curvas IoU da validação por classe foram omitidas para simplificar a visualização, apenas seus resultados finais no teste serão mostrados. A Tabela 3.11 apresenta os resultados de

acurácia finais dos testes desses treinamentos em termos de IoU por classe e mIoU, assim como o valor da função de perda de treino e validação.

Tabela 3.10: Configurações dos treinamentos da análise sistemática do tamanho total de batch, com acúmulo de gradiente e tipos de normalização diferentes para o conjunto de treinamento Lig-qRankDB-SP-1.5-2.2 C347CA56.

Nome	Tamanho Total de Batch	Quantidade de Acúmulo de Gradiente	Tipo de Normalização	Nome da Curva
Treinamento 1	16	1	BN	V1 b=16 BN
Treinamento 2	8	1	BN	V2 b=8 BN
Treinamento 3	16	1	IN	V3 b=16 IN
Treinamento 4	16	2	BN	V4 b=16 BN acum=2
Treinamento 5	32	1	BN	V5 b=32 BN
Treinamento 6	32	2	IN	V6 b=32 IN acum=2
Treinamento 7	64	1	BN	V7 b=64 BN
Treinamento 8	64	4	IN	V8 b=64 IN acum=4
Treinamento 9	128	8	IN	V9 b=128 IN acum=8
Treinamento 10	256	4	IN	V10 b=256 IN acum=4

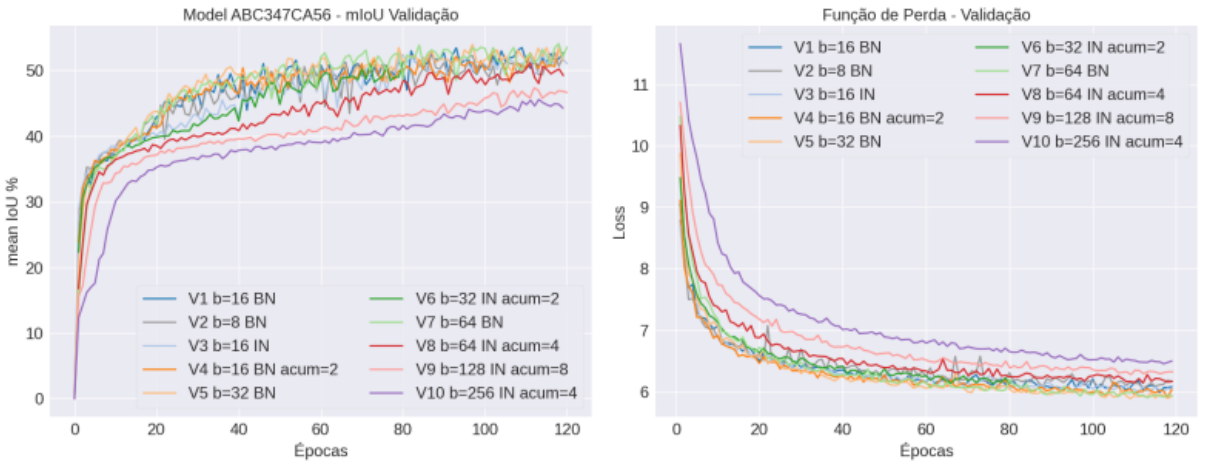

Figura 3.9: Curvas de aprendizado da análise sistemática do tamanho do *batch* da validação dos dez treinamentos com o conjunto de dados "Lig-qRankDB-SP-1.5-2.2" para o "Vocabulário de Átomos Genéricos e Ciclos C347CA56". Curvas da acurácia mIoU e da função de perda da validação. As curvas de IoU por classe foram omitidas para focar a comparação do impacto do tipo de normalização, do tamanho do batch e do uso de acúmulo de gradiente apenas na acurácia média da validação. As curvas de treino também foram omitidas para facilitar a visualização dos resultados da validação.

Os resultados apresentados na Figura 3.9 e na Tabela 3.11 mostram que os treinamentos com tamanhos totais de *batch* iguais a 8, 16, 32 e 64 sem acúmulo de gradiente tiveram os melhores resultados. O Treinamento 5 com tamanho total de *batch* igual a 32 e tipo de normalização BN teve o melhor resultado no teste mIoU. Os Treinamentos 9 e 10 com tamanhos totais de *batch* maiores e iguais a 128 e 256 apresentaram piora no aprendizado. Os Treinamentos 6 e 8 com tamanho total de *batch* igual a 32 e 64, respectivamente, evidenciam que o uso do acúmulo de gradiente junto com o tipo de normalização IN acaba prejudicando o aprendizado quando comparado com os Treinamentos 5 e 7, respectivamente. O impacto negativo é pior para os *batchs* maiores dos Treinamentos 9 e 10,

Tabela 3.11: Resultados de acurácia e da função de perda para os treinamentos da análise sistemática do tamanho total de *batch*

	Treino		Validação		Teste		Teste IoU							
Nome	**Loss**	**mIoU**	**Loss**	**mIoU**	**Loss**	**mIoU**	**Background**	**Átomo**	**C3**	**C4**	**C7**	**C5**	**CA5**	**C6**	**CA6**
Treinamento 1	6.10	50.7	6.04	52.7	6.14	48.6	86.3	58.6	18.9	28.6	9.1	62.2	62.4	49.0	62.1
Treinamento 2	6.25	47.7	6.06	51.0	6.13	48.4	86.3	58.6	19.2	19.0	16.5	61.8	63.0	49.1	62.0
Treinamento 3	5.94	51.6	5.99	51.0	6.10	49.4	86.4	58.9	15.2	34.8	11.8	62.6	62.1	49.9	62.0
Treinamento 4	6.10	50.5	6.01	51.0	6.11	48.9	86.3	58.4	18.4	34.6	10.0	61.5	62.2	47.9	60.6
Treinamento 5	5.90	53.4	5.92	53.0	6.01	**51.5**	86.3	58.9	20.7	35.6	29.0	62.3	61.0	48.6	61.8
Treinamento 6	6.04	50.4	6.13	51.2	6.20	49.5	86.2	58.3	17.2	28.3	23.2	61.7	60.4	48.7	61.3
Treinamento 7	5.80	54.8	5.89	53.5	5.99	50.2	86.3	59.3	14.8	33.7	22.7	61.5	62.2	49.9	62.4
Treinamento 8	5.81	52.6	6.13	49.7	6.20	48.8	86.2	58.3	15.3	31.6	12.1	61.7	61.7	48.9	61.1
Treinamento 9	5.98	49.9	6.32	46.6	6.39	45.3	85.9	57.4	8.7	17.1	13.8	59.8	60.3	47.0	59.1
Treinamento 10	6.15	49.4	6.49	44.3	6.51	44.5	85.6	56.9	13.8	17.1	10.0	56.6	57.1	45.1	58.2

mas não alterou o aprendizado do Treinamento 4 com tamanho total de *batch* 16. Esse impacto negativo pode ser visto no achatamento das curvas de mIoU da validação V3, V6, V8, V9 e V10 no início da sua convergência após aproximadamente 20 épocas e uma convergência mais lenta para o Treinamento 10 com um atraso no início do aprendizado.

Todo os treinamentos apresentaram resultados muito parecidos no teste mIoU, com exceção dos Treinamentos 9 e 10 que foram piores. Em relação a acurácia IoU do teste por classe é perceptível uma maior variação da acurácia das classes minoritárias C3, C4 e C7 entre os treinamentos, um indicativo de baixa convergência, enquanto para as demais classes essa variação não passa de 3% IoU, excluindo os Treinamentos 9 e 10. O Treinamento 1 foi considerado o melhor resultado por utilizar um tamanho total de *batch* pequeno, e logo, possibilitar o treinamento em computadores com menos capacidade de processamento e memória sem perda de desempenho. Nas próximas análises o tamanho total de *batch* do Treinamento 1, igual a 16, sem acúmulo de gradiente e tipo de normalização BN, foi escolhido para ser utilizado seguindo os resultados desse primeiro conjunto de treinamentos da análise sistemática.

O próximo conjunto de treinamentos da análise sistemática foi feito para avaliar o impacto do uso de diferentes tipos de imagem dos ligantes no desempenho do modelo. Foi decidido comparar os seguintes tipos de imagens: qRank0.75, qRankMask, qRankMask_5_7_9 e qRankMask_5_75_95. A configuração dos 4 treinamentos dessa análise sistemática são apresentados na Tabela 3.12. O Treinamento 1 representa o melhor modelo com a imagem qRankMask_5_75_95 para referência na comparação entre as análises. O Treinamento 2 utiliza a imagem qRank0.75. O Treinamento 3 utiliza a imagem qRankMask. O Treinamento 4 utiliza a imagem qRankMask_5_7_9. As curvas de aprendizado da validação desse segundo conjunto de treinamentos da análise sistemática são apresentadas na Figura 3.10. As curvas foram nomeadas com a letra 'V' seguida do índice do treinamento e o tipo de imagem utilizada. As curvas IoU da validação por classe foram omitidas para simplificar a visualização, apenas seus resultados finais no teste serão mostrados. A Tabela 3.13 apresenta os resultados de acurácia finais dos testes desses treinamentos em termos de IoU por classe e mIoU, assim como o valor da função de perda de treino e validação.

Tabela 3.12: Configurações dos treinamentos da análise sistemática do tipo de imagem dos ligantes para o conjunto de treinamento Lig-qRankDB-SP-1.5-2.2 C347CA56

Nome	Tipo de Imagem	Nome da Curva
Treinamento 1	qRankMask_5_75_95	V1 qRankMask_5_75_95
Treinamento 2	qRank0.75	V2 qRank0.75
Treinamento 3	qRankMask	V3 qRankMask
Treinamento 4	qRankMask_5_7_9	V4 qRankMask_5_7_9

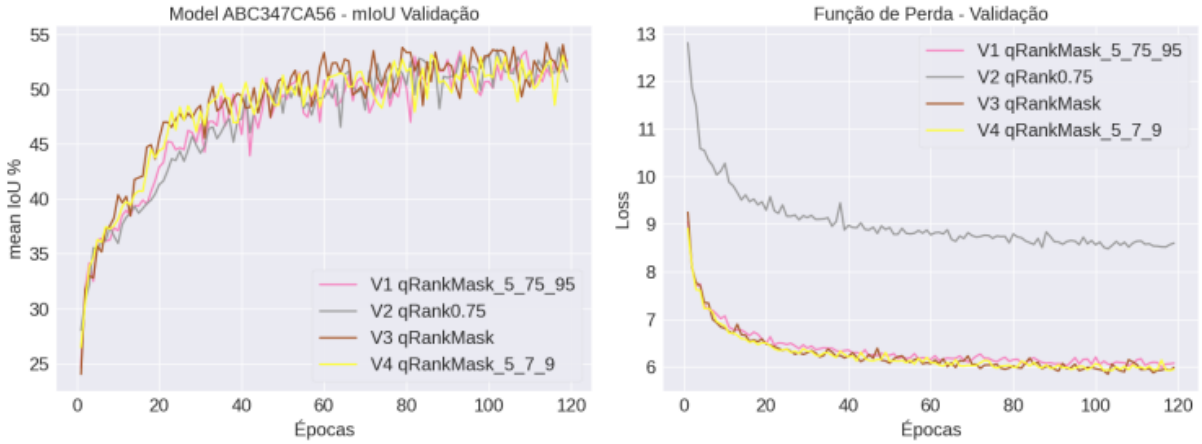

Figura 3.10: Curvas de aprendizado da análise sistemática do tipo da imagem dos ligantes com os resultados da validação de quatro treinamentos com o conjunto de dado "Lig-qRankDB-SP-1.5-2.2" para o "Vocabulário de Átomos Genéricos e Ciclos C347CA56". Curvas da acurácia mIoU e da função de perda da validação. As curvas de IoU por classe foram omitidas para simplificar a comparação do tipo da imagem usando apenas a acurácia média mIoU da validação. As curvas de treino também foram omitidas para facilitar a visualização dos resultados da validação.

Tabela 3.13: Resultados de acurácia e da função de perda para os treinamentos da análise sistemática do tipo de imagem dos ligantes

	Treino		Validação		Teste		Teste IoU								
Nome	Loss	mIoU	Loss	mIoU	Loss	mIoU	Background	Átomo	C3	C4	C7	C5	CA5	C6	CA6
Treinamento 1	6.10	50.7	6.04	52.7	6.14	48.6	86.3	58.6	18.9	28.6	9.1	62.2	62.4	49.0	62.1
Treinamento 2	8.74	50.3	8.54	53.2	8.71	49.4	76.8	**61.1**	19.6	**36.2**	17.7	61.3	61.3	48.6	**62.2**
Treinamento 3	5.96	52.5	5.86	51.9	5.97	**50.3**	**86.5**	59.2	17.3	22.0	**30.3**	62.9	62.8	**49.1**	62.1
Treinamento 4	6.00	51.4	5.94	51.5	5.99	50.1	**86.5**	58.6	**20.**	28.8	20.7	**63.0**	**63.3**	48.4	61.5

A acurácia mIoU da validação dos diferentes tipo de imagem apresentou valores muito similares na média. Na Tabela 3.13 do resultado do teste desses modelos é possível observar que para o Treinamento 2 com a imagem qRank0.75 sua função de perda tem valores bem maiores do que os demais treinamentos e a acurácia IoU da classe do background é cerca de 10% menor. Além disso, a classe do átomo com a imagem qRank0.75 teve um aumento de acurácia IoU de quase 3% de IoU e as demais classes tiveram pouca variação. Para as demais imagens, não existe uma grande diferença nos seus resultados. O resultado dessa analise sistemática evidencia que a forma que os diferentes tipos de

imagem foram criados não afetou o desempenho dos modelos com o conjunto de treinamento "Lig-qRankDB-SP-1.5-2.2 C347CA56", com exceção de imagens sem a máscara do ligante que possuem menos pontos da classe background, e logo, apresentam uma menor acurácia nessa classe. Além disso, a imagem qRank0.75 possui pontos da classe background mais próximos da borda da região do ligante, e neste caso em que apenas esses pontos de background estão presentes é observada uma queda na acurácia dessa classe e isso evidencia que o erro correspondente está próximo à borda da região do ligante. A imagem qRankMask_5_75_95 foi escolhida para ser utilizada nos treinamentos finais.

O último conjunto de treinamentos da análise sistemática avalia o impacto dos principais hiperparâmetros e configurações no desempenho do modelo final. As configurações de todos os 6 treinamentos dessa análise são apresentadas na Tabela 3.14. Todos os treinamentos foram feitos com um tamanho total de *batch* igual a 16, sem acúmulo de gradiente, tipo de normalização BN e tipo de imagem qRankMask_5_75_95, seguindo as configurações do Treinamento 1 - Lig-qRankDB-SP-1.5-2.2 C347CA56. O Treinamento 1 é o modelo de referência com o melhor resultado até o momento na configuração final (resultado da Tabela 3.9). O Treinamento 2 é o modelo com otimizador Adam e parâmetros $beta1 = 0.9$ e $beta2 = 0.999$, chamado de 'Opt Adam'. O Treinamento 3 é o modelo com a função de perda wCE, com os mesmos pesos atribuídos à função wSL no Treinamento 1. O Treinamento 4 é o modelo com a rede MinkUNet34C, sem dilatação, chamado de 'No Conv Atrous'. O Treinamento 5 é o modelo sem rotação das imagens de treino, R=0%, chamado de 'No Rotation'. O Treinamento 6 é o modelo sem peso na função de perda SL, chamado de 'No Loss Weight'. As curvas de aprendizado do treino e da validação desse terceiro conjunto de treinamentos da análise sistemática são apresentadas na Figura 3.11. Na função que desenha as curvas de aprendizado, foi implementado um método de smoothing para suavização da variação dos treinamentos por época, baseado no método da Média Móvel Exponencial com a taxa de smoothing variando entre 0 e 1. A taxa de smoothing utilizada nessas curvas é apresentada na descrição da figura. A Tabela 3.15 apresenta os resultados de acurácia finais dos testes desses treinamentos em termos de IoU por classe e mIoU, assim como o valor da função de perda de treino e validação.

