package com.twitter.cr\_mixer.module.similarity\_engine

import com.google.inject.Provides

import com.twitter.cr\_mixer.config.TimeoutConfig

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

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

import com.twitter.cr\_mixer.param.decider.CrMixerDecider

import com.twitter.cr\_mixer.param.decider.DeciderConstants

import com.twitter.cr\_mixer.similarity\_engine.ConsumersBasedUserVideoGraphSimilarityEngine

import com.twitter.cr\_mixer.similarity\_engine.SimilarityEngine.DeciderConfig

import com.twitter.cr\_mixer.similarity\_engine.SimilarityEngine.GatingConfig

import com.twitter.cr\_mixer.similarity\_engine.SimilarityEngine.SimilarityEngineConfig

import com.twitter.cr\_mixer.similarity\_engine.StandardSimilarityEngine

import com.twitter.cr\_mixer.thriftscala.SimilarityEngineType

import com.twitter.finagle.stats.StatsReceiver

import com.twitter.inject.TwitterModule

import com.twitter.recos.user\_video\_graph.thriftscala.ConsumersBasedRelatedTweetRequest

import com.twitter.recos.user\_video\_graph.thriftscala.RelatedTweetResponse

import com.twitter.storehaus.ReadableStore

import javax.inject.Named

import javax.inject.Singleton

object ConsumersBasedUserVideoGraphSimilarityEngineModule extends TwitterModule {

@Provides

@Singleton

@Named(ModuleNames.ConsumersBasedUserVideoGraphSimilarityEngine)

def providesConsumersBasedUserVideoGraphSimilarityEngine(

@Named(ModuleNames.ConsumerBasedUserVideoGraphStore)

consumersBasedUserVideoGraphStore: ReadableStore[

ConsumersBasedRelatedTweetRequest,

RelatedTweetResponse

],

timeoutConfig: TimeoutConfig,

statsReceiver: StatsReceiver,

decider: CrMixerDecider

): StandardSimilarityEngine[

ConsumersBasedUserVideoGraphSimilarityEngine.Query,

TweetWithScore

] = {

new StandardSimilarityEngine[

ConsumersBasedUserVideoGraphSimilarityEngine.Query,

TweetWithScore

](

implementingStore = ConsumersBasedUserVideoGraphSimilarityEngine(

consumersBasedUserVideoGraphStore,

statsReceiver),

identifier = SimilarityEngineType.ConsumersBasedUserVideoGraph,

globalStats = statsReceiver,

engineConfig = SimilarityEngineConfig(

timeout = timeoutConfig.similarityEngineTimeout,

gatingConfig = GatingConfig(

deciderConfig =

Some(DeciderConfig(decider, DeciderConstants.enableUserVideoGraphTrafficDeciderKey)),

enableFeatureSwitch = None

)

),

memCacheConfig = None

)

}

}