package com.twitter.cr\_mixer.param

import com.twitter.conversions.DurationOps.\_

import com.twitter.cr\_mixer.model.EarlybirdSimilarityEngineType

import com.twitter.cr\_mixer.model.EarlybirdSimilarityEngineType\_ModelBased

import com.twitter.cr\_mixer.model.EarlybirdSimilarityEngineType\_RecencyBased

import com.twitter.cr\_mixer.model.EarlybirdSimilarityEngineType\_TensorflowBased

import com.twitter.finagle.stats.NullStatsReceiver

import com.twitter.logging.Logger

import com.twitter.timelines.configapi.BaseConfig

import com.twitter.timelines.configapi.BaseConfigBuilder

import com.twitter.timelines.configapi.DurationConversion

import com.twitter.timelines.configapi.FSBoundedParam

import com.twitter.timelines.configapi.FSEnumParam

import com.twitter.timelines.configapi.FSName

import com.twitter.timelines.configapi.FSParam

import com.twitter.timelines.configapi.FeatureSwitchOverrideUtil

import com.twitter.timelines.configapi.HasDurationConversion

import com.twitter.timelines.configapi.Param

import com.twitter.util.Duration

object EarlybirdFrsBasedCandidateGenerationParams {

object CandidateGenerationEarlybirdSimilarityEngineType extends Enumeration {

protected case class SimilarityEngineType(rankingMode: EarlybirdSimilarityEngineType)

extends super.Val

import scala.language.implicitConversions

implicit def valueToEarlybirdRankingMode(x: Value): SimilarityEngineType =

x.asInstanceOf[SimilarityEngineType]

val EarlybirdRankingMode\_RecencyBased: SimilarityEngineType = SimilarityEngineType(

EarlybirdSimilarityEngineType\_RecencyBased)

val EarlybirdRankingMode\_ModelBased: SimilarityEngineType = SimilarityEngineType(

EarlybirdSimilarityEngineType\_ModelBased)

val EarlybirdRankingMode\_TensorflowBased: SimilarityEngineType = SimilarityEngineType(

EarlybirdSimilarityEngineType\_TensorflowBased)

}

object FrsBasedCandidateGenerationEarlybirdSimilarityEngineTypeParam

extends FSEnumParam[CandidateGenerationEarlybirdSimilarityEngineType.type](

name = "frs\_based\_candidate\_generation\_earlybird\_ranking\_mode\_id",

default =

CandidateGenerationEarlybirdSimilarityEngineType.EarlybirdRankingMode\_RecencyBased,

enum = CandidateGenerationEarlybirdSimilarityEngineType

)

object FrsBasedCandidateGenerationRecencyBasedEarlybirdMaxTweetsPerUser

extends FSBoundedParam[Int](

name = "frs\_based\_candidate\_generation\_earlybird\_max\_tweets\_per\_user",

default = 100,

min = 0,

/\*\*

\* Note max should be equal to EarlybirdRecencyBasedCandidateStoreModule.DefaultMaxNumTweetPerUser.

\* Which is the size of the memcached result list.

\*/

max = 100

)

object FrsBasedCandidateGenerationEarlybirdMaxTweetAge

extends FSBoundedParam[Duration](

name = "frs\_based\_candidate\_generation\_earlybird\_max\_tweet\_age\_hours",

default = 24.hours,

min = 12.hours,

/\*\*

\* Note max could be related to EarlybirdRecencyBasedCandidateStoreModule.DefaultMaxNumTweetPerUser.

\* Which is the size of the memcached result list for recency based earlybird candidate source.

\* E.g. if max = 720.hours, we may want to increase the DefaultMaxNumTweetPerUser.

\*/

max = 96.hours

)

with HasDurationConversion {

override val durationConversion: DurationConversion = DurationConversion.FromHours

}

object FrsBasedCandidateGenerationEarlybirdFilterOutRetweetsAndReplies

extends FSParam[Boolean](

name = "frs\_based\_candidate\_generation\_earlybird\_filter\_out\_retweets\_and\_replies",

default = true

)

val AllParams: Seq[Param[\_] with FSName] = Seq(

FrsBasedCandidateGenerationEarlybirdSimilarityEngineTypeParam,

FrsBasedCandidateGenerationRecencyBasedEarlybirdMaxTweetsPerUser,

FrsBasedCandidateGenerationEarlybirdMaxTweetAge,

FrsBasedCandidateGenerationEarlybirdFilterOutRetweetsAndReplies,

)

lazy val config: BaseConfig = {

val booleanOverrides = FeatureSwitchOverrideUtil.getBooleanFSOverrides(

FrsBasedCandidateGenerationEarlybirdFilterOutRetweetsAndReplies,

)

val doubleOverrides = FeatureSwitchOverrideUtil.getBoundedDoubleFSOverrides()

val intOverrides = FeatureSwitchOverrideUtil.getBoundedIntFSOverrides(

FrsBasedCandidateGenerationRecencyBasedEarlybirdMaxTweetsPerUser

)

val durationFSOverrides =

FeatureSwitchOverrideUtil.getDurationFSOverrides(

FrsBasedCandidateGenerationEarlybirdMaxTweetAge

)

val enumOverrides = FeatureSwitchOverrideUtil.getEnumFSOverrides(

NullStatsReceiver,

Logger(getClass),

FrsBasedCandidateGenerationEarlybirdSimilarityEngineTypeParam,

)

BaseConfigBuilder()

.set(booleanOverrides: \_\*)

.set(doubleOverrides: \_\*)

.set(intOverrides: \_\*)

.set(enumOverrides: \_\*)

.set(durationFSOverrides: \_\*)

.build()

}

}