Tabela 3.14: Configurações dos treinamentos da análise sistemática dos hiperparâmetros para o conjunto de treinamento Lig-qRankDB-SP-1.5-2.2 C347CA56

Nome	Rede	Otimizador	Função de Perda (Loss)	Peso da Loss	Taxa de Rotação (%)	Nome da Curva
Treinamento 1	MinkUNet34C_CONVATROUS_HYBRID	SGD	wSL	Sim	50	V1 Referência
Treinamento 2	MinkUNet34C_CONVATROUS_HYBRID	**Adam**	wSL	Sim	50	V2 Opt Adam
Treinamento 3	MinkUNet34C_CONVATROUS_HYBRID	SGD	**wCE**	Sim	50	V3 Loss CE
Treinamento 4	**MinkUNet34C**	SGD	wSL	Sim	50	V4 No Conv Atrous
Treinamento 5	MinkUNet34C_CONVATROUS_HYBRID	SGD	wSL	Sim	**0**	V5 No Rotation
Treinamento 6	MinkUNet34C_CONVATROUS_HYBRID	SGD	**SL**	**Não**	50	V6 No Loss Weight

Tabela 3.15: Resultados de acurácia e da função de perda para os treinamentos da análise sistemática dos hiperparâmetros

	Treino		Validação		Teste		Teste IoU								
Nome	**Loss**	**mIoU**	**Loss**	**mIoU**	**Loss**	**mIoU**	**Background**	**Átomo**	**C3**	**C4**	**C7**	**C5**	**CA5**	**C6**	**CA6**
Treinamento 1	6.10	50.7	6.04	52.7	6.14	**48.6**	**86.3**	**58.6**	**18.9**	28.6	9.1	**62.2**	**62.4**	**49.0**	**62.1**
Treinamento 2	7.89	32.9	7.66	32.5	7.75	32.2	84.0	51.5	0.0	0.0	0.0	39.9	38.6	28.8	46.8
Treinamento 3	0.60	38.0	0.61	39.8	0.63	40.1	76.3	50.2	5.3	7.2	19.9	51.7	55.1	42.7	52.6
Treinamento 4	6.18	50.0	6.33	47.5	6.42	46.6	85.8	57.6	12.6	34.0	6.5	58.3	59.8	46.0	59.2
Treinamento 5	4.75	66.5	6.36	49.8	6.46	**48.7**	85.8	57.1	15.4	**35.5**	**23.0**	58.5	58.4	46.1	58.4
Treinamento 6	6.00	42.4	6.00	42.1	6.09	41.7	**86.3**	57.9	0.0	0.0	0.0	61.7	60.0	48.9	60.1

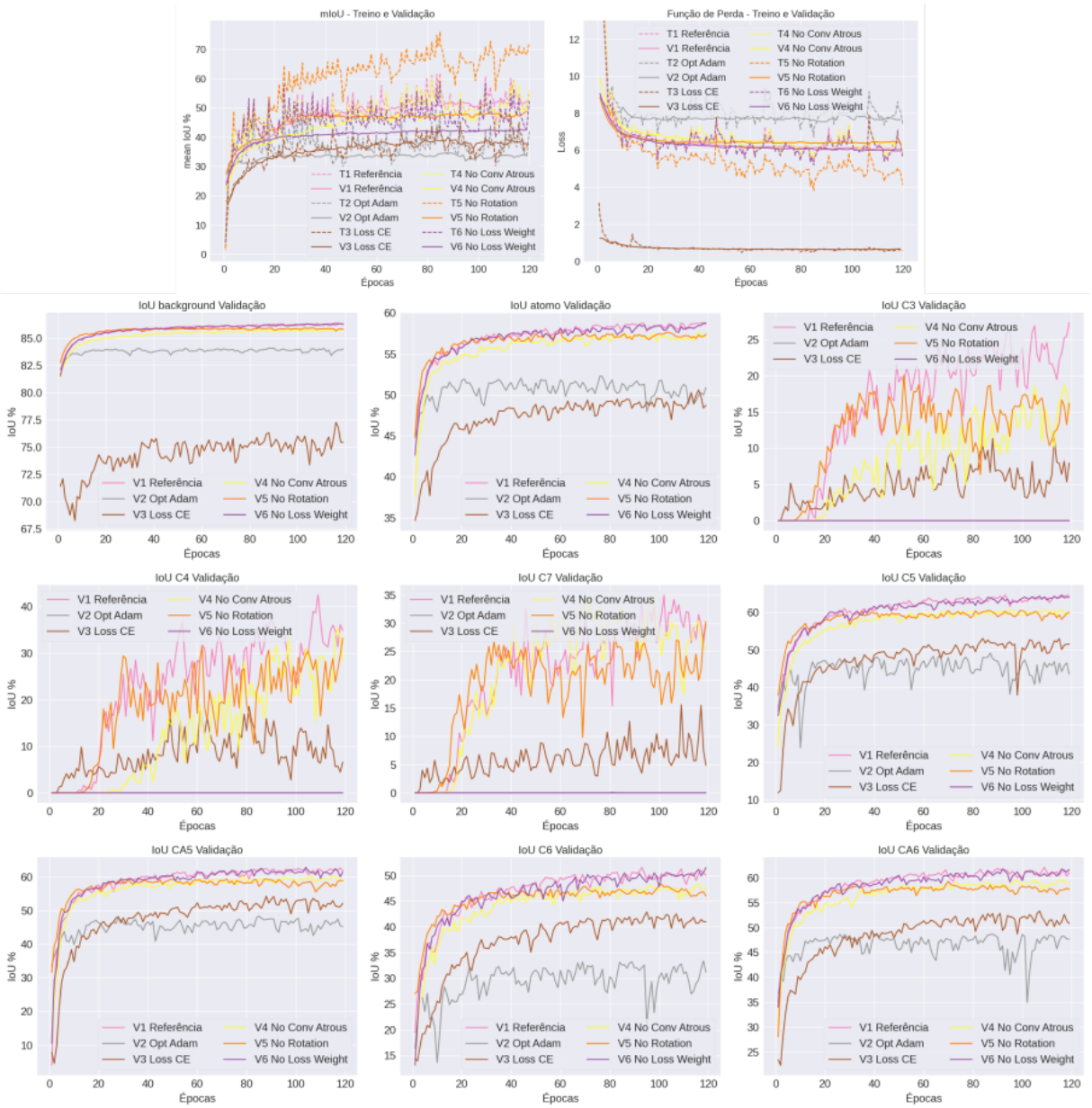

Figura 3.11: Curvas de aprendizado da análise sistemática dos hiperparâmetros da rede do treino e da validação dos seis treinamentos com o conjunto de dado "Lig-qRankDB-SP-1.5-2.2" para o "Vocabulário de Átomos Genéricos e Ciclos C347CA56". Curvas da acurácia IoU da validação por classe com smothing igual a 0.4. E curvas da acurácia mIoU e das funções de perda do treino e da validação com smothing igual a 0.6.

O Treinamento 1 manteve o melhor resultado nessa análise sistemática e demonstra o ganho na combinação de diversas configurações no desempenho final do modelo.

O Treinamento 2 com otimizador Adam teve o pior desempenho. Nas classes majoritárias "background" e "átomo", esse treinamento consegue ficar melhor do que o Treinamento 3 com a função de perda wCE. Nas demais classes, seu aprendizado tem o pior resultado, ficando com acurácia zerada nas classes minoritárias. É possível perceber uma estabilização nas curvas de aprendizado desse treinamento após 20 épocas, com acurácias IoU em torno de 20% menores do que as do Treinamento 1 em todas as classes, exceto nas classes do "background" em que a diferença diminui para apenas 2.3% e do "átomo" em que a diferença é de 7.1% (resultados na Tabela 3.15). Esse resultado evidencia o impacto do otimizador para classes menos abundantes e difíceis de convergir. Neste caso o otimizador SGD teve um resultado muito superior ao Adam.

O Treinamento 3 com a função de perda wCE e os mesmos pesos usados no Treinamento 1 referência, apresenta uma convergência mais lenta e se estabiliza em uma acurácia menor após cerca de 60 épocas. As classes majoritárias demoram mais épocas para iniciar sua convergência e atingem uma acurácia IoU cerca de 10% menor que o treinamento referência. E as classes minoritárias começam a convergir mais cedo, mas com uma menor inclinação nas suas curvas de aprendizado e atingem uma acurácia IoU de 10 à 20% menores que a referência. Esse treinamento apresentou uma acurácia mIoU no teste mIoU cerca de 8% menor do que a referência. Isso ilustra a robustez da função de perda wSL para lidar com possíveis ruídos na rotulação e auxiliar na convergência de classes minoritárias e difíceis de convergir.

O Treinamento 4 com a rede MinkUNet34C, sem dilatação, teve uma acurácia no teste em termos de mIoU cerca de 2% menor do que a referência. As classes cíclicas majoritárias apresentaram acurácia IoU cerca de 3% menor e as classes cíclicas menores e minoritárias (C3 e C4) apresentaram convergência mais lenta do que a referência. Apenas a classe C7 apresentou um comportamento bem similar à referência. Esse resultado corrobora para comprovar o benefício do uso da convolução dilatada para detectar padrões de objetos sem necessidade de diminuir a resolução da imagem durante sua passagem pela rede. Diferentes taxas de dilatação poderiam ser avaliadas aqui para chegar em um melhor desempenho do modelo.

O Treinamento 5 sem rotação aleatória das imagens do conjunto de treino apresenta uma menor diversidade no seu treinamento e isso impacta a duração do se aprendizado em termos de épocas. Esse modelo começa a apresentar sinais de *overfitting* bem cedo, a partir de 10 épocas é possível perceber que a função de perda da validação começa a se estagnar e a se distanciar da função de perda do treino, a qual continua diminuindo (Figura 3.11). Nesse momento, também é possível observar que a acurácia mIoU da validação começa a se estagnar e a acurácia mIoU do treino continua aumentando. O *overfitting* fica evidente a partir de 35 épocas, nesse momento é possível observar uma inflexão na curva da função de perda da validação característica de *overfitting*. A partir desse momento, a função de perda da validação começa a aumentar e se distanciar ainda mais da função de perda do treino, que continua diminuindo até uma diferença maior do que 2. Nas curvas IoU

de validação das classes, é possível perceber a partir de 35 épocas uma estabilização do aprendizado seguida de um início de degradação. Esse comportamento ilustra muito bem que o uso de rotação aleatória nas imagens do conjunto de treino adiciona diversidade ao treinamento e permite um treinamento mais longo sem *overfitting*. Diferentes taxas de rotação poderiam ser avaliadas aqui para definir o melhor valor a ser utilizado e verificar seu impacto na duração do treinamento.

O Treinamento 6 sem peso na função de perda SL tem um comportamento bem similar ao do melhor resultado referência, mas não converge nas classes minoritárias mantendo suas acurácias IoU zeradas. O aprendizado degradado das classes minoritárias diminui a acurácia média do modelo no teste em termos de mIoU a qual fica cerca de 7% menor do que a referência, mesmo com o aprendizado das demais classes não sendo muito impactado.

A representação visual do resultado de acurácia do teste desse segundo conjunto da análise sistemática apresentado na Tabela 3.15 permite entender o impacto de cada configuração no desempenho do Treinamento 1 referência, que teve os melhores resultados. É possível perceber que a combinação de diferentes configurações promove uma melhora não percebida quando cada configuração é analisada individualmente. Essa análise permite concluir que o otimizador SGD e a função de perda wSL foram fundamentais para melhorar a acurácia do modelo, junto com a rotação aleatória que permitiu treinamentos mais longos. O uso de peso na função de perda ajudou na convergência das classes minoritárias, mas não foi suficiente para lidar com o grande desbalanço entre as classes do "Vocabulário de Átomos Genéricos e Ciclos C347CA56". A convolução dilatada proporcionou uma pequena melhora na acurácia do modelo, diferentes taxas de dilatação deveriam ser avaliadas para verificar se é possível melhorar esse impacto. O mesmo deveria ser feito para escolha dos pesos das classes na função de perda.

Outras análises que poderiam ser feitas para melhor o desempenho do modelo referência são: refinar o valor dos parâmetros do otimizador; avaliar outros otimizadores; e refinar os parâmetros da função de perda. Neste momento não foi possível realizar essas comparações.

3.1.4 Modelo Final dos Vocabulários de Classes SP, Simplificados e por Tipo de Átomo

A melhor configuração para o conjunto de treinamento "Lig-qRankDB-SP-1.5-2.2" com o "Vocabulário de Átomos Genéricos e Ciclos C347CA56" foi utilizada para treinar novos modelos com os outros vocabulários propostos. Em alguns casos, um treinamento com tamanho total de *batch* igual a 128 também será mostrado para verificar se esta configuração é dependente do vocabulário utilizado. Os vocabulários avaliados foram: "Vocabulário de Classes SP", "Vocabulário de Classes SP e Ciclos C347CA56", "Vocabulário da Região do Ligante", "Vocabulário de Átomos e Ciclos Genéricos", "Vocabulário de Tipos de Átomos" e "Vocabulário de Tipos de Átomos Agrupados". Para cada vocabulário será apresentada a distribuição das suas classes no conjunto de dados "Lig-qRankDB-1.5-2.2", suas respectivas curvas de aprendizado da validação e do treino, e resultados de acurácia do teste dos modelos obtidos. Os mapeamentos aplicados para obter todos esses vocabulários foram apresentados nas Tabelas 2.2 e 2.3. Os vocabulários de classes SP e de tipos de átomos

com ciclos não serão apresentados devido a seus baixos desempenhos.

Os resultados dos treinamentos baseados no vocabulário SP sem simplificação serão apresentados juntos. Os Treinamentos 1 e 3 foram feitos com o mapeamento para o "Vocabulário de Classes SP" e os Treinamento 2 e 4 foram feitos com o mapeamento para o "Vocabulário de Classes SP e Ciclos C347CA56". Esses treinamentos foram executados até 80 épocas utilizando o método "hold-out" para validação cruzada, onde as entradas com $k = 1$ foram utilizadas para teste e validação. Os Treinamentos 3 e 4 replicam os Treinamentos 1 e 2, com um tamanho total de *batch* igual a 128. A distribuição das classes dos Treinamentos 1 e 2 estão apresentadas nas Figuras 3.12 e 3.13, respectivamente. Na Figura 3.14 são apresentadas suas curvas de aprendizado e a Tabela 3.16 mostra os resultados de acurácia do teste desses dois treinamentos. A matriz de confusão do teste do Treinamento 1 é apresentada na Tabela 3.17 e a do teste do Treinamento 2 é apresentada na Tabela 3.18.

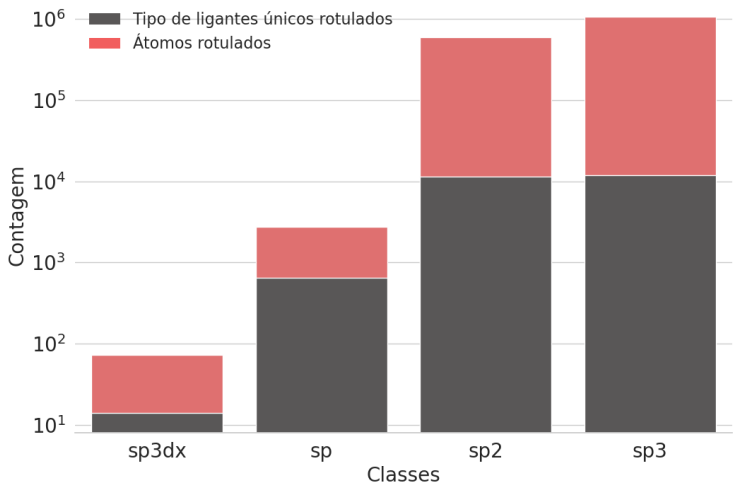

Figura 3.12: Distribuição da ocorrência das classes do "Vocabulário de Classes SP" por átomo dos ligantes presentes no conjunto de dados "Lig-qRankDB-SP-1.5-2.2", onde $d_{max} = 14830.8$.

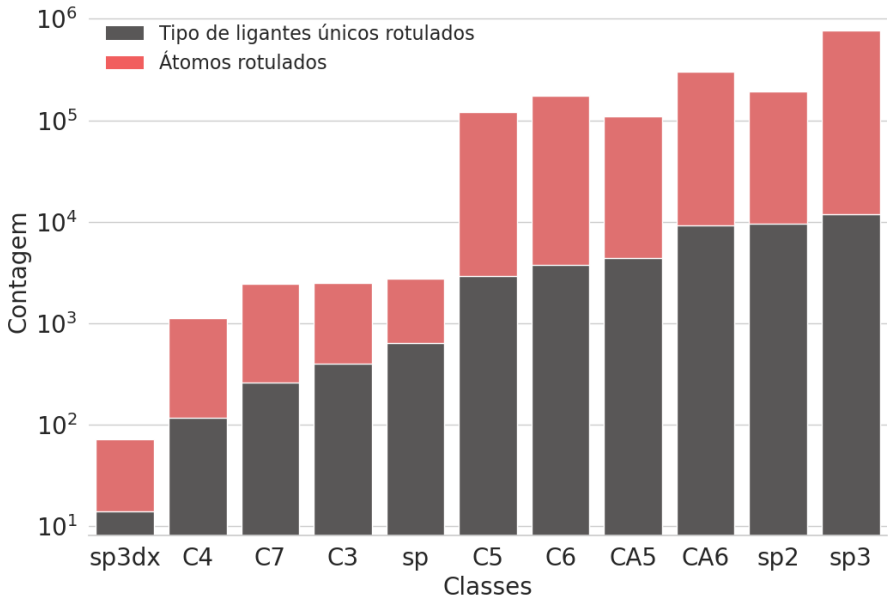

Figura 3.13: Distribuição da ocorrência das classes do "Vocabulário de Classes SP e Ciclos C347CA56" por átomo dos ligantes presentes no conjunto de dados "Lig-qRankDB-1.5-2.2", onde $d_{max} = 10654$.

