package com.twitter.search.earlybird.search.relevance.scoring;

import com.twitter.search.common.util.ml.prediction\_engine.BaseLegacyScoreAccumulator;

import com.twitter.search.common.util.ml.prediction\_engine.LightweightLinearModel;

import com.twitter.search.earlybird.search.relevance.LinearScoringData;

import com.twitter.search.modeling.tweet\_ranking.TweetScoringFeatures;

/\*\*

\* Legacy score accumulator in Earlybird with specific features added.

\* This class is created to avoid adding LinearScoringData as a dependency to search's common ML

\* library.

\*

\* @deprecated This class is retired and we suggest to switch to SchemaBasedScoreAccumulator.

\*/

@Deprecated

public class LegacyScoreAccumulator extends BaseLegacyScoreAccumulator<LinearScoringData> {

/\*\*

\* Constructs with a model and LinearScoringData

\*/

LegacyScoreAccumulator(LightweightLinearModel model) {

super(model);

}

/\*\*

\* Update the accumulator score with features, after this function the score should already

\* be computed.

\*

\* @deprecated This function is retired and we suggest to switch to updateScoresWithFeatures in

\* SchemaBasedScoreAccumulator.

\*/

@Override

@Deprecated

protected void updateScoreWithFeatures(LinearScoringData data) {

addContinuousFeature(TweetScoringFeatures.LUCENE\_SCORE, data.luceneScore);

addContinuousFeature(TweetScoringFeatures.TEXT\_SCORE, data.textScore);

addContinuousFeature(TweetScoringFeatures.TWEET\_AGE\_IN\_SECONDS, data.tweetAgeInSeconds);

addContinuousFeature(TweetScoringFeatures.REPLY\_COUNT, data.replyCountPostLog2);

addContinuousFeature(TweetScoringFeatures.RETWEET\_COUNT, data.retweetCountPostLog2);

addContinuousFeature(TweetScoringFeatures.FAV\_COUNT, data.favCountPostLog2);

addContinuousFeature(TweetScoringFeatures.REPLY\_COUNT\_V2, data.replyCountV2);

addContinuousFeature(TweetScoringFeatures.RETWEET\_COUNT\_V2, data.retweetCountV2);

addContinuousFeature(TweetScoringFeatures.FAV\_COUNT\_V2, data.favCountV2);

addContinuousFeature(TweetScoringFeatures.EMBEDS\_IMPRESSION\_COUNT,

data.getEmbedsImpressionCount(false));

addContinuousFeature(TweetScoringFeatures.EMBEDS\_URL\_COUNT, data.getEmbedsUrlCount(false));

addContinuousFeature(TweetScoringFeatures.VIDEO\_VIEW\_COUNT, data.getVideoViewCount(false));

addContinuousFeature(TweetScoringFeatures.QUOTED\_COUNT, data.quotedCount);

addContinuousFeature(TweetScoringFeatures.WEIGHTED\_RETWEET\_COUNT, data.weightedRetweetCount);

addContinuousFeature(TweetScoringFeatures.WEIGHTED\_REPLY\_COUNT, data.weightedReplyCount);

addContinuousFeature(TweetScoringFeatures.WEIGHTED\_FAV\_COUNT, data.weightedFavCount);

addContinuousFeature(TweetScoringFeatures.WEIGHTED\_QUOTE\_COUNT, data.weightedQuoteCount);

addBinaryFeature(TweetScoringFeatures.HAS\_URL, data.hasUrl);

addBinaryFeature(TweetScoringFeatures.HAS\_CARD, data.hasCard);

addBinaryFeature(TweetScoringFeatures.HAS\_VINE, data.hasVine);

addBinaryFeature(TweetScoringFeatures.HAS\_PERISCOPE, data.hasPeriscope);

addBinaryFeature(TweetScoringFeatures.HAS\_NATIVE\_IMAGE, data.hasNativeImage);

addBinaryFeature(TweetScoringFeatures.HAS\_IMAGE\_URL, data.hasImageUrl);

addBinaryFeature(TweetScoringFeatures.HAS\_NEWS\_URL, data.hasNewsUrl);

addBinaryFeature(TweetScoringFeatures.HAS\_VIDEO\_URL, data.hasVideoUrl);

