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

import java.io.IOException;

import com.google.common.annotations.VisibleForTesting;

import org.apache.lucene.search.Explanation;

import com.twitter.search.common.relevance.features.RelevanceSignalConstants;

import com.twitter.search.common.schema.base.ImmutableSchemaInterface;

import com.twitter.search.common.schema.earlybird.EarlybirdFieldConstants.EarlybirdFieldConstant;

import com.twitter.search.earlybird.common.config.EarlybirdConfig;

import com.twitter.search.earlybird.thrift.ThriftSearchResultMetadata;

import com.twitter.search.earlybird.thrift.ThriftSearchResultMetadataOptions;

import com.twitter.search.earlybird.thrift.ThriftSearchResultsRelevanceStats;

public class SpamVectorScoringFunction extends ScoringFunction {

private static final int MIN\_TWEEPCRED\_WITH\_LINK =

EarlybirdConfig.getInt("min\_tweepcred\_with\_non\_whitelisted\_link", 25);

// The engagement threshold that prevents us from filtering users with low tweepcred.

private static final int ENGAGEMENTS\_NO\_FILTER = 1;

@VisibleForTesting

static final float NOT\_SPAM\_SCORE = 0.5f;

@VisibleForTesting

static final float SPAM\_SCORE = -0.5f;

public SpamVectorScoringFunction(ImmutableSchemaInterface schema) {

super(schema);

}

@Override

protected float score(float luceneQueryScore) throws IOException {

if (documentFeatures.isFlagSet(EarlybirdFieldConstant.FROM\_VERIFIED\_ACCOUNT\_FLAG)) {

return NOT\_SPAM\_SCORE;

}

int tweepCredThreshold = 0;

if (documentFeatures.isFlagSet(EarlybirdFieldConstant.HAS\_LINK\_FLAG)

&& !documentFeatures.isFlagSet(EarlybirdFieldConstant.HAS\_IMAGE\_URL\_FLAG)

&& !documentFeatures.isFlagSet(EarlybirdFieldConstant.HAS\_VIDEO\_URL\_FLAG)

&& !documentFeatures.isFlagSet(EarlybirdFieldConstant.HAS\_NEWS\_URL\_FLAG)) {

// Contains a non-media non-news link, definite spam vector.

tweepCredThreshold = MIN\_TWEEPCRED\_WITH\_LINK;

}

int tweepcred = (int) documentFeatures.getFeatureValue(EarlybirdFieldConstant.USER\_REPUTATION);

// For new user, tweepcred is set to a sentinel value of -128, specified at

// src/thrift/com/twitter/search/common/indexing/status.thrift

if (tweepcred >= tweepCredThreshold

|| tweepcred == (int) RelevanceSignalConstants.UNSET\_REPUTATION\_SENTINEL) {

return NOT\_SPAM\_SCORE;

}

double retweetCount =

documentFeatures.getUnnormalizedFeatureValue(EarlybirdFieldConstant.RETWEET\_COUNT);

double replyCount =

documentFeatures.getUnnormalizedFeatureValue(EarlybirdFieldConstant.REPLY\_COUNT);

double favoriteCount =

documentFeatures.getUnnormalizedFeatureValue(EarlybirdFieldConstant.FAVORITE\_COUNT);

// If the tweet has enough engagements, do not mark it as spam.

if (retweetCount + replyCount + favoriteCount >= ENGAGEMENTS\_NO\_FILTER) {

return NOT\_SPAM\_SCORE;

}

return SPAM\_SCORE;

}

@Override

protected Explanation doExplain(float luceneScore) {

return null;

}

@Override

public ThriftSearchResultMetadata getResultMetadata(ThriftSearchResultMetadataOptions options) {

return null;

}

@Override

public void updateRelevanceStats(ThriftSearchResultsRelevanceStats relevanceStats) {

}

}