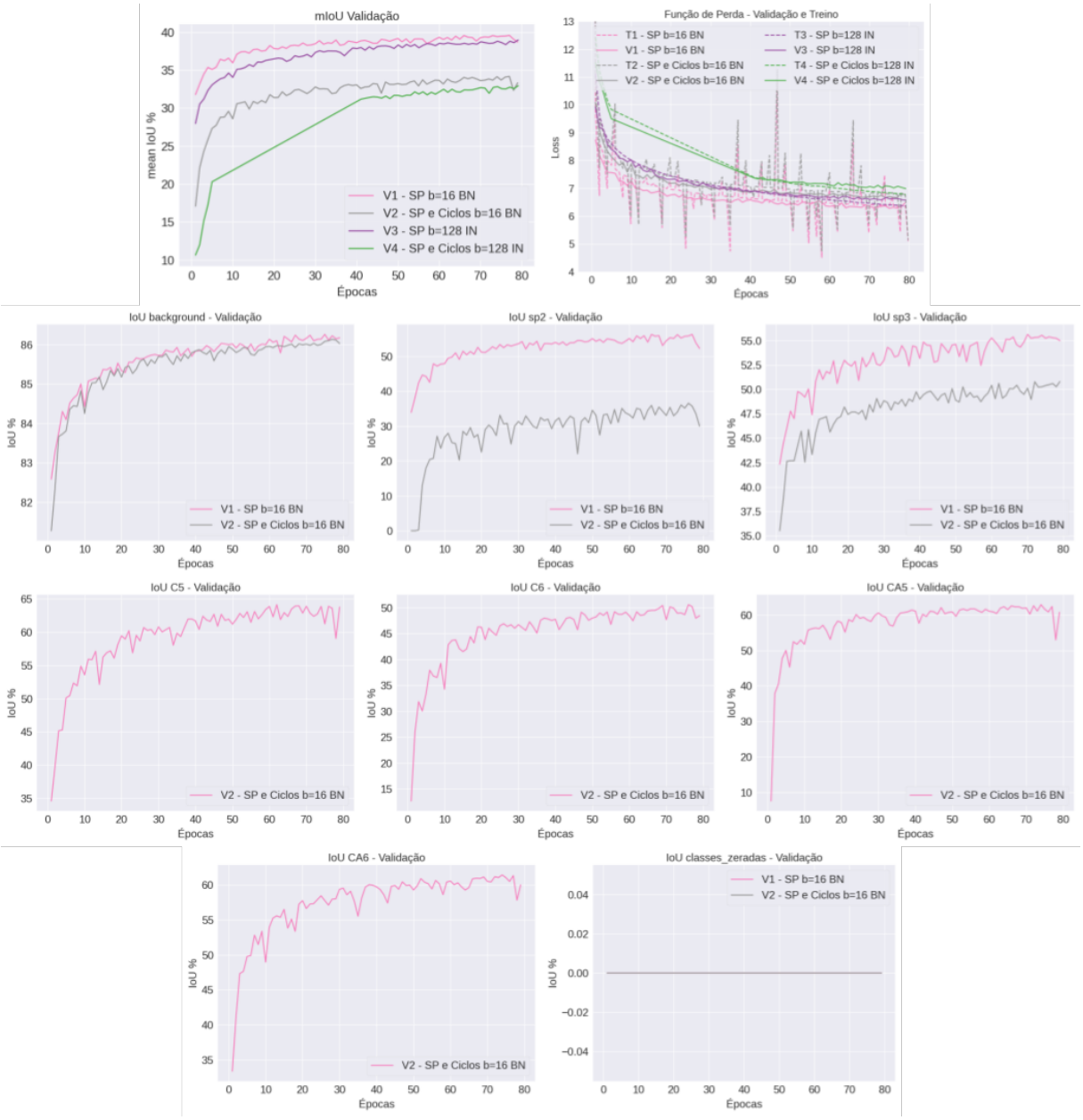

Figura 3.14: Curvas de aprendizado da validação dos dois treinamentos com o conjunto de dados "Lig-qRankDB-1.5-2.2" e "Vocabulário de Classes SP" e "Vocabulário de Classes SP e Ciclos C347CA56". Outros dois treinamentos com tamanho total de *batch* igual a 128 também foram incluídos nas curvas da acurácia média e da função de perda. São apresentadas as curvas da acurácia IoU da validação por classe, curvas de mIoU e da função de perda da validação e do treino dos treinamentos com cada um dos dois vocabulários SP. As classes que ficaram com a acurácia IoU zerada estão sendo mostradas com a mesma imagem para facilitar a visualização e foram chamadas de "classes_zeradas". As classes "classes_zeradas" são as seguintes: sp, sp3dx, C3, C4, C7.

Tabela 3.16: Resultados de acurácia e da função de perda para os dois treinamentos com o conjunto de dados "Lig-qRankDB-1.5-2.2" e "Vocabulário de Classes SP" e "Vocabulário de Classes SP e Ciclos C347CA56".

	Treino		**Validação**		**Teste**	
Nome	**Loss**	**mIoU**	**Loss**	**mIoU**	**Loss**	**mIoU**
Treinamento 1 SP	6.35	39.08	6.32	39.33	6.37	39.33
Treinamento 2 SP e Ciclos	6.77	33.36	6.67	34.08	6.77	33.78

Tabela 3.17: Matriz de confusão do teste do Treinamento 1 com o "Vocabulário de Classes SP"

	Background	**sp**	**sp2**	**sp3**	**sp3dx**
Background	**86.18**	0.00	1.16	3.39	0.00
sp	49.20	**0.00**	7.46	43.34	0.00
sp2	17.55	0.00	**55.22**	13.11	0.00
sp3	22.55	0.00	3.48	**55.25**	0.00
sp3dx	5.43	0.00	0.00	94.57	**0.00**

Tabela 3.18: Matriz de confusão do teste do Treinamento 2 com o "Vocabulário de Classes SP e Ciclos C347CA56"

	Background	**sp**	**sp2**	**sp3**	**sp3dx**	**C3**	**C4**	**C5**	**C6**	**C7**	**CA5**	**CA6**
Background	**86.13**	0.00	0.35	2.61	0.00	0.00	0.00	0.28	0.45	0.00	0.23	0.61
sp	46.50	**0.00**	8.54	41.07	0.00	0.00	0.00	1.96	0.56	0.00	0.62	0.74
sp2	22.18	0.00	**36.36**	19.74	0.00	0.00	0.00	0.27	0.27	0.00	0.26	0.67
sp3	24.05	0.00	3.06	**50.58**	0.00	0.00	0.00	0.31	0.64	0.00	0.13	0.55
sp3dx	4.35	0.00	0.00	95.65	**0.00**	0.00	0.00	0.00	0.00	0.00	0.00	0.00
C3	46.29	0.00	4.16	47.43	0.00	**0.00**	0.00	0.00	0.84	0.00	0.19	1.09
C4	38.37	0.00	3.35	24.27	0.00	0.00	**0.00**	28.02	0.10	0.00	0.43	5.46
C5	14.02	0.00	0.31	3.24	0.00	0.00	0.00	**61.28**	1.59	0.00	4.21	1.30
C6	20.18	0.00	0.41	4.84	0.00	0.00	0.00	0.71	**48.86**	0.00	0.54	6.00
C7	31.98	0.00	3.77	14.55	0.00	0.00	0.00	0.00	41.55	**0.00**	1.24	6.91
CA5	12.47	0.00	0.32	2.72	0.00	0.00	0.00	0.55	0.62	0.00	**61.23**	2.98
CA6	15.67	0.00	0.47	2.92	0.00	0.00	0.00	0.40	2.47	0.00	1.76	**60.89**

Os treinamentos baseados nas classes SP mostrados na Figura 3.14 apresentaram uma trajetória muito mais comportada em comparação com os treinamentos iniciais (Figura 3.3). Com exceção das classes minoritárias 'sp', 'sp3dx', 'C3', 'C4' e 'C7', todas as demais conseguiram atingir um IoU acima de 30%. A classe 'sp2', apesar de ser majoritária apresentou uma baixa acurácia IoU no teste do Treinamento 2 igual a 36.36%, comportando-se como uma classe difícil de convergir. No Treinamento 1 a classe 'sp2' teve um comportamento bem parecido com a classe 'sp3' e atingiram acurácias mIoU no teste maiores do que 55% (Tabelas 3.17 e 3.18. Nos dois treinamentos, a maioria das classes minoritárias foram confundidas com a classe 'sp3' majoritária, sendo a maior confusão entre a classe 'sp3dx' e a 'sp3'. A única exceção foi a classe 'C7' que foi mais confundida com a classe 'C6', como nos treinamentos feitos com o "Vocabulário de Átomos Genéricos e Ciclos C347CA56" apresentados anteriormente (Tabela 3.9). Essa diferença pode ser explicada pelo fato de que os ciclos de tamanho 7 tem um tamanho e forma bem parecidos com

os ciclos de tamanho 6 e isso facilita a confusão entre as duas classes. Todas as classes apresentam uma certa taxa de confusão com o ruído de fundo, sendo que para as classes minoritárias de ciclos essa confusão foi ainda maior. A confusão com o ruído de fundo além de erros na predição também inclui erros devido a diferenças entre a conformação predita e a esperada, e possivelmente predições de esferas atômicas menores do que os raios utilizados na rotulação. O fato de existir pouca confusão entre o ruído de fundo e as demais classes corrobora com a suposição levantada, a grande maioria do que é esperado de ruído de fundo está sendo predito como ruído. E soma-se a essa predição as outras classes que estão sendo confundidas com o ruído de fundo.

Para as classes minoritárias 'sp', 'C3', 'C4' e 'C7' com mais do que 1000 átomos rotulados, o que corresponde a duas ordens de grandeza menos que as classes majoritárias, seria interessante avaliar o uso de pesos na função de perda para lidar com esse desbalanço. Mas a baixa ocorrência da classe 'sp3dx' com menos do que 100 átomos rotulados é um indicativo de inviabilidade de aprendizagem para essa classe.

A acurácia média da validação em termos de mIoU e da função de perda dos Treinamentos 3 e 4 adicionais da Figura 3.16 mostram que o tamanho total de *batch* igual a 16 manteve o melhor resultado nessa modelagem com os vocabulários SP.

O Treinamento 2 demonstra que a junção das classes SP com a simplificação de ciclos de tamanhos de 3 a 7 acaba por prejudicar a acurácia das classes SP, com maior impacto no aprendizado da classe 'sp2'. As classes de ciclos aromáticos cobrem átomos com hibridização 'sp2' e isso pode ter prejudicado a convergência dessa classe.

Os modelos obtidos nesses treinamentos com os vocabulários SP não apresentaram boa acurácia para as classes SP. Apenas no Treinamento 1 a acurácia IoU da classe 'sp3' e 'sp2' passam o valor de 50%, mas as demais classes SP mantêm suas acurácias zeradas. Isso reflete na acurácia média mIoU dos modelos que ficou abaixo de 40%, um resultado que não é muito interessante e demonstra que essa modelagem deveria ser refatorada.

Em seguida, os resultados dos treinamentos baseados no vocabulário SP com simplificação serão apresentados juntos. O "Treinamento 1 AB" foi feito com o mapeamento para o "Vocabulário da Região do Ligante" e o "Treinamento 2 ABC" foi feito com o mapeamento para o "Vocabulário de Átomos e Ciclos Genéricos". Esses treinamentos foram executados até 80 épocas utilizando o método "hold-out" para validação cruzada, onde as entradas com $k = 1$ foram utilizadas para teste e validação. A classe de átomos genéricos do Treinamento 1 foi utilizada para rotular 2565834 átomos. A distribuição das classes do Treinamento 2 no conjunto de dados "Lig-qRankDB-SP-1.5-2.2" está apresentada na Figura 3.15. Na Figura 3.16 são apresentadas suas curvas de aprendizado da validação e a Tabela 3.19 mostra os resultados do treino, validação e teste desses dois treinamentos. Nas curvas de acurácia média mIoU e da função de perda, também são mostrados outros dois treinamentos equivalentes com tamanho total de *batch* igual a 128. A matriz de confusão do teste do Treinamento 1 AB é apresentada na Tabela 3.20 e do teste do Treinamento 2 ABC é apresentada na Tabela 3.21.

Figura 3.15: Distribuição da ocorrência das classes do "Vocabulário de Átomos e Ciclos Genéricos" por átomo dos ligantes presentes no conjunto de dados "Lig-qRankDB-1.5-2.2", onde $d_{max} = 2.1$.

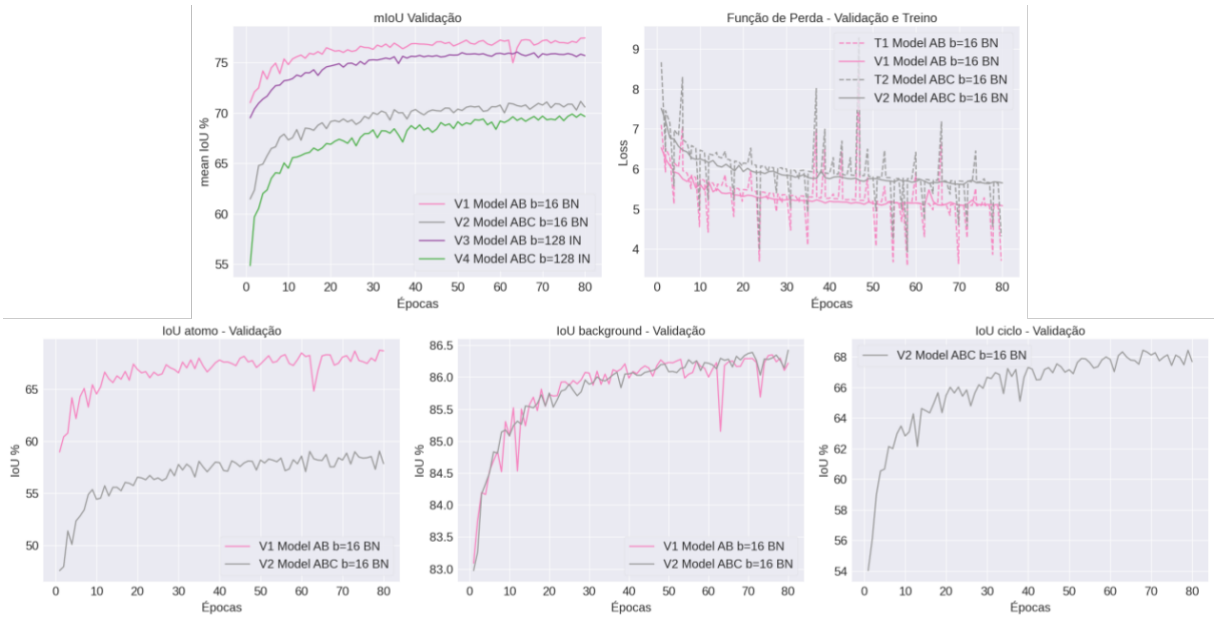

Figura 3.16: Curvas de aprendizado da validação dos dois treinamentos com o conjunto de dados "Lig-qRankDB-1.5-2.2" e "Vocabulário da Região do Ligante" e "Vocabulário de Átomos e Ciclos Genéricos". Outros dois treinamentos com tamanho total de *batch* igual a 128 também foram incluídos nas curvas da acurácia média mIoU. São apresentadas as curvas da acurácia IoU da validação por classe, curvas de mIoU e da função de perda da validação e do treino dos treinamentos com cada um dos dois vocabulários simplificados.

Tabela 3.19: Resultados de acurácia e da função de perda para os dois treinamentos com o conjunto de dados "Lig-qRankDB-1.5-2.2" e "Vocabulário da Região do Ligante" e "Vocabulário de Átomos e Ciclos Genéricos".

	Treino		**Validação**		**Teste**	
Nome	**Loss**	**mIoU**	**Loss**	**mIoU**	**Loss**	**mIoU**
Treinamento 1 AB	5.06	77.00	5.08	77.45	5.12	77.42
Treinamento 2 ABC	5.64	70.52	5.65	70.59	5.72	70.43

Tabela 3.20: Matriz de confusão do Teste do Treinamento 3 com o Conjunto de dados "Lig-qRankDB-SP-1.5-2.2" e "Vocabulário da Região do Ligante".

