package com.twitter.follow\_recommendations.flows.post\_nux\_ml

import com.twitter.conversions.DurationOps.\_

import com.twitter.follow\_recommendations.common.models.CandidateUser

import com.twitter.follow\_recommendations.common.rankers.weighted\_candidate\_source\_ranker.CandidateShuffler

import com.twitter.follow\_recommendations.common.rankers.weighted\_candidate\_source\_ranker.ExponentialShuffler

import com.twitter.timelines.configapi.DurationConversion

import com.twitter.timelines.configapi.FSBoundedParam

import com.twitter.timelines.configapi.FSParam

import com.twitter.timelines.configapi.HasDurationConversion

import com.twitter.timelines.configapi.Param

import com.twitter.util.Duration

abstract class PostNuxMlParams[A](default: A) extends Param[A](default) {

override val statName: String = "post\_nux\_ml/" + this.getClass.getSimpleName

}

object PostNuxMlParams {

// infra params:

case object FetchCandidateSourceBudget extends PostNuxMlParams[Duration](90.millisecond)

// WTF Impression Store has very high tail latency (p9990 or p9999), but p99 latency is pretty good (~100ms)

// set the time budget for this step to be 200ms to make the performance of service more predictable

case object FatigueRankerBudget extends PostNuxMlParams[Duration](200.millisecond)

case object MlRankerBudget

extends FSBoundedParam[Duration](

name = PostNuxMlFlowFeatureSwitchKeys.MLRankerBudget,

default = 400.millisecond,

min = 100.millisecond,

max = 800.millisecond)

with HasDurationConversion {

override val durationConversion: DurationConversion = DurationConversion.FromMillis

}

// product params:

case object TargetEligibility extends PostNuxMlParams[Boolean](true)

case object ResultSizeParam extends PostNuxMlParams[Int](3)

case object BatchSizeParam extends PostNuxMlParams[Int](12)

case object CandidateShuffler

extends PostNuxMlParams[CandidateShuffler[CandidateUser]](

new ExponentialShuffler[CandidateUser])

case object LogRandomRankerId extends PostNuxMlParams[Boolean](false)

// whether or not to use the ml ranker at all (feature hydration + ranker)

case object UseMlRanker

extends FSParam[Boolean](PostNuxMlFlowFeatureSwitchKeys.UseMlRanker, false)

// whether or not to enable candidate param hydration in postnux\_ml\_flow

case object EnableCandidateParamHydration

extends FSParam[Boolean](PostNuxMlFlowFeatureSwitchKeys.EnableCandidateParamHydration, false)

// Whether or not OnlineSTP candidates are considered in the final pool of candidates.

// If set to `false`, the candidate source will be removed \*after\* all other considerations.

case object OnlineSTPEnabled

extends FSParam[Boolean](PostNuxMlFlowFeatureSwitchKeys.OnlineSTPEnabled, false)

// Whether or not the candidates are sampled from a Plackett-Luce model

case object SamplingTransformEnabled

extends FSParam[Boolean](PostNuxMlFlowFeatureSwitchKeys.SamplingTransformEnabled, false)

// Whether or not Follow2Vec candidates are considered in the final pool of candidates.

// If set to `false`, the candidate source will be removed \*after\* all other considerations.

case object Follow2VecLinearRegressionEnabled

extends FSParam[Boolean](

PostNuxMlFlowFeatureSwitchKeys.Follow2VecLinearRegressionEnabled,

false)

// Whether or not to enable AdhocRanker to allow adhoc, non-ML, score modifications.

case object EnableAdhocRanker

extends FSParam[Boolean](PostNuxMlFlowFeatureSwitchKeys.EnableAdhocRanker, false)

// Whether the impression-based fatigue ranker is enabled or not.

case object EnableFatigueRanker

extends FSParam[Boolean](PostNuxMlFlowFeatureSwitchKeys.EnableFatigueRanker, true)

// whether or not to enable InterleaveRanker for producer-side experiments.

case object EnableInterleaveRanker

extends FSParam[Boolean](PostNuxMlFlowFeatureSwitchKeys.EnableInterleaveRanker, false)

// whether to exclude users in near zero user state

case object ExcludeNearZeroCandidates

extends FSParam[Boolean](PostNuxMlFlowFeatureSwitchKeys.ExcludeNearZeroCandidates, false)

case object EnablePPMILocaleFollowSourceInPostNux

extends FSParam[Boolean](

PostNuxMlFlowFeatureSwitchKeys.EnablePPMILocaleFollowSourceInPostNux,

false)

case object EnableInterestsOptOutPredicate

extends FSParam[Boolean](PostNuxMlFlowFeatureSwitchKeys.EnableInterestsOptOutPredicate, false)

case object EnableInvalidRelationshipPredicate

extends FSParam[Boolean](

PostNuxMlFlowFeatureSwitchKeys.EnableInvalidRelationshipPredicate,

false)

// Totally disabling SGS predicate need to disable EnableInvalidRelationshipPredicate as well

case object EnableSGSPredicate

extends FSParam[Boolean](PostNuxMlFlowFeatureSwitchKeys.EnableSGSPredicate, true)

case object EnableHssPredicate

extends FSParam[Boolean](PostNuxMlFlowFeatureSwitchKeys.EnableHssPredicate, true)

// Whether or not to include RepeatedProfileVisits as one of the candidate sources in the PostNuxMlFlow. If false,

// RepeatedProfileVisitsSource would not be run for the users in candidate\_generation.

case object IncludeRepeatedProfileVisitsCandidateSource

extends FSParam[Boolean](

PostNuxMlFlowFeatureSwitchKeys.IncludeRepeatedProfileVisitsCandidateSource,

false)

case object EnableRealGraphOonV2

extends FSParam[Boolean](PostNuxMlFlowFeatureSwitchKeys.EnableRealGraphOonV2, false)

case object GetFollowersFromSgs

extends FSParam[Boolean](PostNuxMlFlowFeatureSwitchKeys.GetFollowersFromSgs, false)

case object EnableRemoveAccountProofTransform

extends FSParam[Boolean](

PostNuxMlFlowFeatureSwitchKeys.EnableRemoveAccountProofTransform,

false)

// quality factor threshold to turn off ML ranker completely

object TurnoffMLScorerQFThreshold

extends FSBoundedParam[Double](

name = PostNuxMlFlowFeatureSwitchKeys.TurnOffMLScorerQFThreshold,

default = 0.3,

min = 0.1,

max = 1.0)

}