addBinaryFeature(TweetScoringFeatures.HAS\_CONSUMER\_VIDEO, data.hasConsumerVideo);

addBinaryFeature(TweetScoringFeatures.HAS\_PRO\_VIDEO, data.hasProVideo);

addBinaryFeature(TweetScoringFeatures.HAS\_QUOTE, data.hasQuote);

addBinaryFeature(TweetScoringFeatures.HAS\_TREND, data.hasTrend);

addBinaryFeature(TweetScoringFeatures.HAS\_MULTIPLE\_HASHTAGS\_OR\_TRENDS,

data.hasMultipleHashtagsOrTrends);

addBinaryFeature(TweetScoringFeatures.IS\_OFFENSIVE, data.isOffensive);

addBinaryFeature(TweetScoringFeatures.IS\_REPLY, data.isReply);

addBinaryFeature(TweetScoringFeatures.IS\_RETWEET, data.isRetweet);

addBinaryFeature(TweetScoringFeatures.IS\_SELF\_TWEET, data.isSelfTweet);

addBinaryFeature(TweetScoringFeatures.IS\_FOLLOW\_RETWEET, data.isRetweet & data.isFollow);

addBinaryFeature(TweetScoringFeatures.IS\_TRUSTED\_RETWEET, data.isRetweet & data.isTrusted);

addContinuousFeature(TweetScoringFeatures.QUERY\_SPECIFIC\_SCORE, data.querySpecificScore);

addContinuousFeature(TweetScoringFeatures.AUTHOR\_SPECIFIC\_SCORE, data.authorSpecificScore);

addBinaryFeature(TweetScoringFeatures.AUTHOR\_IS\_FOLLOW, data.isFollow);

addBinaryFeature(TweetScoringFeatures.AUTHOR\_IS\_TRUSTED, data.isTrusted);

addBinaryFeature(TweetScoringFeatures.AUTHOR\_IS\_VERIFIED, data.isFromVerifiedAccount);

addBinaryFeature(TweetScoringFeatures.AUTHOR\_IS\_NSFW, data.isUserNSFW);

addBinaryFeature(TweetScoringFeatures.AUTHOR\_IS\_SPAM, data.isUserSpam);

addBinaryFeature(TweetScoringFeatures.AUTHOR\_IS\_BOT, data.isUserBot);

addBinaryFeature(TweetScoringFeatures.AUTHOR\_IS\_ANTISOCIAL, data.isUserAntiSocial);

addContinuousFeature(TweetScoringFeatures.AUTHOR\_REPUTATION, data.userRep);

addContinuousFeature(TweetScoringFeatures.SEARCHER\_LANG\_SCORE, data.userLangMult);

addBinaryFeature(TweetScoringFeatures.HAS\_DIFFERENT\_LANG, data.hasDifferentLang);

addBinaryFeature(TweetScoringFeatures.HAS\_ENGLISH\_TWEET\_AND\_DIFFERENT\_UI\_LANG,

data.hasEnglishTweetAndDifferentUILang);

addBinaryFeature(TweetScoringFeatures.HAS\_ENGLISH\_UI\_AND\_DIFFERENT\_TWEET\_LANG,

data.hasEnglishUIAndDifferentTweetLang);

addBinaryFeature(TweetScoringFeatures.IS\_SENSITIVE\_CONTENT, data.isSensitiveContent);

addBinaryFeature(TweetScoringFeatures.HAS\_MULTIPLE\_MEDIA, data.hasMultipleMediaFlag);

addBinaryFeature(TweetScoringFeatures.AUTHOR\_IS\_PROFILE\_EGG, data.profileIsEggFlag);

addBinaryFeature(TweetScoringFeatures.AUTHOR\_IS\_NEW, data.isUserNewFlag);

addContinuousFeature(TweetScoringFeatures.MENTIONS\_COUNT, data.numMentions);

addContinuousFeature(TweetScoringFeatures.HASHTAGS\_COUNT, data.numHashtags);

addContinuousFeature(TweetScoringFeatures.LINK\_LANGUAGE\_ID, data.linkLanguage);

addContinuousFeature(TweetScoringFeatures.LANGUAGE\_ID, data.tweetLangId);

addBinaryFeature(TweetScoringFeatures.HAS\_VISIBLE\_LINK, data.hasVisibleLink);

}

}