	Background	**Átomo**
Background	**86.20**	6.38
Átomo	16.86	**68.63**

Tabela 3.21: Matriz de confusão do Teste do Treinamento 4 com o Conjunto de dados "Lig-qRankDB-1.5-2.2" e "Vocabulário de Átomos e Ciclos Genéricos".

	Background	**Ciclo**	**Átomo**
Background	**86.33**	1.98	2.83
Ciclo	14.98	**66.88**	2.84
Átomo	24.66	3.58	**58.06**

Os resultados com os vocabulários simplificados na melhor configuração apresentaram uma melhora em relação aos primeiros resultados positivos com essas simplificações (Figuras 3.4 e 3.5). O Treinamento 1 AB apresentou uma melhora de mais do que 5% na classe de átomo e o Treinamento 2 ABC apresentou quase 2x de melhora nas classes de átomo e ciclo genéricos. Quando a classe de ciclos genéricos passa a ser considerada, é observada uma queda no aprendizado da classe de átomos genéricos com uma diferença de 10% de acurácia IoU entre os Treinamentos 1 e 2 nas Tabelas 3.20 e 3.21. Essa diferença indica que o padrão de subestruturas cíclicas nas imagens dos ligantes é melhor entendido pelo modelo do que o padrão deixado por átomos fora de ciclos, e logo, a acurácia dos átomos em ciclos aumenta a acurácia de átomos genéricos quando esses são considerados juntos no Treinamento 1. A maior confusão dos modelos simplificados continua sendo com o ruído de fundo. É interessante perceber que a modelagem mais complexa do melhor treinamento apresentado na Tabela 3.9, que diferencia os ciclos pelo tamanho e aromaticidade, melhora a acurácia da classe de Átomos fora de ciclos do Treinamento 2 para 59.5% e ainda mantém 3 classes de ciclos com acurácia acima de 60% (classes C5, CA5 e CA6). A comparação desses resultados demonstra o impacto da modelagem na acurácia do modelo e indica que a separação dos ciclos por tamanho ajuda o modelo a aprender melhor os padrões deixados por essas classes nas imagens dos ligantes.

Os resultados dos treinamentos baseados no vocabulário de tipo de átomo serão apresentados juntos a seguir. O "Treinamento 1 TiposÁtomos" foi feito com o mapeamento para o "Vocabulário de Tipos de Átomos" e o "Treinamento 2 GruposÁtomos" foi feito com

o mapeamento para o "Vocabulário de Tipos de Átomos Agrupados". Esses treinamentos foram executados até 160 épocas utilizando o método "hold-out" para validação cruzada, onde as entradas com $k = 1$ foram utilizadas para teste e validação. A distribuição das classes do Treinamento 1 e 2 estão apresentadas nas Figuras 3.17 e 3.18, respectivamente. Na Figura 3.19, são apresentadas suas curvas de aprendizado e a Tabela 3.22 mostra os resultados de acurácia média e da função de perda do treino, validação e teste desses dois treinamentos. A matriz de confusão do teste do Treinamento 1 é apresentada na Tabela 3.23 e do teste do Treinamento 2 é apresentada na Tabela 3.24. Outros dois treinamentos que replicam os Treinamentos 1 e 2 com tamanho total de *batch* igual a 128 também são mostrados nas curvas da acurácia média mIoU e da função de perda da validação na Figura 3.19.

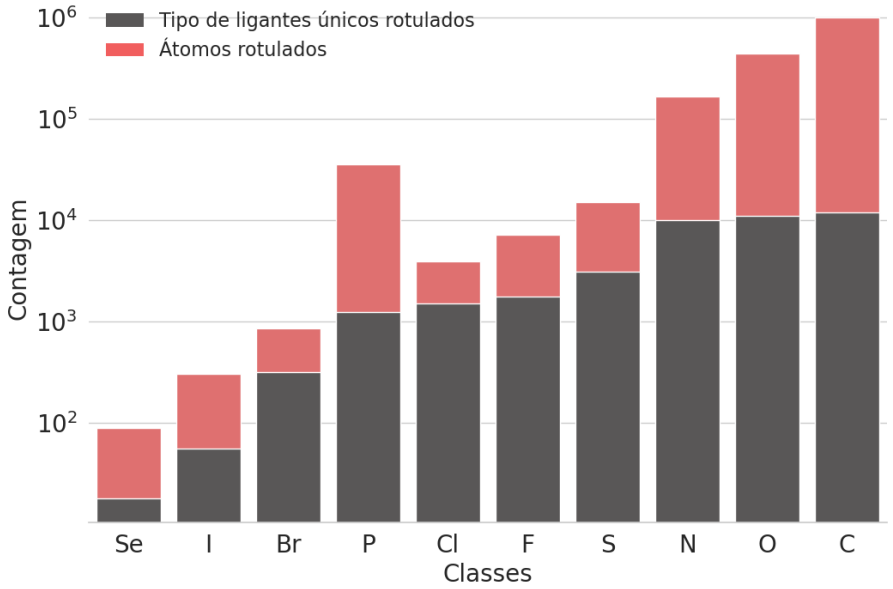

Figura 3.17: Distribuição da ocorrência das classes do "Vocabulário de Tipos de Átomos" por átomo dos ligantes presentes no conjunto de dados "Lig-qRankDB-SP-1.5-2.2", onde $d_{max} = 11256.7$.

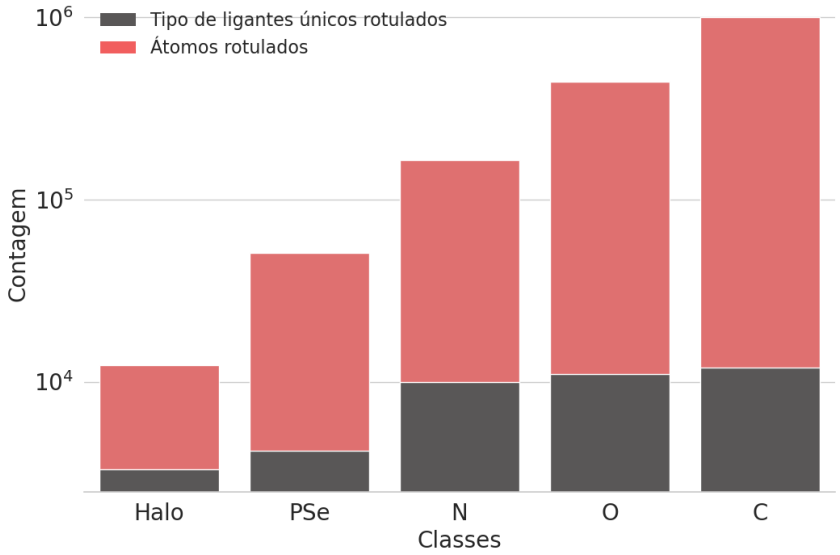

Figura 3.18: Distribuição da ocorrência das classes do "Vocabulário de Tipos de Átomos Agrupados" por átomo dos ligantes presentes no conjunto de dados "Lig-qRankDB-1.5-2.2", onde $d_{max} = 81.5$.

Tabela 3.22: Resultados de acurácia mIoU e da função de perda para os dois treinamentos com o conjunto de dados "Lig-qRankDB-1.5-2.2" e 'Vocabulário de Tipos de Átomos" e "Vocabulário de Tipos de Átomos Agrupados".

	Treino		**Validação**		**Teste**	
Nome	**Loss**	**mIoU**	**Loss**	**mIoU**	**Loss**	**mIoU**
Treinamento 1 Átomos	6.01	38.41	5.96	40.05	6.10	39.60
Treinamento 2 Átomos Agrupados	5.98	57.48	5.96	59.10	6.08	57.55

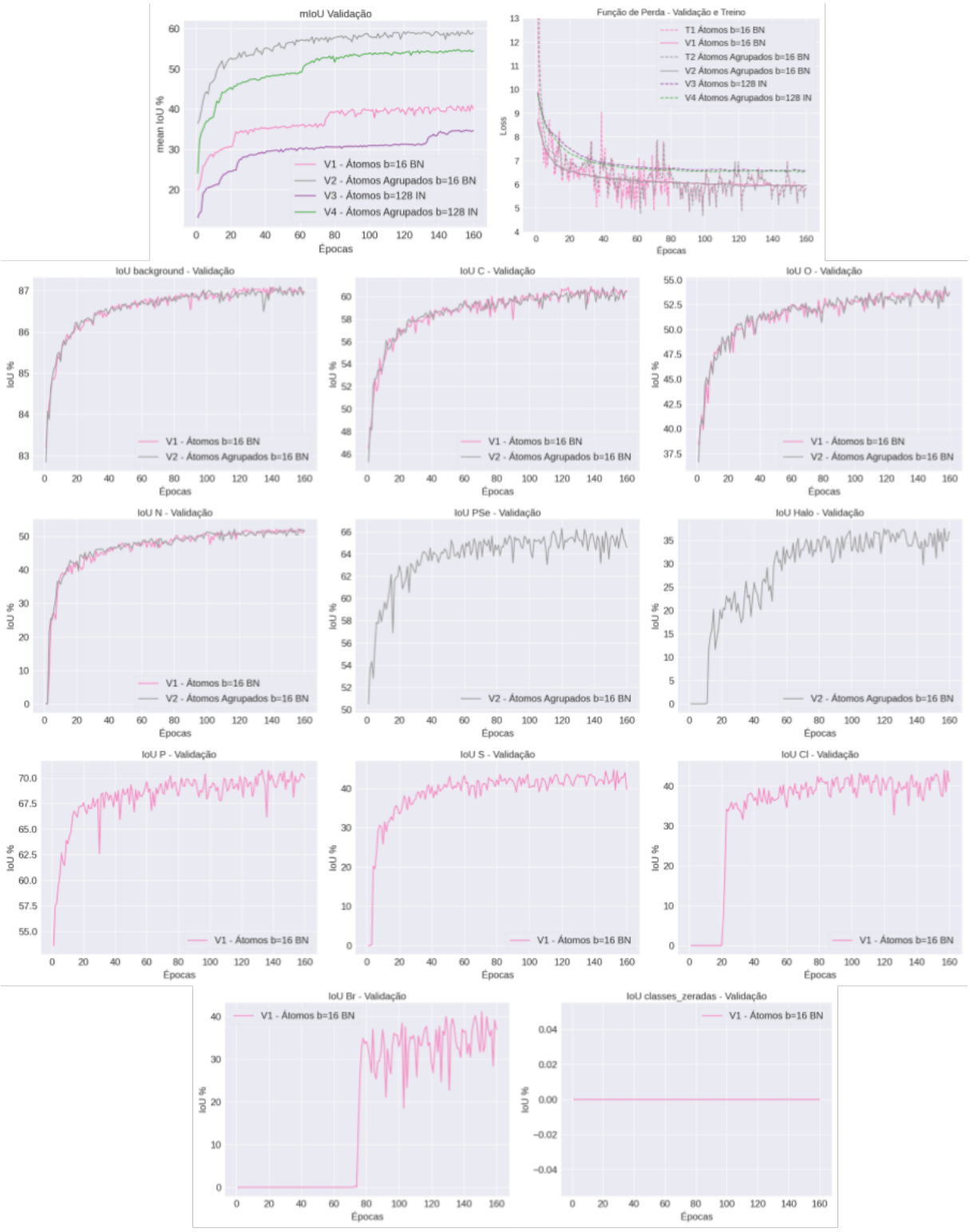

Figura 3.19: Curvas de aprendizado da validação dos dois treinamentos com o conjunto de dados "Lig-qRankDB-1.5-2.2" e "Vocabulário de Tipos de Átomos" e "Vocabulário de Tipos de Átomos Agrupados". Outros dois treinamentos com tamanho total de *batch* igual a 128 também foram incluídos nas curvas da acurácia média mIoU e da função de perda. São apresentadas as curvas da acurácia IoU da validação por classe, curvas de mIoU e da função de perda da validação e do treino dos treinamentos com cada um dos dois vocabulários. As classes que ficaram com a acurácia IoU zerada estão sendo mostradas com a mesma imagem para facilitar a visualização e foram chamadas de "classes_zeradas". As classes "classes_zeradas" são as seguintes: F, I e Se.

Tabela 3.23: Matriz de confusão do Teste do Treinamento 1 com o Conjunto de dados "Lig-qRankDB-1.5-2.2" e 'Vocabulário de Tipos de Átomos"

	Background	C	O	N	P	S	F	I	Se	Cl	Br
Background	**86.6**	3.2	1.2	0.2	0.0	0.0	0.0	0.0	0.0	0.0	0.0
C	19.4	**59.8**	1.9	0.8	0.0	0.1	0.0	0.0	0.0	0.0	0.0
O	22.5	7.2	**52.1**	0.6	0.4	0.1	0.0	0.0	0.0	0.0	0.0
N	15.8	16.1	5.3	**50.7**	0.1	0.0	0.0	0.0	0.0	0.0	0.0
P	11.2	3.4	5.4	0.0	**65.9**	0.7	0.0	0.0	0.0	0.0	0.0
S	15.2	12.7	8.5	0.2	6.2	**44.4**	0.0	0.0	0.0	0.5	0.0
F	37.3	12.9	48.8	0.1	0.1	0.2	**0.0**	0.0	0.0	0.6	0.0
I	50.9	0.0	0.0	0.0	0.0	0.0	0.0	**0.0**	0.0	0.0	49.1
Se	12.5	0.0	87.5	0.0	0.0	0.0	0.0	0.0	**0.0**	0.0	0.0
Cl	23.5	8.8	9.2	0.0	0.0	0.0	0.0	0.0	0.0	**41.1**	1.5
Br	15.3	0.0	0.3	0.0	0.0	7.2	0.0	0.0	0.0	8.9	**36.5**

Tabela 3.24: Matriz de confusão do Teste do Treinamento 2 com o Conjunto de dados "Lig-qRankDB-1.5-2.2" e "Vocabulário de Tipos de Átomos Agrupados"

	Background	C	O	N	PSe	Halo
Background	**86.64**	3.21	1.32	0.28	0.11	0.01
C	19.06	**60.13**	2.06	1.05	0.17	0.04
O	20.96	6.39	**53.02**	0.84	0.60	0.12
N	14.59	14.00	4.13	**51.89**	0.18	0.04
PSe	9.51	4.23	5.38	0.16	**63.56**	0.05
Halo	28.03	9.53	17.18	0.22	0.81	**30.01**

Os resultados dos Treinamentos 1 e 2 mostram que a informação presente na densidade eletrônica é suficiente para identificar um átomo de carbono (classe C) com aproximadamente 60% de acurácia IoU, um átomo de oxigênio (classe O) ou de nitrogênio (classe N) com mais de 50% de acurácia IoU, um átomo de fosforo (classe P) com 65% de acurácia IoU e os demais com menos do que 45% de acurácia IoU (Tabelas 3.23 e 3.24). Os átomos minoritários, selênio (classe Se), iodo (classe I) e flúor (classe F), em ordem crescente de ocorrência, tiveram suas acurácias zeradas no Treinamento 1. O átomo de flúor possui quase 10000 átomos rotulados nesse conjunto, configurando-se como uma classe difícil de convergir e com uma confusão de 48.8% com o átomo de oxigênio no Treinamento 1. O átomo de selênio foi confundido com o átomo de oxigênio em 87% dos seus pontos e o átomo de iodo foi confundido com o átomo de bromo em 49.1% dos seus pontos. O átomo de iodo também foi a classe que apresentou maior confusão com o ruído de fundo, igual a 50.9% dos seus pontos. A baixa ocorrência das demais classes minoritárias pode explicar a degradação do seu aprendizado. Os átomos de cloro (classe Cl) e de bromo (classe Br), apesar de minoritários, apresentaram acurácias IoU de 41.1% e 36.5%, respectivamente, no Treinamento 1 o que indica que o padrão deixado por esses átomos na densidade é mais facilmente aprendido pelo modelo. O átomo de enxofre (S), por outro lado, apesar de possuir mais do que 10000 átomos rotulados nesse conjunto teve uma acurácia IoU de 44.4%, com pouca confusão com os átomos de carbono e oxigênio, comportando-se como uma classe difícil de convergir.

