Curriculum learning for personalized structural variant embeddings from population scale datasets

Jacob Krol

1 Abstract

Prioritizing pathological structural variants (SVs) in cancer genomes is challenging due to the sheer amount of genomic variation in cancer genomes. Population scale genomic datasets have been used to contrast healthy variants from cancer variants, but these methods require specialized data structures, algorithms, and scientists to exhaustively compare SV evidence across populations. This work explores the efficacy of personalized embeddings derived from SVs for applications in rare cancer variant prioritization and reducing the computational burden of population scale SV analyses.

2 Introduction

SVs are large genomic mutations which often drive cancer onset and progression [3]. However, the sheer amount of SVs present in cancer genomes makes prioritizing pathological variants challenging for diagnosis and treatment. An effective method to prioritize SVs is to compare SV frequencies between cancer and healthy populations, such as the Pan-Cancer Analysis of Whole Genomes [5] (PCAWG) (cancer) or 1000 genomes project [4] (healthy). Ranking the differentially present variants in tumor samples versus the healthy populations increases sensitivity and precision of verifying pathological SVs [2]. However, performing population scale prioritization of variants is computationally expensive and requires specialized tools [2] , databases [8], and specialized scientists to perform the analysis. This size of databases will continuously increase, and it is becoming a greater challenge to perform computationally tractable analyses. Latent variables commonly called embeddings are often used to represent large, compelx data with, generally, lower dimensional representations. Among many embedding methods, Siamese neural networks are specialized for learning representations from paired inputs with applications in signature [1] and face verification [9]. In this work, a Siamese neural network will train on a symbolic curriculum learning objective to learn low-dimensional, personalized embedding representations from SVs using population scale datasets. The novel contributions of this work include 1) the first model to embed individuals by SVs, 2) a novel loss function to minimize the difference between embedding distances and the kullback-leibler divergence of paired-input SV distributions, and 3) a symbolic curriculum learning objective which requires the model to learn representations within and across populations. An effective SV embedding model for individuals has personalized medicine applications such as gathering evidence about similar cancer patients to assist in diagnosis, treatments, and fundamental understanding of genomic variation in cancer.

3 Related work

Population filtering of variants required development of tools to effectively handle the large scale of datasets. For instance, GIGGLE [8] optimizes storage and querying of genomic intervals using B+ trees, and STIX [2] provides an interface to GIGGLE with greater resolution searches, statistical tests, and improved metadata support. Together, GIGGLE and STIX capture patterns of SVs across populations, and there is an opportunity to explore whether personalized representations from SVs are feasible for reducing computational burden and personalized medicine applications. Neural networks are powerful tools for learning embeddings, yet most off-the-shelf neural networks are designed for common learning tasks like classification or regression. Compared to classification and regression, representation learning with neural networks has niche, yet powerful applications. Siamese neural networks have been used for representation learning in signature verification [1], face verification [9], and other domains related to learning embedding reprsentations of concepts to capture some amount of concept (dis)similarity geometrically in the embedding space. In this work, a Siamese neural network will learn to project the genomic variation of individuals into an embedding space. However, learning a shared embedding space for genomic variation across both cancer and healthy populations proposes the challenge of detrivializing the model's optimization to separate the two populations without much consideration of within population genomic variation. For complex learning goals, curriculum learning is a technique which trains a model on different learning objectives, often sorted ascending by difficulty, to achieve better performance on complex tasks [6]. In this work, curriculum learning will enforce two curriculums for learning SV representations: 1) a within population curriculum and 2) a cross population curriculum.

4 Methods

4.1 Problem setup

The Siamese neural network's objective is to take as input the SV distribution of two individuals and learn low dimensional embedding representations via a symbolic curriculum learning approach.

