Boltz-1

Democratizing Biomolecular Interaction Modeling

Jeremy Wohlwend[*12], Gabriele Corso[*12], Saro Passaro[*12],
Mateo Reveiz[12], Ken Leidal[3], Wojtek Swiderski[3], Tally Portnoi[12],
Itamar Chinn[12], Jacob Silterra[12], Tommi Jaakkola[12], Regina Barzilay[12]

Correspondence to {jwohlwend,gcorso,saro00}@csail.mit.edu

Abstract

Understanding biomolecular interactions is fundamental to advancing fields like drug discovery and protein design. In this paper, we introduce Boltz-1, an open-source deep learning model incorporating innovations in model architecture, speed optimization, and data processing achieving AlphaFold3-level accuracy in predicting the 3D structures of biomolecular complexes. Boltz-1 demonstrates a performance on-par with state-of-the-art commercial models on a range of diverse benchmarks, setting a new benchmark for commercially accessible tools in structural biology. By releasing the training and inference code, model weights, datasets, and benchmarks under the MIT open license, we aim to foster global collaboration, accelerate discoveries, and provide a robust platform for advancing biomolecular modeling.

Contents

1 Overview **2**

2 Data pipeline **3**

2.1 Data source and processing 3

2.2 Validation and test sets curation 3

2.3 Dense MSA pairing algorithm 4

2.4 Unified cropping algorithm 4

2.5 Robust pocket-conditioning 5

3 Modeling **5**

3.1 Architectural modifications 6

3.2 Training and inference procedures 6

3.3 Confidence model 8

3.4 Optimizations 8

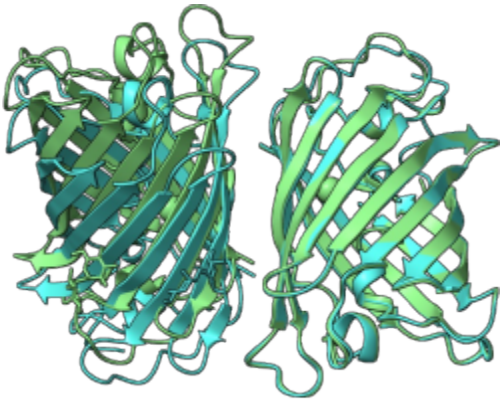
4 Results 9

5 Limitations 11

6 Conclusion 12

7 Acknowledgments 12

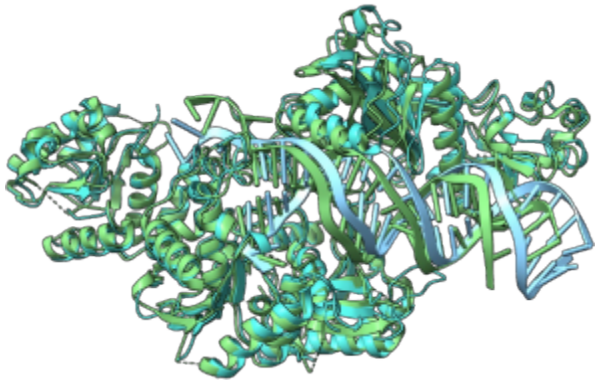
*Equal contribution, [1] MIT CSAIL, [2] MIT Jameel Clinic, [3] Genesis Research, a part of Genesis Therapeutics



BOLTZ-1 prediction on PDB structure 8ILL

LDDT: 0.87

TM-score: 0.92



BOLTZ-1 prediction on PDB structure 8IFK

LDDT: 0.75

TM-score: 0.94

Figure 1: Example predictions of Boltz-1 on targets from the test set.

1 Overview

Biomolecular interactions drive almost all biological mechanisms, and our ability to understand these interactions guides the development of new therapeutics and the discovery of disease drivers. In 2020, AlphaFold2 [Jumper et al., 2021] demonstrated that deep learning models can reach experimental accuracy for single-chain protein structure prediction on a large class of protein sequences. However, a critical question about modeling biomolecular complexes in 3D space remained open.

In the past few years, the research community has made significant progress toward solving this pivotal problem. In particular, the use of deep generative models has proven to be effective in modeling the interaction between different biomolecules with DiffDock [Corso et al., 2022] showing significant improvements over traditional molecular docking approaches and, most recently, AlphaFold3 [Abramson et al., 2024] reaching unprecedented accuracy in the prediction of arbitrary biomolecular complexes.

