package com.twitter.frigate.pushservice.take.predicates

import com.twitter.finagle.stats.StatsReceiver

import com.twitter.frigate.pushservice.config.Config

import com.twitter.frigate.pushservice.predicate.BqmlHealthModelPredicates

import com.twitter.frigate.pushservice.predicate.BqmlQualityModelPredicates

import com.twitter.frigate.pushservice.predicate.HealthPredicates

import com.twitter.frigate.pushservice.predicate.OONSpreadControlPredicate

import com.twitter.frigate.pushservice.predicate.OONTweetNegativeFeedbackBasedPredicate

import com.twitter.frigate.pushservice.predicate.OutOfNetworkCandidatesQualityPredicates

import com.twitter.frigate.pushservice.predicate.PredicatesForCandidate

import com.twitter.frigate.pushservice.predicate.PNegMultimodalPredicates

import com.twitter.frigate.pushservice.predicate.TargetEngagementPredicate

import com.twitter.frigate.pushservice.predicate.TweetEngagementRatioPredicate

import com.twitter.frigate.pushservice.predicate.TweetLanguagePredicate

import com.twitter.frigate.pushservice.predicate.TweetWithheldContentPredicate

trait BasicTweetPredicates {

def config: Config

implicit def statsReceiver: StatsReceiver

final lazy val basicTweetPredicates =

List(

HealthPredicates.sensitiveMediaCategoryPredicate(),

HealthPredicates.profanityPredicate(),

PredicatesForCandidate.disableOutNetworkTweetPredicate(config.edgeStore),

TweetEngagementRatioPredicate.QTtoNtabClickBasedPredicate(),

TweetLanguagePredicate.oonTweeetLanguageMatch(),

HealthPredicates.userHealthSignalsPredicate(config.userHealthSignalStore),

HealthPredicates.authorSensitiveMediaPredicate(config.producerMediaRepresentationStore),

HealthPredicates.authorProfileBasedPredicate(),

PNegMultimodalPredicates.healthSignalScorePNegMultimodalPredicate(

config.tweetHealthScoreStore),

BqmlHealthModelPredicates.healthModelOonPredicate(

config.filteringModelScorer,

config.producerMediaRepresentationStore,

config.userHealthSignalStore,

config.tweetHealthScoreStore),

BqmlQualityModelPredicates.BqmlQualityModelOonPredicate(config.filteringModelScorer),

HealthPredicates.tweetHealthSignalScorePredicate(config.tweetHealthScoreStore),

HealthPredicates

.tweetHealthSignalScorePredicate(config.tweetHealthScoreStore, applyToQuoteTweet = true),

PredicatesForCandidate.nullCastF1ProtectedExperientPredicate(

config.cachedTweetyPieStoreV2

),

OONTweetNegativeFeedbackBasedPredicate.ntabDislikeBasedPredicate(),

OONSpreadControlPredicate.oonTweetSpreadControlPredicate(),

OONSpreadControlPredicate.oonAuthorSpreadControlPredicate(),

HealthPredicates.healthSignalScoreMultilingualPnsfwTweetTextPredicate(

config.tweetHealthScoreStore),

PredicatesForCandidate

.recommendedTweetAuthorAcceptableToTargetUser(config.edgeStore),

HealthPredicates.healthSignalScorePnsfwTweetTextPredicate(config.tweetHealthScoreStore),

HealthPredicates.healthSignalScoreSpammyTweetPredicate(config.tweetHealthScoreStore),

OutOfNetworkCandidatesQualityPredicates.NegativeKeywordsPredicate(

config.postRankingFeatureStoreClient),

PredicatesForCandidate.authorNotBeingDeviceFollowed(config.edgeStore),

TweetWithheldContentPredicate(),

PredicatesForCandidate.noOptoutFreeFormInterestPredicate,

PredicatesForCandidate.disableInNetworkTweetPredicate(config.edgeStore),

TweetEngagementRatioPredicate.TweetReplyLikeRatioPredicate(),

TargetEngagementPredicate(

config.userTweetPerspectiveStore,

defaultForMissing = true

),

)

}

/\*\*

\* This trait is a new version of BasicTweetPredicates

\* Difference from old version is that basicTweetPredicates are different

\* basicTweetPredicates here don't include Social Graph Service related predicates

\*/

trait BasicTweetPredicatesWithoutSGSPredicates {

def config: Config

implicit def statsReceiver: StatsReceiver

final lazy val basicTweetPredicates = {

List(

HealthPredicates.healthSignalScoreSpammyTweetPredicate(config.tweetHealthScoreStore),

PredicatesForCandidate.nullCastF1ProtectedExperientPredicate(

config.cachedTweetyPieStoreV2

),

TweetWithheldContentPredicate(),

TargetEngagementPredicate(

config.userTweetPerspectiveStore,

defaultForMissing = true

),

PredicatesForCandidate.noOptoutFreeFormInterestPredicate,

HealthPredicates.userHealthSignalsPredicate(config.userHealthSignalStore),

HealthPredicates.tweetHealthSignalScorePredicate(config.tweetHealthScoreStore),

BqmlQualityModelPredicates.BqmlQualityModelOonPredicate(config.filteringModelScorer),

BqmlHealthModelPredicates.healthModelOonPredicate(

config.filteringModelScorer,

config.producerMediaRepresentationStore,

config.userHealthSignalStore,

config.tweetHealthScoreStore),

)

}

}