4.2 Data

SVs are gathered from two different populations and studies: PCAWG (cancer) and 1000 genomes project (healthy). The SV evidence is directly gathered from the short-read sequences of each study. Briefly, short-reads are small segments of the sample genome aligned to a reference, and the resulting alignments are stored in a binary alignment map (BAMs) file. For each individual, SV evidence is computed by the excord program which reports SV evidence in the form of genomic intervals; excord results are stored in a generic genomic interval file format called browser extensible data (BED) files. Next, GIGGLE [8] builds B+ tree data structure to index the intervals from full corpus of BED files to support scalable searching of many genomic interval files given a query interval. This study focuses on gene fusion SVs which narrows the queries to only protein coding regions of the genome. The final input data to the model is sample-wise gene fusion SV count distributions.

4.3 Siamese network (distributed component)

The Siamese network consists of a single network which encodes the paired inputs via an architecture similar to a standard feed forward neural network. The final layer of the encoder is the embedding size.

4.4 Curriculum learning (symbolic component)

In curriculum one, the model performs representation learning of embeddings by minimizing the difference between the euclidean distance of the encoded embedding representations and the ground truth Kullback-Leibler divergence of the input SV distributions.

$$L = (||E_P - E_Q|| - D_{KL}(P||Q))^2$$

This initial learning is "local" since the input pairs are individuals from the same population.

Curriculum two requires the model to contrast between embeddings of different populations using triplet loss.

$$L(A, P, N) = max(||E_A - E_P||_2 - ||E_A - E_N||_2 + \alpha, 0)$$

where E_A, E_P, and E_N are the embeddings of an anchor, positive, and negative such that anchor and positive are sample from the population, while the E_N sample is from a different population.

Typically curriculum learning is performed in order of least difficult task to most difficult task. For this work however, curriculum one is expected to be much more challenging than curriculum two. Intuitively, the SV distributions across populations are substantially different which implies curriculum 2 is more trivial than curriculum 1. The motivation for training on the more difficult task first is to ensure population-wise embedding subspaces have non-trivial geometry; a trivial geometry may place all embeddings of a population arbitrarily distant from one another within a population, yet significantly distant from embeddings of other populations. The curriculum learning setup can be tuned by reversing the order or even cycling the learning objective periodically between iterations.

4.5 Parameter tuning

The number of hidden layers, activation functions, dropout, batch normalization, and importantly the size of the embedding layers are tuned based on empirical performance on the validation set. The tradeoff between efficiency and efficacy of embeddings is be determined by trying different embedding sizes in the final layer of the encoder. Different optimizers and learning rates is evaluated to balancing efficiency and efficacy of convergence.

5 Evaluation

5.1 Embedding quality (recall at top K)

The embedding quality is measured with respect to the original distance between SV distributions between individuals by recall at top K. For a single individual (query), the most similar SV distributions are ranked ascending by Kullback-Liebler divergence to the individual's distribution. This ground truth ranked list is compared to the ranked list of similar embeddings; meaning, a K-nearest neighbor (KNN) search is performed in the embedding space for the query individual.

5.2 Population precision (precision at top K)

Similar to embedding quality, the population precision is measured by comparing the ranked list of KL and KNN and counting the fraction of top K hits with the same population membership.

5.3 Computational efficiency (storage and retrieval)

Data storage improvements is evaluated by comparing the storage space required for the initial SV distributions to the storage space occupied by the embeddings representations. Data retrieval is measured by the difference in runtime for computing the KL divergence between all possible pairwise sample combinations compared to the runtime of pairwise KNN search in the embedding space.

5.4 Baselines

The gold standard for similarity measurements between SV distributions is the KL divergence. In addition, principal component analysis (PCA) and Johnson-Lindenstrauss transformation are other embedding techniques which can be compared against the Siamese neural network. PCA is a commonly used technique for extracting latent variables with a reasonable amount of explainability, yet the effectiveness of PCA depends on having some linear correlations in the input dataset. Meanwhile, Johnson-Lindenstrauss uses a random Gaussian matrix to transform samples while preserving pairwise distances well, yet the extent to which pairwise distances are conserved depends on the embedding size chosen and number of samples in the dataset [7].

6 Experimental

6.1 Data curation

The data curation steps are identical for the 1000 genomes and PCAWG studies.