O "Vocabulário de Tipos de Átomos Agrupados" utilizado no Treinamento 2 foi uma tentativa de melhorar a acurácia das classes S, F, I, Se, Cl e Br que não passaram de 50% de acurácia no Treinamento 1. Os átomos da família dos halogênios (I, Cl e Br) foram agrupados na classe 'Halo' e os demais (S, F e Se) foram agrupados junto com o átomo de fósforo na classe 'PSe'. Esses agrupamentos ajudaram a melhorar a acurácia média do Treinamento 2 pois removeram as classes zeradas, mas as classes agrupadas não apresentaram muita melhora em relação a melhor acurácia das classes separadamente. A classe PSe teve uma acurácia menor do que a classe P, um indicativo de que a acurácia dessa ultima classe acaba por mascarar a baixa convergência das demais classes. O mesmo comportamento é observado na classe Halo, e a junção do átomo de iodo nesse grupo fez com que sua confusão com o background ficasse em 28% de IoU. Além disso, uma possível causa para a grande confusão entre a classe dos halogênios e o background é o fato do raio atômico experimental desses átomos ser maior do que o raio dos demais átomos, que são mais abundantes. Isso pode fazer com que o modelo infira um raio atômico menor para os halogênios e induza um maior erro com o background na borda das esferas atômicas dessa classe. Para os átomos de carbono, oxigênio e nitrogênio, esse agrupamento promoveu uma melhora de aproximadamente 1% nas suas acurácias IoU e diminuiu aproximadamente na mesma taxa a confusão entre essas classes.

As duas modelagens baseadas no tipo do átomo tiveram resultados muito melhores do que o esperado por especialistas da área. Esse é um resultado muito interessante para mostrar a capacidade da técnica de aprendizado profundo em extrair informações de dados complexos, as quais muitas vezes não são percebidas a olho nu. Não foi avaliado o uso de pesos nas classes desses dois treinamentos para lidar com o desbalanço, isso poderia contribuir para a convergência das classes S, Cl e Br atingir mais do que 50% de acurácia IoU. Dada a complexidade dessa modelagem, os resultados obtidos nos Treinamentos 1 e 2 são muito promissores. Uma predição correta com esse modelo pode desvendar a estrutura completa do ligante com detalhes dos tipos de átomos em cada posição da região do ligante. Isso é um resultado muito promissor na direção de criar sugestões de esqueletos completos para os ligantes a partir das predições dos modelos.

3.2 Teste da Aplicação

A acurácia apresentada para os modelos de aprendizado profundo correspondem à taxa de acerto do modelo em imagens do conjunto de teste. Quando esse modelo é utilizado na aplicação para uma nova entrada, as imagens criadas a partir dessa nova entrada seguem uma metodologia um pouco diferente da utilizada na criação das imagens do conjunto de teste. No caso de uma entrada nova desconhecida, não se sabe as posições atômicas do ligante desconhecido para extrair uma caixa delimitadora na fronteira das suas posições atômicas. O que se faz é utilizar o volume do *blob* encontrado para estimar o tamanho da sua caixa delimitadora, como foi descrito na Seção 2.5.3. Essa metodologia cria caixas delimitadoras cúbicas para todos os *blobs*, as quais acabam sendo muito maiores do que o ideal após serem expandidas para garantir que o ligante será encontrado no seu interior. Porém, isso pode incluir imagens de outras moléculas na caixa delimitadora do *blob*,

como outros ligantes adjacentes, reagentes, águas e mais ruído. O tamanho da caixa delimitadora do *blob* vai afetar a transformação dos seus valores na escala quantile rank e pode criar imagens com a distribuição dos valores da intensidade um pouco diferentes daqueles do conjunto de teste. Essas diferenças podem afetar o desempenho final do modelo de aprendizado profundo.

Por esse motivo, tornou-se necessário avaliar o desempenho da aplicação para certificar que a acurácia apresentada pelo modelo é mantida. A avaliação da aplicação desenvolvida neste trabalho foi feita apenas para o modelo final com o conjunto de dados "Lig-qRankDB-1.5-2.2" e o "Vocabulário de Átomos Genéricos e Ciclos C347CA56", e foi comparada com os resultados apresentados nas Tabelas 3.9 e 3.8.

A metodologia seguida consistiu na execução da aplicação para todas as entradas do teste do conjunto de dados "Lig-qRankDB-1.5-2.2", para as quais $k = 1$. Uma tabela com as posições centrais (x, y, z) dos ligantes desse conjunto de teste foi utilizada para executar a aplicação e fazer a busca por *blobs* apenas nessas regiões. Dessa forma, a aplicação buscou por *blobs* nas posições dos ligantes do conjunto de teste e, para os que foram encontrados, criou suas imagens em nuvem de pontos 3D seguindo a metodologia implementada na aplicação. Com esse resultado, foi possível obter as entradas necessárias para testar a acurácia do modelo na aplicação. Em seguida, um *script* de rotulação adaptado foi utilizado para rotular as imagens dos *blobs* dos ligantes criados. Esse *script* utiliza a rotulação da estrutura do ligante esperada e segue a mesma metodologia utilizada na rotulação das imagens do conjunto de teste "Lig-qRankDB-SP-1.5-2.2". Assim foi obtido a saída esperada para as entradas criadas com a aplicação. Com esses dois resultados foi possível executar, na pipeline de treinamento, um teste do modelo final utilizando as imagens criadas pela aplicação e devidamente rotuladas. Esse teste forneceu a acurácia da Aplicação desenvolvida neste trabalho.

A Aplicação foi utilizada para criar as imagens dos ligantes com três configurações diferentes em relação ao parâmetro do nível de contorno σ (utilizado na busca pelos *blobs*). Assim, também foi possível avaliar o impacto deste parâmetro na acurácia da aplicação. Nos demais parâmetros, foram mantidos seus valores padrão. O primeiro teste foi feito com um contorno igual a 2 σ, o segundo teste com um contorno igual a 2.5 σ e o terceiro com um contorno igual a 3 σ. As configurações desses três testes estão apresentadas na Tabela 3.25 junto com suas acurácias em termo de mIoU e a porcentagem de entradas que tiveram suas imagens criadas corretamente. O uso do contorno igual a 2 σ permitiu criar 2714 imagens das 3036 presentes no conjunto de teste, o contorno igual a 2.5 σ criou 2570 imagens e o contorno igual a 3 σ criou 2381 imagens. As imagens que não foram criadas foi devido a erros na função de busca pelos *blobs*, que para esses casos não conseguiu encontrar o *blob* do ligante correspondente. As matrizes de confusão apresentadas nas Tabelas 3.26, 3.27 e 3.28 mostram a acurácia IoU por classe da aplicação com os diferentes valores no parâmetro do nível de contorno σ.

Tabela 3.25: Configurações e Resultados de acurácia mIoU dos testes da aplicação com o conjunto de teste "Lig-qRankDB-SP-1.5-2.2" e $k = 1$

	Parâmetros da Aplicação				Resultados	Teste	
Nome	**Nível de Contorno σ**	**Mínimo Número de Átomo (Volume do *Blob*)**	**Nota Mínima do *Blob***	**Intensidade Mínima do Pico do *Blob***	**Número de Imagens dos *Blobs* Criadas**	**Loss**	**mIoU**
Teste 2 σ	2	5	0	0	2714	6.157	40.86
Teste 2.5 σ	2.5	5	0	0	2570	6.099	43.48
Teste 3 σ	3	5	0	0	2381	6.164	45.11

Tabela 3.26: Matriz de confusão do teste da aplicação com contorno 2 σ

	Background	**Átomo**	**C5**	**CA5**	**C6**	**CA6**	**C3**	**C4**	**C7**
Background	**87.47**	4.97	0.29	0.26	1.41	0.71	0.00	0.01	0.01
Átomo	18.82	**45.71**	0.29	0.22	0.62	0.70	0.00	0.00	0.02
C5	15.19	3.98	**52.79**	3.00	2.33	1.53	0.00	0.00	0.00
CA5	12.71	2.84	0.73	**54.49**	0.57	3.83	0.00	0.00	0.00
C6	15.76	4.82	0.67	0.32	**30.05**	5.85	0.01	0.00	0.04
CA6	14.05	3.26	0.39	1.55	1.50	**55.02**	0.00	0.00	0.00
C3	33.22	35.49	0.00	0.00	0.55	1.37	**13.57**	0.00	0.04
C4	23.87	6.11	17.70	0.37	4.09	3.60	0.00	**13.68**	0.00
C7	23.21	9.71	0.00	0.30	22.52	3.28	0.00	0.00	**14.97**

Tabela 3.27: Matriz de confusão do teste da aplicação com contorno 2.5 σ

	Background	**Átomo**	**C5**	**CA5**	**C6**	**CA6**	**C3**	**C4**	**C7**
Background	**87.68**	4.48	0.26	0.24	1.44	0.71	0.00	0.00	0.01
Átomo	19.35	**48.95**	0.29	0.24	0.63	0.72	0.00	0.00	0.02
C5	14.42	3.80	**56.06**	2.91	2.81	1.46	0.00	0.00	0.00
CA5	12.39	3.13	0.61	**56.91**	0.57	3.98	0.00	0.00	0.00
C6	15.68	4.78	0.52	0.37	**32.44**	5.84	0.00	0.00	0.05
CA6	14.08	3.02	0.35	1.34	1.32	**57.04**	0.00	0.00	0.00
C3	36.61	39.57	0.00	0.00	0.88	1.59	**15.97**	0.00	0.00
C4	28.85	12.06	23.47	0.56	4.45	5.57	0.00	**20.59**	0.00
C7	19.63	9.42	0.00	0.30	25.70	0.07	0.00	0.00	**15.72**

Tabela 3.28: Matriz de confusão do teste da aplicação com contorno 3 σ

	Background	**Átomo**	**C5**	**CA5**	**C6**	**CA6**	**C3**	**C4**	**C7**
Background	**87.76**	4.03	0.25	0.23	1.39	0.66	0.00	0.00	0.01
Átomo	20.25	**51.55**	0.31	0.21	0.66	0.67	0.00	0.00	0.02
C5	14.82	3.50	**57.94**	2.77	2.56	1.40	0.00	0.00	0.00
CA5	13.65	2.92	0.86	**58.48**	0.50	3.72	0.00	0.00	0.00
C6	16.26	4.20	0.48	0.27	**36.02**	5.58	0.00	0.00	0.02
CA6	15.06	2.92	0.31	1.23	1.24	**58.75**	0.00	0.00	0.00
C3	37.90	38.54	0.00	0.00	0.69	1.57	**15.81**	0.00	0.00
C4	37.05	5.63	25.41	0.29	4.66	0.19	0.00	**21.73**	0.00
C7	20.50	8.86	0.00	0.24	26.15	0.00	0.03	0.00	**17.92**

A acurácia média mIoU do teste da Aplicação utilizando os contornos 2σ, 2.5σ e 3σ foi de 40.9%, 43.5% e 45.1% de acerto, respectivamente. Em comparação com o melhor

resultado dessa modelagem com o "Vocabulário de Átomos Genéricos e Ciclos C347CA56" apresentado na Tabela 3.9, a Aplicação apresentou uma piora na acurácia média com diminuição de 8.9%, 6.3% e 4.7% na mIoU, respectivamente. O melhor resultado da Aplicação foi obtido com o contorno de 3σ, em contrapartida esse contorno foi o que menos encontrou os *blobs* dos ligantes, com apenas 78.4% das imagens do conjunto de teste criadas. Em comparação com o melhor resultado dessa modelagem, todas as classes preditas com a Aplicação tiveram quedas de acurácia próximas a diferença média, com exceção da classe C6. A classe C6 apresentou um queda na sua acurácia igual a 15% na Aplicação à 3σ, e um aumento na confusão da classe C7 com a C6 que passou de 18% para 26% aproximadamente. O mesmo comportamento aconteceu para a acurácia da Aplicação com os demais contornos σ.

Seria necessário uma investigação desses resultados para entender quais ligantes estão sendo perdidos quando um contorno mais fino é aplicado e o que exatamente está causando a queda na acurácia do modelo. A presença de moléculas adjacentes ao ligante na sua imagem final se configura como falsos negativos e adiciona mais ruído na rotulação. Esse ruído pode estar impactando todas as classes do modelo. A partir desse resultado o parâmetro do contorno σ deve ser definido de forma a equilibrar para cada objetivo de pesquisa a porcentagem de detecção de *blobs* com a acurácia final das predições.

O ligante 6MY da entrada 5JTT do PDB (apresentado na Figura 3.7) foi escolhido para fazer uma análise do impacto de diferentes resoluções dessa entrada no comportamento do modelo do Treinamento 1 com o conjunto Lig-qRankDB-SP-1.5-2.2 C347CA56 na aplicação *NP³ Blob Label*. É possível fazer o refinamento de um dado de cristalografia e limitar sua resolução em um dado corte maior do que a sua menor resolução obtida até o momento. A menor resolução da entrada 5JTT no PDB é 1.85 Å, nesse teste foi escolhido limitar sua resolução nos seguintes valores: 1.9 Å, 2 Å, 2.1 Å, 2.2 Å, 2.5 Å, 2.8 Å, 3 Å, 3.3 Å e 3.5 Å. Foram realizados outros 9 refinamentos com o Dimple a partir do resultado do refinamento do conjunto de treinamento dessa entrada utilizando cada um dos limites de resolução listados e apenas um ciclo de refinamento foi definido nos parâmetros para não desviar muito o dado experimental do modelo atômico do ligante depositado. Em seguida a aplicação *NP³ Blob Label* foi utilizada para criar a imagem do ligante 6MY na entrada do conjunto de teste (com a melhor resolução) e nas 9 entradas criadas simulando resoluções piores, e para obter as predições do modelo do Treinamento 1. O parâmetro de nível de contorno para buscar por *blobs* foi definido igual a 3σ e os demais parâmetros mantiveram seus valores padrão. Essas 10 imagens foram rotuladas separadamente seguindo a metodologia desse trabalho para obter as classes esperadas para esse ligante e avaliar as predições do modelo. Os resultados dessa análise são apresentados na Figura 3.20, e o impacto do limite de resolução é ilustrado nas imagens correspondentes do ligante nas entradas criadas e nas suas respectivas predições.

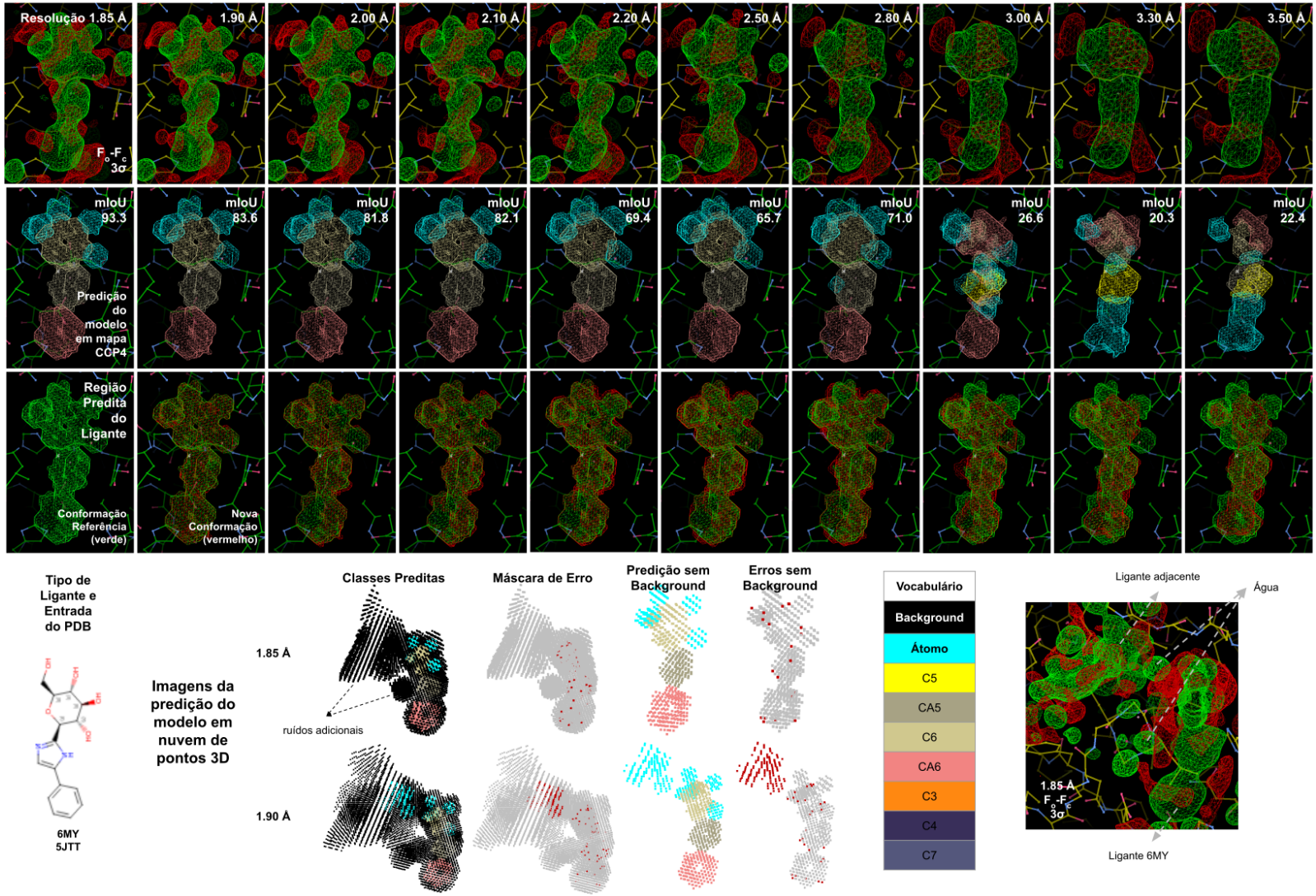

Figura 3.20: Exemplo do impacto de diferentes resoluções da mesma entrada do PDB 5JTT para o ligante 6MY no resultado da aplicação *NP³ Blob Label*. Mais detalhes no texto.