In this manuscript, we present Boltz-1, the first fully commercially accessible open-source model reaching AlphaFold3 reported levels of accuracy. By making the training and inference code, model weights, datasets, and benchmarks freely available under the MIT license, we aim to empower researchers, developers, and organizations around the world to experiment, validate, and innovate with Boltz-1. At a high level, Boltz-1 follows the general framework and architecture presented by Abramson et al. [2024], but it also presents several innovations which include:

1. New algorithms to more efficiently and robustly pair MSAs, crop structure at training time, and condition predictions on user-defined binding pockets;
2. Changes to the flow of the representations in the architecture and the diffusion training and inference procedures;
3. Revision of the confidence model both in terms of architectural components as well as the framing of the task as a fine-tuning of the model's trunk layers.

In the following sections, we detail these changes as well as benchmark the performance of Boltz-1 with other publicly available models. Our experimental results show that Boltz-1 delivers performance on par with the state-of-the-art commercial models on a wide range of structures and metrics.

Given the dynamic nature of this open-source project, this manuscript and its linked GitHub repository[1] will be regularly updated with improvements from our core team and the community. We aspire for this project and its associated codebase to serve as a catalyst for advancing our understanding of biomolecular interactions and a driver for the design of novel biomolecules.

[1] https://github.com/jwohlwend/boltz

2 Data pipeline

Boltz-1 operates on proteins represented by their amino acid sequence, ligands represented by their smiles strings (and covalent bonds), and nucleic acids represented by their genomic sequence. This input is then augmented by adding multiple sequence alignment (MSA) and predicted molecular conformations. Unlike AlphaFold3, we do not include input templates, due to their limited impact on the performance of large models.

In this section, we first outline how the structural training data, as well as the MSA and conformer, were obtained and describe the curation of our validation and test sets. Then, we describe three important algorithmic developments applied to data curation and augmentation that we find to be critical:

1. A new algorithm to pair MSAs for multimeric protein complexes from taxonomy information (2.3)
2. A unified cropping algorithm that combines the spatial and contiguous cropping strategies used in previous work (2.4)
3. A robust pocket-conditioning algorithm tailored to common use cases (2.5)

2.1 Data source and processing

PDB structural data For training we use all PDB structures [Berman et al., 2000] released before 2021-09-30 (same training cut-off date as AlphaFold3) and with a resolution of at least 9A˚. We parse the Biological Assembly 1 from these structures from their mmCIF file. For each polymer chain, we use the reference sequence and align it to the residues available in the structure. For ligands, we use the CCD dictionary to create the conformers and to match atoms from the structure. We remove leaving atoms when (1) the ligand is covalently bound and (2) that atom does not appear in the PDB structure. Finally, we follow the same process as AlphaFold3 for data cleaning, which includes the ligand exclusion list, the minimum number of resolved residues, and the removal of clashing chains.

MSA and molecular conformers We construct MSAs for the full PDB data using the colabfold_search tool [Mirdita et al., 2022] (which leverages MMseqs2 [Steinegger and S¨oding, 2017]), using default parameters (versions: uniref30_2302, colabfold_envdb_202108). We then assign taxonomy labels to all UniRef sequences using the taxonomy annotation provided by UniProt [Consortium, 2015]. For the initial molecular conformers that are provided to the model, we pre-compute a single conformer for all CCD codes using the RDKit's ETKDGv3 [Wang et al., 2022].

Structure prediction training pipeline We train the structure prediction model (see Section 3.2 for details of the confidence model training) for a total of 68k steps with a batch size of 128. During the first 53k iterations, we use a crop size of 384 tokens and 3456 atoms and draw structures equally from the PDB dataset and the OpenFold distillation dataset (approximately 270K structures, using the MSAs they provided) [Ahdritz et al.,2024]. For the last 15k iterations, we only sampled from the PDB structures and had a crop size of 512 tokens and 4608 atoms. As a comparison AlphaFol3 trained a similar architecture for nearly 150k steps with a batch size of 256, which required approximately four times the computing time. We attribute some of this drastic reduction to the various innovations we detail in the remainder of this section and the next.

2.2 Validation and test sets curation

To address the absence of a standardized benchmark for all-atom structures, we are releasing a new PDB split designed to help the community converge on reliable and consistent benchmarks for all-atom structure prediction tasks.

Our training, validation and test splitting strategy largely follows Abramson et al. [2024]. We first cluster the protein sequences in PDB by sequence identity with the command mmseqs easy-cluster