- Download BAMs from the project's (1000 genomes or PCAWG) public databases.
- Compute SV evidence for each BAM with excord.
- Build a GIGGLE index for all excord BEDs in the population.
- Query the GIGGLE index for protein coding genes of GRCh37.
- Intersect the GIGGLE results back with the protein coding genes of GRCh37 to get a count distribution of gene fusion SVs.
- Normalize SV count distribution by taking the L2 norm of. Or, global normalization can be applied by min-max scaling each sample's SV distribution with respect to the population distribution of SVs.

6.2 Training data

The training data is used fully and tested fully to assess solely the computational benefits of SV representations, and a 60/20/20 split partitions the data into training, validation, and testing for assessing the generalizability of new data. Generalizability is more important for personalized medicine applications and dynamic databases while the full train-test approach better assesses a static database use case.

6.3 Model training

Pytorch, Keras, or Tensorflow will be used for the Siamese neural network architecture and training. Model training will be visualized using Weights and Biases (https://wandb.ai/site).

References

[1] Jane Bromley et al. “Signature verification using a "Siamese" time delay neural network”. In: *Proceedings of the 6th International Conference on Neural Information Processing Systems*. NIPS'93. Denver, Colorado: Morgan Kaufmann Publishers Inc., 1993, pp. 737–744.

[2] Murad Chowdhury et al. “Searching thousands of genomes to classify somatic and novel structural variants using STIX”. In: *Nature Methods* 19.4 (2022), pp. 445–448. ISSN: 1548-7105. DOI: 10.1038/s41592-022-01423-4. URL: https://doi.org/10.1038/s41592-022-01423-4.

[3] Marco Raffaele Cosenza, Bernardo Rodriguez-Martin, and Jan O. Korbel. “Structural Variation in Cancer: Role, Prevalence, and Mechanisms”. In: *Annual Review of Genomics and Human Genetics* 23 (Aug. 2022). Epub 2022 Jun 2, pp. 123–152. ISSN: 1545-293X. DOI: 10.1146/annurev-genom-120121-101149. URL: https://doi.org/10.1146/annurev-genom-120121-101149.

[4] Susan Fairley et al. “The International Genome Sample Resource (IGSR) collection of open human genomic variation resources”. In: *Nucleic Acids Research* 48.D1 (Oct. 2019), pp. D941–D947. ISSN: 0305-1048. DOI: 10.1093/nar/gkz836. eprint: https://academic.oup.com/nar/article-pdf/48/D1/D941/31697918/gkz836.pdf. URL: https://doi.org/10.1093/nar/gkz836.

[5] Mary J. Goldman et al. “A user guide for the online exploration and visualization of PCAWG data”. In: *Nature Communications* 11.1 (2020), p. 3400. ISSN: 2041-1723. DOI: 10.1038/s41467-020-16785-6. URL: https://doi.org/10.1038/s41467-020-16785-6.

[6] Guy Hacohen and Daphna Weinshall. “On The Power of Curriculum Learning in Training Deep Networks”. In: *Proceedings of the 36th International Conference on Machine Learning*. Ed. by Kamalika Chaudhuri and Ruslan Salakhutdinov. Vol. 97. Proceedings of Machine Learning Research. PMLR, Sept. 2019, pp. 2535–2544. URL: https://proceedings.mlr.press/v97/hacohen19a.html.

[7] William Johnson and J. Lindenstrauss. “Extensions of Lipschitz mappings into a Hilbert space”. In: *Conference in Modern Analysis and Probability* 26 (Jan. 1982), pp. 189–206.

[8] Ryan M. Layer et al. “GIGGLE: a search engine for large-scale integrated genome analysis”. In: *Nature Methods* 15.2 (2018), pp. 123–126. ISSN: 1548-7105. DOI: 10.1038/nmeth.4556. URL: https://doi.org/10.1038/nmeth.4556.

[9] Yaniv Taigman et al. “DeepFace: Closing the Gap to Human-Level Performance in Face Verification”. In: *2014 IEEE Conference on Computer Vision and Pattern Recognition*. 2014, pp. 1701–1708. DOI: 10.1109/CVPR.2014.220.