A primeira linha de imagens da Figura 3.20 mostra o mapa da densidade residual desse ligante no contorno de 3σ para a entrada 5JTT em diferentes resoluções. As seguintes resoluções são mostradas: 1.85 Å, 1.9 Å, 2 Å, 2.1 Å, 2.2 Å, 2.5 Å, 2.8 Å, 3 Å, 3.3 Å e 3.5 Å. A segunda linha de imagens mostra o resultado da predição do modelo do Treinamento 1 - Lig-qRankDB-SP-1.5-2.2 C347CA56 para todas as resoluções em mapas CCP4, com a acurácia média do modelo em termos de mIoU no canto superior direito de cada imagem. A terceira linha de imagens mostra o resultado da predição do modelo para a região do ligante, que consiste na remoção de todos os pontos preditos como *background*. A região do ligante predita na resolução de 1.85 Å (mostrada em verde) foi utilizada como referência de conformação para comparar com as predições nas demais resoluções (mostradas em vermelho). As imagens mostradas nas três primeiras linhas foram feitas na ferramenta Coot com o resultado da aplicação e foram focalizadas na densidade residual do ligante 6MY. Na parte inferior dessa figura é apresentado a imagem do ligante em representação 2D e a sua imagem e predições em nuvem de pontos criadas pela aplicação *NP³ Blob Label* para as duas menores resoluções, com destaque para a presença de ruídos adicionais. As cores das classes do vocabulário mostrado no canto inferior da figura foram utilizadas para colorir o resultado da predição. No canto inferior direito é mostrado o mapa da densidade residual da entrada 5JTT na resolução 1.85 Å para o mesmo ligante em um ângulo que evidencia a presença de águas e de outro ligante adjacentes que adicionaram ruído nas imagens criadas.

A Figura 3.20 mostra como a resolução da entrada pode afetar a qualidade da imagem do ligante na densidade residual e impactar o resultado de predição da aplicação *NP³ Blob Label*. Resoluções fora do intervalo de resolução das entradas do conjunto de treinamento Lig-qRankDB-SP-1.5-2.2 C347CA56 também foram incluídas nessa análise para avaliar o limite de resolução em que a predição retorna resultados confiáveis, para este exemplo com o ligante 6MY da entrada 5JTT do PDB. A partir de 2.8 Å a nuvem eletrônica do ligante 6MY começa a ficar sem definição do contorno dos seus átomos e das suas subestruturas cíclicas, e a cada incremento de resolução a imagem fica mais borrada. A falta de definição na imagem do ligante nas resoluções maiores ou iguais a 3 Å resulta em uma queda na acurácia do modelo e a média mIoU fica abaixo de 30%. Desde o primeiro corte de resolução igual a 1.9 Å é observado uma queda de 10% na acurácia média da aplicação, resultante de falsos negativos na predição da classe de átomos devido a ruídos de águas e de um outro ligante adjacente adicionados nas imagens dos ligantes (mostrados na parte inferior da Figura 3.20). Outro recorte do mapa da densidade residual do ligante 6MY é mostrado no canto inferior direito dessa figura para evidenciar a origem do ruído adicionado nas imagens do ligante criadas com a aplicação. A caixa delimitadora desse ligante na aplicação acaba englobando a densidade residual das moléculas adjacentes, isso afeta a transformação dos valores na escala quantile rank e amplifica esses ruídos. Isso não acontece com a imagem do ligante no conjunto de teste (Figura 3.7), pois nesse caso sua caixa delimitadora foi construída a partir da fronteira das posições dos seus átomos no modelo atômico da entrada 5JTT, e logo, evitou a inclusão da densidade eletrônica residual de moléculas adjacentes.

Para as entradas do ligante 6MY criadas dentro da faixa de resolução do conjunto de treinamento, até 2.2 Å, o resultado da predição do ligante é muito bom com exceção dos

ruídos adjacentes criados. Fora da faixa de resolução do conjunto de treinamento, acima de 2.2 Å, o modelo ainda retorna bons resultados até 2.8 Å de resolução. Nesse corte de resolução a predição da classe C6 deixa de conter o centro desse ciclo (predito como *background* até então), e a predição do ciclo CA5 central passa a incluir um átomo errado vindo da inclusão da densidade residual de uma água adjacente na imagem do ligante. As demais classes são afetadas a partir de 3 Å de resolução, quando os ciclos C6 e CA5 passam a ser confundidos com ciclos CA6 e C5, respectivamente, e átomos errados são preditos nos seus entornos. A partir de 3.3 Å de resolução o ciclo CA6 não é mais predito e passa a ser confundido com uma "calda" de átomos fora de ciclos. Um gráfico da acurácia média mIoU do modelo do Treinamento 1 - Lig-qRankDB-SP-1.5-2.2 C347CA56 versus a resolução das entradas criadas para o ligante 6MY é mostrado na Figura 3.21.

Figura 3.21: Impacto da resolução para o ligante 6MY da entrada 5JTT do PDB na acurácia média da sua predição com o modelo obtido no Treinamento 1 - Lig-qRankDB-SP-1.5-2.2 C347CA56.

A Figura 3.21 evidencia a queda na acurácia do modelo para o ligante 6MY da entrada 5JTT do PDB a partir de 3 Å de resolução. Apesar dessa analise ter sido feita apenas com uma entrada de ligante e não representar o conjunto inteiro ou um entrada genérica, é possível compreender que a acurácia da aplicação *NP³ Blob Label* pode cair substancialmente para resoluções maiores onde a qualidade da imagem do ligante na densidade é muito baixa ou que estão acima da maior resolução presente no conjunto de treinamento, que para o conjunto Lig-qRankDB-SP-1.5-2.2 C347CA56 utilizado neste exemplo é igual a 2.2 Å. E para este último caso, a partir de 3 Å de resolução o resultado da predição deixa de ser confiável. Esse resultado está diretamente relacionado a qualidade da densidade residual do ligante. Seria necessário repetir essa análise para todas os ligantes do conjunto de teste para propor um valor de corte da resolução das entradas em que a aplicação ainda pode retornar boas predições. Esse resultado está de acordo com a variação da acurácia média do modelo observada em relação a resolução de todas as entradas do conjunto de teste (Figura 3.8).

Os resultados de teste da aplicação *NP³ Blob Label* indicam uma necessidade de se pensar em alternativas para a criação da imagem da caixa delimitadora dos *blobs*. Seria interessante testar outras metodologias nesta etapa para que o tamanho da caixa delimitadora de cada *blob* fique mais adequado para a dimensão da sua imagem na densidade residual, evite a inclusão de ruídos adjacentes e mantenha a acurácia do modelo utilizado. Em relação ao tempo da aplicação, a média de tempo de processamento completo dos *blobs* dos ligantes do conjunto de teste foi de aproximadamente 5 segundos por ligante com o hardware utilizado.

Capítulo 4

Conclusões

Este trabalho representa a primeira solução dentre as encontradas que aplicou aprendizado de máquina profundo na imagem 3D da densidade eletrônica residual de ligantes e esse resultado foi disponibilizado em um workflow chamado de *NP³ Blob Label*. Os resultados obtidos mostraram que é possível treinar um modelo de aprendizado profundo para fazer a segmentação semântica de imagens da densidade eletrônica residual de ligantes e identificar subestruturas químicas que preenchem e explicam a nuvem eletrônica observada. E logo, é possível concluir que as subestruturas químicas das moléculas presentes no cristal de proteína deixam padrões na distribuição da densidade eletrônica residual que possibilitam esse aprendizado.

Outras modelagens podem ser investigadas para relacionar tais padrões com informações estruturais relevantes. A metodologia aplicada fornece um primeiro conjunto de modelagens das imagens dos ligantes na densidade residual, junto com um arcabouço de funções que serão disponibilizadas e poderão ser adaptadas e melhoradas para novas soluções.

Os bancos de dados de imagens 3D da densidade eletrônica residual de ligantes rotuladas criados neste projeto podem ser utilizados em novas aplicações que poderão contribuir para avançar em problemas da comunidade de cristalografia de proteínas.

O esforço da comunidade científica de cristalografia ao longo dos anos, com o auxílio de linhas de luz de raio X automatizadas, pipelines de processamento e análise de dados automatizadas, junto com o grande investimento nessas infraestruturas e os especialistas que desenvolveram e mantêm esses sistemas complexos [105], permitiu que esta grande quantidade de dados de mais de 150 mil estruturas estivessem disponíveis hoje no PDB de forma centralizada, organizada e gratuita. A disponibilidade desses dados, junto com iniciativas como o pacote Gemmi [111] para auxiliar na sua manipulação, permitem que hoje diferentes técnicas de mineração, análise de dados e aprendizado de máquina sejam aplicadas para extrair mais conhecimento e implementar novas soluções para a comunidade de cristalografia. Este trabalho se beneficiou de todo esse esforço para conseguir contribuir com uma nova solução baseada em aprendizado profundo para interpretação da densidade eletrônica residual de ligantes desconhecidos a partir da segmentação da sua imagem em subestruturas químicas.

Os modelos de segmentação da imagem 3D da densidade eletrônica de ligantes obtidos tiveram boa acurácia, apesar de não existir uma referência para um corte de acurácia

aceitável a inspeção visual dos resultados evidenciou que valores acima de 50% ainda podem apresentar predições corretas ou muito próximas do esperado. As modelagens simplificadas foram capazes de identificar a região ocupada pelo ligante com acurácia mIoU igual a 77.4% e identificar átomos e ciclos genéricos com mIoU igual a 70.4%. Quando diferentes tamanhos e tipos de ciclos foram adicionados na modelagem a acurácia média mIoU do modelo obtido foi para 50.5% na validação cruzada "k-fold". Esse modelo foi capaz de identificar ciclos de tamanho 5 e 6, aromáticos ou não, com acurácias de 51 a 64% em termos de IoU. As modelagens baseadas no tipo do átomo tiveram mIoU igual a 39.6% para identificar cada tipo de átomo separadamente e 57.6% para identificar tipos de átomos agrupados. Os padrões presentes nas imagens da densidade eletrônica de ligantes foram suficiente para identificar um átomo de carbono com 60% de acurácia, de oxigênio com 53% e de nitrogênio com quase 52% em termos de IoU. As modelagens baseadas na hibridização SP não apresentaram acurácias interessantes com mIoU de 39.3% para as classes SP e 33.8% para as classes SP e ciclos. Todas as modelagens conseguiram remover ruído de fundo da imagem dos ligantes com uma acurácia acima de 85%. O desbalanço entre as classes pode afetar bastante a acurácia média do modelo e evidencia a dificuldade de lidar com classes raras, este é o caso dos ciclos minoritários, ou difíceis de convergir, como é o caso das classes SP. Esses resultados destacam a importância de começar com modelagens mais simples e a partir de bons resultados avançar para modelagens mais complexas a medida que os limites e capacidades dos modelos são entendidos.

O uso da técnica de *undersampling*, junto com a função de perda SL robusta a ruídos na rotulação e o otimizador SGD foram essenciais para melhorar a convergência e acurácia final dos modelos. As análises sistemáticas apresentadas evidenciam o impacto dos hiperparâmetros utilizados, a importância de diversidade no treinamento para lidar com *overfitting* e o uso de peso na função de perda para contornar pequenos desbalanços.

A aplicação *NP³ Blob Label* desenvolvida neste trabalho permite utilizar os modelos obtidos em uma nova entrada com pouca perda de acurácia, pelo menos 4.7% é esperado. Essa perda de acurácia evidencia uma necessidade de avaliar diferentes formas de criar a imagem dos *blobs* na aplicação e idealmente padronizar a metodologia de criação da caixa delimitadora do *blob* com aquela utilizada na criação dos bancos de dados de imagens de ligantes. O *NP³ Blob Label* é capaz de criar os mapas da densidade residual, buscar por *blobs* não modelados, criar as imagens dos *blobs* encontrados e utilizar os modelos para rotular essas imagens. Combinar os resultados dos modelos para segmentação no tipo de átomo e para segmentação em diferentes tipos de ciclos pode auxiliar bastante na interpretação da densidade. O primeiro modelo pode trazer informações micro sobre os átomos da estrutura do ligante, enquanto o segundo modelo pode trazer informações macro sobre o arranjo dessa estrutura em subestruturas cíclicas. Essa aplicação pode ser aplicada em larga escala para um conjunto grande de coletas de dados de cristalografia e acelerar a varredura desses dados.

Todos esses resultados são muito promissores e incentivam a aplicação dessa metodologia para outros problemas da área. Uma aplicação de aprendizado profundo pode se beneficiar das imagens criadas para, por exemplo, fazer a classificação dos ligantes mais comumente encontrados no PDB, e possivelmente obter uma solução competitiva para o problema enfrentado pelas soluções do Arp/wArp, Phenix e CheckMyBlob. Outra possí-

vel aplicação para a metodologia implementada neste projeto é a segmentação de imagens de densidade eletrônica de proteínas em aminoácidos, que corresponde ao problema de reconstrução da estrutura química de proteínas. Todas essas soluções vêm para auxiliar pesquisadores nas suas análises, as dificuldades encontradas e limitações das soluções fortalecem a importância do conhecimento técnico e especifico desses profissionais para tomada de decisão.

Apesar do PDB fornecer uma quantidade suficiente de dados para treinar um modelo de aprendizado profundo, foi evidenciado que essa base possui algumas limitações quanto a ruído nos dados, enviesamento para estruturas comumente encontradas e a um desbalanço muito grande dependendo da modelagem proposta. Isso alerta a comunidade científica para se atentar ao conjunto de dados utilizado em aplicações de métodos de aprendizado de máquina, para evitar que enviesamentos sejam inseridos nos modelos e acurácias erradas sejam obtidas. À medida que essa base de dados cresce e mais validações são implementadas muitas dessas limitações podem ser resolvidas e novas possibilidades de melhoria e de aplicação podem surgir.

Para o problema de reconstrução da estrutura de ligantes, a solução obtida neste projeto fornece uma primeira automatização nessa direção que possibilita pensar em novas formas de fazer essa reconstrução a partir dos resultados dos modelos. Uma ideia que se apresentou promissora nas discussões deste trabalho e se coloca como uma possibilidade para caminhos futuros, foi a extração do eixo medial da imagem das predições dos modelos para guiar a construção de um esqueleto inicial para o ligante desconhecido. O eixo medial fornece uma estrutura de esqueleto que pode ser povoada com átomos seguindo as predições de cada região para reconstruir as estruturas cíclicas e fazer as conexões entre as partes preditas. A criação de estruturas completas para o ligantes possibilitaria também fazer uma validação do resultado da solução dos modelos mais fiel ao dado de cristalografia de proteínas, a partir da nota de encaixe das estruturas sugeridas. Dessa forma possíveis ruídos na rotulação poderiam ser contornados com uma avaliação direta com o dado experimental.

Com estruturas completas é possível pensar em pipelines para execução de soluções de *fragment-screening* para otimização do encaixe das estruturas propostas e ordenação dos melhores resultados. Soluções de *fragment screening* otimizam a conformação de subestruturas químicas para melhorar suas correlações com a densidade eletrônica observada [64, 100, 39]. A partir de ciclos de refinamento [59, 52], onde a estrutura proposta é validada contra o dado experimental, é possível inferir a acurácia da proposta e então realizar operações de modificação para aperfeiçoamento das estruturas químicas candidatas e melhor preenchimento da densidade eletrônica extra observada.

