package com.twitter.home\_mixer.product.scored\_tweets.feature\_hydrator.adapters.twhin\_embeddings

import com.twitter.ml.api.DataType

import com.twitter.ml.api.Feature

import com.twitter.ml.api.FeatureContext

import com.twitter.ml.api.RichDataRecord

import com.twitter.ml.api.util.ScalaToJavaDataRecordConversions

import com.twitter.ml.api.{thriftscala => ml}

import com.twitter.timelines.prediction.common.adapters.TimelinesMutatingAdapterBase

sealed trait TwhinEmbeddingsAdapter extends TimelinesMutatingAdapterBase[Option[ml.FloatTensor]] {

def twhinEmbeddingsFeature: Feature.Tensor

override def getFeatureContext: FeatureContext = new FeatureContext(

twhinEmbeddingsFeature

)

override def setFeatures(

embedding: Option[ml.FloatTensor],

richDataRecord: RichDataRecord

): Unit = {

embedding.foreach { floatTensor =>

richDataRecord.setFeatureValue(

twhinEmbeddingsFeature,

ScalaToJavaDataRecordConversions.scalaTensor2Java(

ml.GeneralTensor

.FloatTensor(floatTensor)))

}

}

}

object TwhinEmbeddingsFeatures {

val TwhinAuthorFollowEmbeddingsFeature: Feature.Tensor = new Feature.Tensor(

"original\_author.twhin.tw\_hi\_n.author\_follow\_as\_float\_tensor",

DataType.FLOAT

)

val TwhinUserEngagementEmbeddingsFeature: Feature.Tensor = new Feature.Tensor(

"user.twhin.tw\_hi\_n.user\_engagement\_as\_float\_tensor",

DataType.FLOAT

)

val TwhinUserFollowEmbeddingsFeature: Feature.Tensor = new Feature.Tensor(

"user.twhin.tw\_hi\_n.user\_follow\_as\_float\_tensor",

DataType.FLOAT

)

}

object TwhinAuthorFollowEmbeddingsAdapter extends TwhinEmbeddingsAdapter {

override val twhinEmbeddingsFeature: Feature.Tensor =

TwhinEmbeddingsFeatures.TwhinAuthorFollowEmbeddingsFeature

override val commonFeatures: Set[Feature[\_]] = Set.empty

}

object TwhinUserEngagementEmbeddingsAdapter extends TwhinEmbeddingsAdapter {

override val twhinEmbeddingsFeature: Feature.Tensor =

TwhinEmbeddingsFeatures.TwhinUserEngagementEmbeddingsFeature

override val commonFeatures: Set[Feature[\_]] = Set(twhinEmbeddingsFeature)

}

object TwhinUserFollowEmbeddingsAdapter extends TwhinEmbeddingsAdapter {

override val twhinEmbeddingsFeature: Feature.Tensor =

TwhinEmbeddingsFeatures.TwhinUserFollowEmbeddingsFeature

override val commonFeatures: Set[Feature[\_]] = Set(twhinEmbeddingsFeature)

}