package com.twitter.follow\_recommendations.common.rankers.ml\_ranker.scoring

import com.twitter.finagle.stats.StatsReceiver

import com.twitter.follow\_recommendations.common.rankers.common.RankerId

import com.twitter.follow\_recommendations.common.rankers.common.RankerId.RankerId

import javax.inject.Inject

import javax.inject.Singleton

@Singleton

class ScorerFactory @Inject() (

postnuxProdScorer: PostnuxDeepbirdProdScorer,

randomScorer: RandomScorer,

stats: StatsReceiver) {

private val scorerFactoryStats = stats.scope("scorer\_factory")

private val scorerStat = scorerFactoryStats.scope("scorer")

def getScorers(

rankerIds: Seq[RankerId]

): Seq[Scorer] = {

rankerIds.map { scorerId =>

val scorer: Scorer = getScorerById(scorerId)

// count # of times a ranker has been requested

scorerStat.counter(scorer.id.toString).incr()

scorer

}

}

def getScorerById(scorerId: RankerId): Scorer = scorerId match {

case RankerId.PostNuxProdRanker =>

postnuxProdScorer

case RankerId.RandomRanker =>

randomScorer

case \_ =>

scorerStat.counter("invalid\_scorer\_type").incr()

throw new IllegalArgumentException("unknown\_scorer\_type")

}

}