Na perspectiva do projeto NP^3, a aplicação criada neste trabalho representa mais um passo na direção da inovação necessária para possibilitar que a pesquisa em descoberta de novos fármacos a partir de produtos naturais da biodiversidade brasileira seja facilitada. O NP^3 se baseia em três técnicas diferentes, duas delas, a espectrometria de massas e os ensaios biológicos, já foram integradas em um workflow automatizado chamado NP^3 MS Workflow, descrito no trabalho de Felicio et al. [32] e cujo artigo está em preparação pela autora desta dissertação. Esses trabalhos possibilitam pensar na integração automatizada dos dados de cristalografia de proteínas, que é a terceira técnica utilizada

pela metodologia do NP^3, com as demais técnicas. Uma possibilidade que surge é a de utilizar as subestruturas cíclicas preditas pelos modelos de segmentação para filtrar nos resultados do NP^3 MS Workflow apenas as identificações de moléculas com essas características. Em seguida as moléculas filtradas podem ser encaixadas nos *blobs* encontrados e os bons resultados podem fornecer propostas de estruturas para o ligante desconhecido. Com a sugestão de estruturas completas a partir dos dados de cristalografia de proteínas integrações mais complexas poderão ser avaliadas.

A interdisciplinaridade deste trabalho mostrou a necessidade de estudo e incorporação de conhecimentos da física, biologia e química para se pensar em modelagens computacionais mais complexas capazes de capturar os padrões deixados na imagem da densidade eletrônica.

Na pesquisa de descoberta de novos fármacos a partir da biodiversidade brasileira seria de grande proveito mais investimentos na área de etnofarmacologia. Dessa forma, seria possível fomentar a incorporação desse conhecimento na descoberta de novos fármacos e valorizar todo conhecimento tradicional associado a biodiversidade, cujos povos resguardam e preservam a megabiodiversidade brasileira.

Agências de fomento, como o Serrapilheira que financiou o projeto do NP^3 e viabilizou o trabalho feito neste projeto de mestrado, deveriam investir mais em ciências humanas para que o progresso tecnológico seja capaz de acompanhar os impactos e consequências das novas inovações, e assim, guiar um caminho mais ético na pesquisa brasileira em todas as áreas do conhecimento. Com soluções mais equitativas e que beneficiem toda a população. E mais do que essas agências do chamado terceiro setor, o Estado é o responsável por garantir o investimento necessário para que tais pesquisas aconteçam nas diversas universidades brasileiras e organizações sociais de todo o país, como o CNPEM, e chegue como uma oportunidade para muitos e não para poucos.

Referências Bibliográficas

[1] Pavel V. Afonine, Billy K. Poon, Randy J. Read, Oleg V. Sobolev, Thomas C. Terwilliger, Alexandre Urzhumtsev, and Paul D. Adams. Real-space refinement in *PHENIX* for cryo-EM and crystallography. *Acta Crystallographica Section D*, 74(6):531–544, 2018.

[2] Adeleke H. Aguda, Vincent Lavallee, Ping Cheng, Tina M. Bott, Labros G. Meimetis, Simon Law, Nham T. Nguyen, David E. Williams, Jadwiga Kaleta, Ivan Villanueva, Julian Davies, Raymond J. Andersen, Gary D. Brayer, and Dieter Brömme. Affinity crystallography: A new approach to extracting high-affinity enzyme inhibitors from natural extracts. *Journal of Natural Products*, 79(8):1962–1970, 2016. PMID: 27498895.

[3] Asad Ahmed, Bhavika Mam, and Ramanathan Sowdhamini. Deelig: A deep learning approach to predict protein-ligand binding affinity. *Bioinformatics and Biology Insights*, 15:11779322211030364, 2021. PMID: 34290496.

[4] Eman Ahmed, Alexandre Saint, Abd El Rahman Shabayek, Kseniya Cherenkova, Rig Das, Gleb Gusev, Djamila Aouada, and Bjorn Ottersten. A survey on Deep Learning Advances on Different 3D Data Representations. *arXiv e-prints*, page arXiv:1808.01462, 2018.

[5] Jun Aishima, Daniel Russel, Leonidas Guibas, Paul Adams, and Axel Brunger. Automated crystallographic ligand building using the medial axis transform of an electron-density isosurface. *Acta crystallographica. Section D, Biological crystallography*, 61:1354–63, 2005.

[6] Gorkem Algan and Ilkay Ulusoy. Label noise types and their effects on deep learning. *ArXiv*, abs/2003.10471, 2020.

[7] Görkem Algan and Ilkay Ulusoy. Image classification with deep learning in the presence of noisy labels: A survey. *Knowledge-Based Systems*, 215:106771, 2021.

[8] Todor Kirilov Avramov, Dan Vyenielo, Josue Gomez-Blanco, Swathi Adinarayanan, Javier Vargas, and Dong Si. Deep learning for validating and estimating resolution of cryo-electron microscopy density maps (+). *Molecules (Basel, Switzerland)*, 24(6):1181, 2019. 30917528[pmid].

[9] Stepan Batsanov. Van der waals radii of elements. *Inorganic Materials*, 37:871–885, 2001.

[10] H. M. Berman, J. Westbrook, Z. Feng, G. Gilliland, T. N. Bhat, H. Weissig, I. N. Shindyalov, and P. E. Bourne. The protein data bank. *Nucleic acids research*, 28(1):235–242, 2000. PMC102472[pmcid].

[11] Helen Berman, Kim Henrick, and Haruki Nakamura. Announcing the worldwide protein data bank. *Nature Structural & Molecular Biology*, 10(12):980–980, 2003.

[12] Daniel Berrar. Cross-validation. In Shoba Ranganathan, Michael Gribskov, Kenta Nakai, and Christian Schönbach, editors, *Encyclopedia of Bioinformatics and Computational Biology*, pages 542–545. Academic Press, Oxford, 2019.

[13] Daria A. Beshnova, Joana Pereira, and Victor S. Lamzin. Estimation of the protein–ligand interaction energy for model building and validation. *Acta Crystallographica Section D*, 73(3):195–202, 2017.

[14] Michael M. Bronstein, Joan Bruna, Yann LeCun, Arthur Szlam, and Pierre Vandergheynst. Geometric deep learning: Going beyond euclidean data. *IEEE Signal Processing Magazine*, 34(4):18–42, 2017.

[15] Mateusz Buda, Atsuto Maki, and Maciej A. Mazurowski. A systematic study of the class imbalance problem in convolutional neural networks. *Neural Networks*, 106:249–259, 2018.

[16] C. G. Carolan and V. S. Lamzin. Automated identification of crystallographic ligands using sparse-density representations. *Acta Crystallographica Section D*, 70(7):1844–1853, 2014.

[17] Roberto Cavararo. Censo demogrÁfico 2010. características gerais da população, religião e pessoas com deficiência. *Instituto Brasileiro de Geografia e Estatística - IBGE*, 2012. Online; accessed 15 June 2022.

[18] Nitesh Chawla, Kevin Bowyer, Lawrence Hall, and W. Kegelmeyer. Smote: Synthetic minority over-sampling technique. *J. Artif. Intell. Res. (JAIR)*, 16:321–357, 2002.

[19] Grzegorz Chojnowski, Joana Pereira, and Victor S. Lamzin. Sequence assignment for low-resolution modelling of protein crystal structures. *Acta Crystallographica Section D*, 75(8):753–763, 2019.

[20] Christopher Choy, JunYoung Gwak, and Silvio Savarese. 4d spatio-temporal convnets: Minkowski convolutional neural networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3075–3084, 2019.

[21] Bochev D. and Rouvray D.H. *Chemical Graph Theory.* CRC Press; 1 edition, 1991.

[22] Manuela Carneiro da Cunha, Sônia Barbosa Magalhães, and Cristina Adams. Alguns direitos especÍficos na legislaÇÃo brasileira. In *Povos tradicionais e biodiversidade no Brasil – Contribuições dos povos indígenas, quilombolas e comunidades*

tradicionais para a biodiversidade, políticas e ameaças, volume 1 of *PARTE I. TERRITÓRIOS E DIREITOS DOS POVOS INDÍGENAS, QUILOMBOLAS E COMUNIDADES TRADICIONAIS*, chapter 4. Sociedade Brasileira para o Progresso da Ciência - SBPC, Rua Maria Antonia, 294 - 4o andar - Vila Buarque - 01222-010 São Paulo - SP - Brasil, 2021/2022.

[23] Manuela Carneiro da Cunha, Sônia Barbosa Magalhães, and Cristina Adams. Biodiversidade e agrobiodiversidade como legados de povos indÍgenas. In *Povos tradicionais e biodiversidade no Brasil – Contribuições dos povos indígenas, quilombolas e comunidades tradicionais para a biodiversidade, políticas e ameaças*, volume 2 of *PARTE II. CONTRIBUIÇÃO DOS POVOS INDÍGENAS, QUILOMBOLAS E COMUNIDADES TRADICIONAIS À BIODIVERSIDADE*, chapter 6. Sociedade Brasileira para o Progresso da Ciência - SBPC, Rua Maria Antonia, 294 - 4o andar - Vila Buarque - 01222-010 São Paulo - SP - Brasil, 2021/2022.

[24] Manuela Carneiro da Cunha, Sônia Barbosa Magalhães, and Cristina Adams. Conhecimentos associados À biodiversidade. In *Povos tradicionais e biodiversidade no Brasil – Contribuições dos povos indígenas, quilombolas e comunidades tradicionais para a biodiversidade, políticas e ameaças*, volume 2 of *PARTE II. CONTRIBUIÇÃO DOS POVOS INDÍGENAS, QUILOMBOLAS E COMUNIDADES TRADICIONAIS À BIODIVERSIDADE*, chapter 8. Sociedade Brasileira para o Progresso da Ciência - SBPC, Rua Maria Antonia, 294 - 4o andar - Vila Buarque - 01222-010 São Paulo - SP - Brasil, 2021/2022.

[25] Manuela Carneiro da Cunha, Sônia Barbosa Magalhães, and Cristina Adams. Dificuldades na efetivaÇÃo dos direitos territoriais. In *Povos tradicionais e biodiversidade no Brasil – Contribuições dos povos indígenas, quilombolas e comunidades tradicionais para a biodiversidade, políticas e ameaças*, volume 1 of *PARTE I. TERRITÓRIOS E DIREITOS DOS POVOS INDÍGENAS, QUILOMBOLAS E COMUNIDADES TRADICIONAIS*, chapter 3. Sociedade Brasileira para o Progresso da Ciência - SBPC, Rua Maria Antonia, 294 - 4o andar - Vila Buarque - 01222-010 São Paulo - SP - Brasil, 2021/2022.

[26] Manuela Carneiro da Cunha, Sônia Barbosa Magalhães, and Cristina Adams. Os territÓrios indÍgenas e tradicionais protegem a biodiversidade? In *Povos tradicionais e biodiversidade no Brasil – Contribuições dos povos indígenas, quilombolas e comunidades tradicionais para a biodiversidade, políticas e ameaças*, volume 2 of *PARTE II. CONTRIBUIÇÃO DOS POVOS INDÍGENAS, QUILOMBOLAS E COMUNIDADES TRADICIONAIS À BIODIVERSIDADE*, chapter 5. Sociedade Brasileira para o Progresso da Ciência - SBPC, Rua Maria Antonia, 294 - 4o andar - Vila Buarque - 01222-010 São Paulo - SP - Brasil, 2021/2022.

[27] Manuela Carneiro da Cunha, Sônia Barbosa Magalhães, and Cristina Adams. Povos indÍgenas. In *Povos tradicionais e biodiversidade no Brasil – Contribuições dos povos indígenas, quilombolas e comunidades tradicionais para a biodiversidade, políticas e ameaças*, volume 6 of *PARTE VI. PESQUISAS INTERCULTURAIS*, chapter 15.

Sociedade Brasileira para o Progresso da Ciência - SBPC, Rua Maria Antonia, 294 - 4o andar - Vila Buarque - 01222-010 São Paulo - SP - Brasil, 2021/2022.

[28] Manuela Carneiro da Cunha, Sônia Barbosa Magalhães, and Cristina Adams. Quem são, quantos são. In *Povos tradicionais e biodiversidade no Brasil – Contribuições dos povos indígenas, quilombolas e comunidades tradicionais para a biodiversidade, políticas e ameaças*, volume 1 of *PARTE I. TERRITÓRIOS E DIREITOS DOS POVOS INDÍGENAS, QUILOMBOLAS E COMUNIDADES TRADICIONAIS*, chapter 1. Sociedade Brasileira para o Progresso da Ciência - SBPC, Rua Maria Antonia, 294 - 4o andar - Vila Buarque - 01222-010 São Paulo - SP - Brasil, 2021/2022.

[29] A. Dai, A. X. Chang, M. Savva, M. Halber, T. Funkhouser, and M. Niessner. Scannet: Richly-annotated 3d reconstructions of indoor scenes. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2432–2443, Los Alamitos, CA, USA, 2017. IEEE Computer Society.

[30] A. H. Dallagnol, G. T. Soldati, and M. T. Silva. Nossos conhecimentos sobre a sociobiodiversidade: salvaguardando uma herança ancestral. `https://agroecologia.org.br/wp-content/uploads/2020/05/Cartilha-Sociobiodiversidade-web-1.pdf`, 2020.

[31] Zbigniew Dauter, Alexander Wlodawer, Wladek Minor, Mariusz Jaskolski, and Bernhard Rupp. Avoidable errors in deposited macromolecular structures: an impediment to efficient data mining. *IUCrJ*, 1(Pt 3):179–193, 2014. 25075337[pmid].

[32] Rafael de Felício, Patricia Ballone, Cristina Freitas Bazzano, Luiz F. G. Alves, Renata Sigrist, Gina Polo Infante, Henrique Niero, Fernanda Rodrigues-Costa, Arthur Zanetti Nunes Fernandes, Luciane A. C. Tonon, Luciana S. Paradela, Renna Karoline Eloi Costa, Sandra Martha Gomes Dias, Andréa Dessen, Guilherme P. Telles, Marcus Adonai Castro da Silva, Andre Oliveira de Souza Lima, and Daniela Barretto Barbosa Trivella. Chemical elicitors induce rare bioactive secondary metabolites in deep-sea bacteria under laboratory conditions. *Metabolites*, 11(2), 2021.

[33] Marc Deller and Bernhard Rupp. Models of protein-ligand crystal structures: Trust, but verify. *Journal of computer-aided molecular design*, 29, 2015.

[34] Antonio Carlos Diegues, Rinaldo Sergio Vieira Arruda, Viviane Capezzuto Ferreira da Silva, Francisca Aida Barboza Figols, and Daniela Andrade. *Biodiversidade e Comunidades Tradicionais no Brasil*. MINISTÉRIO DO MEIO AMBIENTE, DOS RECURSOS HÍDRICOS E DA AMAZÔNIA LEGAL and NUPAUB - NÚCLEO DE PESQUISAS SOBRE POPULAÇÕES HUMANAS E ÁREAS ÚMIDAS BRASILEIRAS—UNIVERSIDADE DE SÃO PAULO, 1999.

[35] Vincent Dumoulin and Francesco Visin. A guide to convolution arithmetic for deep learning. *ArXiv*, abs/1603.07285, 2016.

[36] Shuchismita Dutta, Kyle Burkhardt, Jasmine Young, Ganesh J. Swaminathan, Takanori Matsuura, Kim Henrick, Haruki Nakamura, and Helen M. Berman. Data

deposition and annotation at the worldwide protein data bank. *Molecular Biotechnology*, 42(1):1–13, 2009.

[37] Paul Emsley. Tools for ligand validation in coot. *Acta crystallographica. Section D, Structural biology*, 73(Pt 3):203–210, 2017. 28291755[pmid].

[38] Paul Emsley and Kevin Cowtan. *Coot*: model-building tools for molecular graphics. *Acta Crystallographica Section D*, 60(12 Part 1):2126–2132, 2004.

[39] Paul Emsley, Bernhard Lohkamp, William G. Scott, and Kevin Cowtan. Features and development of coot. *Acta Crystallographica Section D - Biological Crystallography*, 66:486–501, 2010.

[40] Philip R. Evans. An introduction to stereochemical restraints. *Acta crystallographica. Section D, Biological crystallography*, 63(Pt 1):58–61, 2007. S090744490604604X[PII].

[41] William Falcon and The PyTorch Lightning team. PyTorch Lightning, 2019.

[42] Raquel Ortega Ferreira. Uso combinado de cristalografia de proteínas e espectrometria de massas para a seleção antecipada de produtos naturais bioativos. Master's thesis, Universidade Estadual de Campinas, 2018.

[43] Priya Goyal, Piotr Dollár, Ross Girshick, Pieter Noordhuis, Lukasz Wesolowski, Aapo Kyrola, Andrew Tulloch, Yangqing Jia, and Kaiming He. Accurate, large minibatch sgd: Training imagenet in 1 hour, 2017.

[44] J. A. Grant and B. T. Pickup. A gaussian description of molecular shape. *The Journal of Physical Chemistry*, 99(11):3503–3510, 1995.

