CLAIMS

What is claimed is:

1. A method of managing lifecycle of machine learning models, the method comprising:

receiving a set of unannotated data;

requesting annotations of samples of the unannotated data to produce an annotated set of data;

building a machine learning model based on the annotated set of data;

deploying the machine learning model to a client system, wherein production annotations are generated;

collecting the generated production annotations and generating a new machine learning model incorporating the production annotations; and

selecting one of the machine learning model built based on the annotated set of data or the new machine learning model;

wherein requesting annotations of samples comprises selecting a sample from the set of unannotated data based on an automated sampler selection, wherein the automated sampler selection is from one of a plurality of samplers in a progression and each of the plurality of samplers uses a different sampling algorithm; and wherein the progression comprises successively changing between samplers of the plurality of the samplers, and wherein each sampler of the plurality of samplers has an expected distribution of outcomes that determine whether to move to a previous or next sampler in the progression.

2. The method of claim 1, further comprising:

reporting one or more measures of quality of the machine learning model including precision, recall, average precision, receiver operator characteristic scores, or F-beta scores.

3. The method of claim 1, further comprising:

sharing the model with a third party.

4. The method of claim 1, wherein the sampling algorithm is selected from a density sampling algorithm; entropy sampling algorithm; estimated error reduction sampling algorithm; exhaustive sampling algorithm; flagged predictions algorithm; hard negative mining sampling algorithm; high confidence sampling algorithm; linear sampling algorithm; map visualization sampling algorithm; metadata search sampling algorithm; minimum margin sampling algorithm; query by committee sampling algorithm; random sampling algorithm; review sampling algorithm; search sampling algorithm; similarity sampling algorithm; sampling of samples for which the input was to skip the sample type algorithm; stratified sampling algorithm; most confident samples algorithm; or most uncertain samples algorithm.

5. The method of claim 1, wherein:

upon receiving a predetermined number of sample annotations with incorrect model predictions, the progression changes between samplers to a previous sampler in the progression; and

upon receiving a predetermined number of sample annotations with consistent model predictions, the progression changes between samplers to a next sampler in the progression.

6. The method of claim 1, wherein building the machine learning model comprises receiving a shared model and initializing weights of an intermediate model to weights of the shared model and trained with different learning rates.

7. The method of claim 1, wherein requesting annotations of samples of the unannotated data comprises requesting exhaustive annotations of a test set of data, and wherein the exhaustive annotations of the test set of data is performed by distant supervision comprising one or more of density sampling, level set trees, or random sampling.

8. The method of claim 1, wherein requesting annotations of samples of the unannotated data comprises presenting a recommendation on a graphical user interface of a sampler from a plurality of samplers for selecting a sample from the set of unannotated data; and

wherein the method further comprises presenting data quality and quantity metrics on the graphical user interface.

9. The method of claim 8, wherein the data quantity metrics comprise one or more of a number of samples trained, a number of positive examples, a number of negative examples, or a number of samples trained for a class of samples.

10. The method of claim 8, wherein the data quality metrics comprise one or more of an accuracy, precision, recall, or F1 score.

11. The method of claim 1, further comprising presenting, on a graphical user

interface, inconsistencies across annotations of the unannotated set of data.

12. The method of claim 1, wherein building the machine learning model comprises selecting an algorithm and loss function to establish the machine learning model.

13. The method of claim 12, wherein selecting the algorithm is based on a model type.

14. The method of claim 12, further comprising:

testing convergence by training a model multiple times on a set of annotated training data that is annotated from the unannotated set of data and measuring a dispersion of quality metrics across runs.

15. The method of claim 14, wherein the quality metrics include a slope of a learning curve.

16. The method of claim 12, wherein the model is trained using default hyperparameters selected for a given model type and the algorithm.

17. The method of claim 16, wherein the hyperparameters are selected using one or more of random selection, grid search, or Bayesian estimation methods.

18. The method of claim 16, wherein one or more of random seeds, algorithm

selection, loss function, hyperparameters, dataset splits, dataset hashes, or class weights are stored for the model.

19. The method of claim 1, where the machine learning model is versioned, changed over, or rolled back.

20. The method of claim 1, further comprising:

monitoring for changes between models via data drift or concept drift,

wherein concept drift is calculated by training models based on quantifying a number of changed predictions between the annotated set of data and the production annotations, and

wherein data drift is measured based on corpus statistics and/or corpus comparisons between the annotated set of data and the production annotations.

21. The method of claim 20, wherein an alert is generated upon identifying data drift or concept drift.

22. The method of claim 21, wherein the data drift or concept drift comprises metrics on unannotated data over time or metrics on model predictions over time.

23. The method of claim 3, wherein sharing the model comprises performing one or more of feature hashing, cryptographic hashing, or random projections.

24. The method of claim 3, wherein sharing the model comprises sharing a gradient

update of the model and the gradient update is added to a layer in a computational graph.

25. The method of claim 3, wherein sharing the model comprises sharing one or more model assets, and wherein the one or more model assets comprises word embeddings trained on datasets, word vectors, sets of annotations, lists of keywords and phrases, lists of examples, language models, lexicons, as well as trained models, and model architectures.

26. The method of claim 25, wherein the one or more model assets is sanitized of personally identifiable information.

27. The method of claim 1, wherein the progression comprises progressing from a seed sampler to a hard negative sampler to a stratified sampler, to an uncertainty sampler.

28. The method claim 1, wherein requesting annotations of samples comprises presenting questions to a user on a graphical user interface for annotation feedback.

29. The method of claim 1, further comprising predicting one or more annotations for a sample of the unannotated data.

30. The method of claim 29, wherein the predicting of the one or more annotations is prior to requesting annotations of samples of the unannotated data.

31. The method of claim 29, further comprising storing the predicted one or more

annotations in a priority queue based on a sampling score.

32. The method of claim 31, wherein the sampling score is a confidence score of the predicted one or more annotations.

33. The method of claim 31, further comprising:

prior to storing the predicted one or more annotations in the priority queue, determining whether the sampling score is greater than a threshold sampling score; and

discarding a prediction having a sampling score that is determined to be less than the threshold sampling score.

34. The method of claim 31, wherein the priority queue stores a predetermined maximum number of predictions, and wherein the method further comprises:

determining that a number of predictions stored in the priority queue is less than the predetermined maximum number of predictions prior to storing the prediction in the priority queue.

35. The method of claim 31, further comprising determining that the sampling score is greater than at least one previously stored prediction in the priority queue prior to storing the prediction in the priority queue.

36. The method of claim 31, further comprising discarding a previously stored prediction in the priority queue having a lowest sampling score.

37. The method of claim 31, wherein requesting annotations of samples of the unannotated data comprises selecting the priority queue from among a plurality of priority queues.

38. A system for managing lifecycle of machine learning models, the system comprises a non-transitory memory device coupled to a processor, the non-transitory memory device storing computer-readable instructions which, when executed by the processor, cause the system to perform the method according to any one of claims 1 to 37.