[45] Yulan Guo, Hanyun Wang, Qingyong Hu, Hao Liu, Li Liu, and Mohammed Bennamoun. Deep learning for 3d point clouds: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 43(12):4338–4364, 2021.

[46] George G. Hall and David Martin. Approximate electron densities for atoms and molecules. *Israel Journal of Chemistry*, 19(1-4):255–259, 1980.

[47] Johan Hattne and Victor S. Lamzin. Pattern-recognition-based detection of planar objects in three-dimensional electron-density maps. *Acta Crystallographica Section D*, 64(8):834–842, 2008.

[48] P W Hawkes. Digital image processing. *Nature*, 276(5689):740–740, 1978.

[49] Coordenação de Recursos Naturais e Estudos Ambientais IBGE. *Biomas e sistema costeiro-marinho do Brasil*, volume 45. IBGE, 2019.

[50] T5 Informatics. Rdkit: Open-source cheminformatics software. `https://www.rdkit.org/`, 2012.

[51] Ajay N. Jain, Ann E. Cleves, Alexander C. Brueckner, Charles A. Lesburg, Qiaolin Deng, Edward C. Sherer, and Mikhail Y. Reibarkh. Xgen: Real-space fitting of complex ligand conformational ensembles to x-ray electron density maps. *Journal of Medicinal Chemistry*, 63(18):10509–10528, 2020. PMID: 32877178.

[52] Pawel A. Janowski, Nigel W. Moriarty, Brian P. Kelley, David A. Case, Darrin M. York, Paul D. Adams, and Gregory L. Warren. Improved ligand geometries in crystallographic refinement using *AFITT* in *PHENIX*. *Acta Crystallographica Section D*, 72(9):1062–1072, 2016.

[53] J Jiménez, S Doerr, G Martínez-Rosell, A S Rose, and G De Fabritiis. DeepSite: protein-binding site predictor using 3D-convolutional neural networks. *Bioinformatics*, 33(19):3036–3042, 2017.

[54] Wengong Jin, Regina Barzilay, and Tommi Jaakkola. Junction tree variational autoencoder for molecular graph generation. *arXiv*, 2019.

[55] Justin Johnson and Taghi Khoshgoftaar. Survey on deep learning with class imbalance. *Journal of Big Data*, 6:27, 2019.

[56] Davood Karimi, Haoran Dou, Simon K. Warfield, and Ali Gholipour. Deep learning with noisy labels: Exploring techniques and remedies in medical image analysis. *Medical Image Analysis*, 65:101759, 2020.

[57] Salman Khan, Mohammed Bennamoun, Ferdous Sohel, and Roberto Togneri. Cost-sensitive learning of deep feature representations from imbalanced data. *IEEE Transactions on Neural Networks and Learning Systems*, PP, 2015.

[58] AI Kitaygorodskiy. *Molekularnye kristally*. Nauka, 1971.

[59] Gerard J. Kleywegt and T. Alwyn Jones. [11] model building and refinement practice. In *Macromolecular Crystallography Part B*, volume 277 of *Methods in Enzymology*, pages 208 – 230. Academic Press, 1997.

[60] Huub Kooijman. Interpretation of crystal structure determinations. `http://www.cryst.chem.uu.nl/huub/notesweb.pdf`, 2005. Online; accessed 04 October 2019.

[61] Marcin Kowiel, Dariusz Brzezinski, Przemyslaw J Porebski, Ivan G Shabalin, Mariusz Jaskolski, and Wladek Minor. Automatic recognition of ligands in electron density by machine learning. *Bioinformatics*, 35(3):452–461, 2018.

[62] Tobias Krojer, Romain Talon, Nicholas Pearce, Patrick Collins, Alice Douangamath, Jose Brandao-Neto, Alexandre Dias, Brian Marsden, and Frank von Delft. The *XChemExplorer* graphical workflow tool for routine or large-scale protein–ligand structure determination. *Acta Crystallographica Section D*, 73(3):267–278, 2017.

[63] Audrey L. Lamb, T. Joseph Kappock, and Nicholas R. Silvaggi. You are lost without a map: Navigating the sea of protein structures. *Biochimica et Biophysica Acta (BBA) - Proteins and Proteomics*, 1854(4):258–268, 2015.

[64] Gerrit G. Langer, Guillaume X. Evrard, Ciaran G. Carolan, and Victor S. Lamzin. Fragmentation-tree density representation for crystallographic modelling of bound ligands. *Journal of Molecular Biology*, 419(3):211 – 222, 2012.

[65] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. Deep learning. *Nature*, 521(7553):436–444, 2015.

[66] Laurence Leherte, Janice Glasgow, Kimberley Baxter, Evan Steeg, and Suzanne Fortier. Analysis of three-dimensional protein images. *Journal of Artificial Intelligence Research*, 7:125–159, 1997.

[67] Jun Li, Wei Zhu, Jun Wang, Wenfei Li, Sheng Gong, Jian Zhang, and Wei Wang. Rna3dcnn: Local and global quality assessments of rna 3d structures using 3d deep convolutional neural networks. *PLOS Computational Biology*, 14(11):1–18, 2018.

[68] Shen Li, Yanli Zhao, Rohan Varma, Omkar Salpekar, Pieter Noordhuis, Teng Li, Adam Paszke, Jeff Smith, Brian Vaughan, Pritam Damania, and Soumith Chintala. Pytorch distributed: Experiences on accelerating data parallel training. *Proc. VLDB Endow.*, 13(12):3005–3018, 2020.

[69] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. Microsoft coco: Common objects in context. In David Fleet, Tomas Pajdla, Bernt Schiele, and Tinne Tuytelaars, editors, *Computer Vision – ECCV 2014*, pages 740–755, Cham, 2014. Springer International Publishing.

[70] Max A Little, Gael Varoquaux, Sohrab Saeb, Luca Lonini, Arun Jayaraman, David C Mohr, and Konrad P Kording. Using and understanding cross-validation strategies. Perspectives on Saeb et al. *GigaScience*, 6(5), 2017. gix020.

[71] Sai Raghavendra Maddhuri Venkata Subramaniya, Genki Terashi, and Daisuke Kihara. Protein secondary structure detection in intermediate-resolution cryo-em maps using deep learning. *Nature Methods*, 16(9):911–917, 2019.

[72] KARLA DO NASCIMENTO MAGALHÃES. *PLANTAS MEDICINAIS DA CAATINGA DO NORDESTE BRASILEIRO: ETNOFARMACOPEIA DO PROFESSOR FRANCISCO JOSÉ DE ABREU MATOS*. PhD thesis, Desenvolvimento e Inovação Tecnológica de Medicamentos da Universidade Federal do Ceará, 2019.

[73] Dominic Masters and Carlo Luschi. Revisiting small batch training for deep neural networks. *ArXiv*, abs/1804.07612, 2018.

[74] João Mello, Cláudia Simões, Eloir Schenkel, Grace Gosmann, Lilian Mentz, and Pedro Petrovick. *Farmacognosia da Planta ao Medicamento*. Editora da Universidade - Universidade Federal do Rio Grande do Sul, 1999.

[75] W. H. Miller. *A treatise on crystallography*. J. and J. J. Deighton / John W. Parker, 1839.

[76] Marius Muja and David G. Lowe. Fast approximate nearest neighbors with automatic algorithm configuration. In *In VISAPP International Conference on Computer Vision Theory and Applications*, pages 331–340, 2009.

[77] Allan H. Murphy. The finley affair: A signal event in the history of forecast verification. *Weather and Forecasting*, 11(1):3 – 20, 1996.

[78] Garib N. Murshudov, Pavol Skubák, Andrey A. Lebedev, Navraj S. Pannu, Roberto A. Steiner, Robert A. Nicholls, Martyn D. Winn, Fei Long, and Alexei A. Vagin. *REFMAC*5 for the refinement of macromolecular crystal structures. *Acta Crystallographica Section D*, 67(4):355–367, 2011.

[79] David J. Newman and Gordon M. Cragg. Natural products as sources of new drugs from 1981 to 2014. *Journal of Natural Products*, 79(3):629–661, 2016. PMID: 26852623.

[80] David J. Newman and Gordon M. Cragg. Natural products as sources of new drugs over the nearly four decades from 01/1981 to 09/2019. *Journal of Natural Products*, 83(3):770–803, 2020. PMID: 32162523.

[81] T. J. Oldfield. *X-LIGAND*: an application for the automated addition of flexible ligands into electron density. *Acta Crystallographica Section D*, 57(5):696–705, 2001.

[82] Marcin Pacholczyk and Marek Kimmel. Exploring the landscape of protein-ligand interaction energy using probabilistic approach. *Journal of Computational Biology*, 18(6):843–850, 2011. PMID: 21091064.

[83] Martin Papenberg and Gunnar Klau. Using anticlustering to partition data sets into equivalent parts. *Psychological Methods*, 26, 2020.

[84] Nicholas M. Pearce, Tobias Krojer, Anthony R. Bradley, Patrick Collins, Radoslaw P. Nowak, Romain Talon, Brian D. Marsden, Sebastian Kelm, Jiye Shi, Charlotte M. Deane, and Frank von Delft. A multi-crystal method for extracting obscured crystallographic states from conventionally uninterpretable electron density. *Nature Communications*, 8(1):15123, 2017.

[85] Nicholas M. Pearce, Tobias Krojer, and Frank von Delft. Proper modelling of ligand binding requires an ensemble of bound and unbound states. *Acta Crystallographica Section D*, 73(3):256–266, 2017.

[86] Mariana Pereira, Irene Fantini, Roberto Lotufo, and Leticia Rittner. An extended-2D CNN for multiclass Alzheimer's Disease diagnosis through structural MRI. In Horst K. Hahn and Maciej A. Mazurowski, editors, *Medical Imaging 2020: Computer-Aided Diagnosis*, volume 11314, pages 438 – 444. International Society for Optics and Photonics, SPIE, 2020.

[87] P G Polishchuk, T I Madzhidov, and A Varnek. Estimation of the size of drug-like chemical space based on gdb-17 data. *Journal Of Computer-Aided Molecular Design*, 27(8):675 – 679, 2013.

[88] Edwin Pozharski, Christian X. Weichenberger, and Bernhard Rupp. Techniques, tools and best practices for ligand electron-density analysis and results from their application to deposited crystal structures. *Acta Crystallographica Section D*, 69(2):150–167, 2013.

[89] C. Qi, Hao Su, Kaichun Mo, and Leonidas J. Guibas. Pointnet: Deep learning on point sets for 3d classification and segmentation. *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 77–85, 2017.

[90] C. Qi, L. Yi, Hao Su, and Leonidas J. Guibas. Pointnet++: Deep hierarchical feature learning on point sets in a metric space. In *NIPS*, 2017.

[91] Randy J. Read. Pushing the boundaries of molecular replacement with maximum likelihood. *Acta Crystallographica Section D*, 57(10):1373–1382, Oct 2001.

[92] A. Rezende and M. Ribeiro. Conhecimento tradicional, plantas medicinais e propriedade intelectual: biopirataria ou bioprospecção? *RBPM - Revista Brasileira de Plantas Medicinais*, 3:37–44, 2005.

[93] Enio Antunes Rezende. *Biopirataria ou bioprospecção? Uma análise crítica da gestão do saber tradicional no Brasil.* PhD thesis, Escola de Administração - Universidade Federal da Bahia, 2008.

[94] David Rolnick, Andreas Veit, Serge J. Belongie, and Nir Shavit. Deep learning is robust to massive label noise. *ArXiv*, abs/1705.10694, 2017.

[95] Hattie Thompson Small and Brown. Handling unbalanced data in deep image segmentation. In *Proceedings of the 30th International Conference on Computer Graphics and Machine Vision*, 2017.

[96] Emily Smith, Gwyndaf Evans, and James Foadi. An effective introduction to structural crystallography using 1d gaussian atoms. *European Journal of Physics*, 38(6):065501, 2017.

[97] Diamond Light Source and CCP4-Harwell. Dimple - mx pipeline. `https://ccp4.github.io/dimple/`, 2017.

[98] Garry L. Taylor. Introduction to phasing. *Acta Crystallographica Section D*, 66(4):325–338, 2010.

[99] Thomas C. Terwilliger, Paul D. Adams, Nigel W. Moriarty, and Judith D. Cohn. Ligand identification using electron-density map correlations. *Acta Crystallographica Section D*, 63(1):101–107, 2007.

[100] Thomas C. Terwilliger, Herbert Klei, Paul D. Adams, Nigel W. Moriarty, and Judith D. Cohn. Automated ligand fitting by core-fragment fitting and extension into density. *Acta crystallographica. Section D, Biological crystallography*, 62(Pt 8):915–922, 2006. S0907444906017161[PII].

[101] Ian J. Tickle. Statistical quality indicators for electron-density maps. *Acta Crystallographica Section D*, 68(4):454–467, 2012.

[102] Wen Torng and Russ B. Altman. 3d deep convolutional neural networks for amino acid environment similarity analysis. *BMC Bioinformatics*, 18(1):302, 2017.

[103] Wen Torng and Russ B Altman. High precision protein functional site detection using 3D convolutional neural networks. *Bioinformatics*, 35(9):1503–1512, 2018.

[104] Alexandre Urzhumtsev, Pavel V. Afonine, Vladimir Y. Lunin, Thomas C. Terwilliger, and Paul D. Adams. Metrics for comparison of crystallographic maps. *Acta Crystallographica Section D*, 70(10):2593–2606, 2014.

[105] Melanie Vollmar and Gwyndaf Evans. Machine learning applications in macromolecular x-ray crystallography. *Crystallography Reviews*, 27(2):54–101, 2021.

[106] Izhar Wallach, Michael Dzamba, and Abraham Heifets. Atomnet: A deep convolutional neural network for bioactivity prediction in structure-based drug discovery. *ArXiv*, 2015.

[107] Panqu Wang, Pengfei Chen, Ye Yuan, Ding Liu, Zehua Huang, Xiaodi Hou, and G. Cottrell. Understanding convolution for semantic segmentation. *2018 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 1451–1460, 2018.

[108] Xiao Wang, Genki Terashi, Charles W Christoffer, Mengmeng Zhu, and Daisuke Kihara. Protein docking model evaluation by 3d deep convolutional neural networks. *Bioinformatics*, 2019.

[109] Yisen Wang, Xingjun Ma, Zaiyi Chen, Yuan Luo, Jinfeng Yi, and James Bailey. Symmetric cross entropy for robust learning with noisy labels. In *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 322–330, 2019.

[110] Martyn D. Winn, Charles C. Ballard, Kevin D. Cowtan, Eleanor J. Dodson, Paul Emsley, Phil R. Evans, Ronan M. Keegan, Eugene B. Krissinel, Andrew G. W. Leslie, Airlie McCoy, Stuart J. McNicholas, Garib N. Murshudov, Navraj S. Pannu, Elizabeth A. Potterton, Harold R. Powell, Randy J. Read, Alexei Vagin, and Keith S. Wilson. Overview of the *CCP*4 suite and current developments. *Acta Crystallographica Section D*, 67(4):235–242, 2011.

[111] Marcin Wojdyr. Gemmi: A library for structural biology. *Journal of Open Source Software*, 7(73):4200, 2022.

[112] Donna Xu, Yaxin Shi, Ivor W. Tsang, Yew-Soon Ong, Chen Gong, and Xiaobo Shen. Survey on multi-output learning. *IEEE Transactions on Neural Networks and Learning Systems*, 31(7):2409–2429, 2020.

[113] Ma Yi-de, Liu Qing, and Qian Zhi-bai. Automated image segmentation using improved pcnn model based on cross-entropy. In *Proceedings of 2004 International Symposium on Intelligent Multimedia, Video and Speech Processing, 2004.*, pages 743–746, 2004.

[114] Fisher Yu and Vladlen Koltun. Multi-scale context aggregation by dilated convolutions. In *International Conference on Learning Representations (ICLR)*, 2016.

[115] Xingrui Yu, Bo Han, Jiangchao Yao, Gang Niu, Ivor Wai-Hung Tsang, and Masashi Sugiyama. How does disagreement help generalization against label corruption? In *ICML*, 2019.

[116] Qian-Yi Zhou, Jaesik Park, and Vladlen Koltun. Open3D: A modern library for 3D data processing. *arXiv:1801.09847*, 2018.

[117] P. H. Zwart, G. G. Langer, and V. S. Lamzin. Modelling bound ligands in protein crystal structures. *Acta Crystallographica Section D*, 60(12 Part 1):2230–2239, 2004.

[118] Özgün Çiçek, Ahmed Abdulkadir, Soeren S. Lienkamp, Thomas Brox, and Olaf Ronneberger. 3d u-net: Learning dense volumetric segmentation from sparse annotation, 2016.