package com.twitter.frigate.pushservice.params

import com.twitter.conversions.DurationOps.\_

import com.twitter.frigate.pushservice.params.InlineActionsEnum.\_

import com.twitter.frigate.pushservice.params.HighQualityCandidateGroupEnum.\_

import com.twitter.timelines.configapi.DurationConversion

import com.twitter.timelines.configapi.FSBoundedParam

import com.twitter.timelines.configapi.FSEnumParam

import com.twitter.timelines.configapi.FSEnumSeqParam

import com.twitter.timelines.configapi.FSParam

import com.twitter.timelines.configapi.HasDurationConversion

import com.twitter.util.Duration

object PushFeatureSwitchParams {

/\*\*

\* List of CRTs to uprank. Last CRT in sequence ends up on top of list

\*/

object ListOfCrtsToUpRank

extends FSParam[Seq[String]]("rerank\_candidates\_crt\_to\_top", default = Seq.empty[String])

object ListOfCrtsForOpenApp

extends FSParam[Seq[String]](

"open\_app\_allowed\_crts",

default = Seq(

"f1firstdegreetweet",

"f1firstdegreephoto",

"f1firstdegreevideo",

"geopoptweet",

"frstweet",

"trendtweet",

"hermituser",

"triangularloopuser"

))

/\*\*

\* List of CRTs to downrank. Last CRT in sequence ends up on bottom of list

\*/

object ListOfCrtsToDownRank

extends FSParam[Seq[String]](

name = "rerank\_candidates\_crt\_to\_downrank",

default = Seq.empty[String])

/\*\*

\* Param to enable VF filtering in Tweetypie (vs using VisibilityLibrary)

\*/

object EnableVFInTweetypie

extends FSParam[Boolean](

name = "visibility\_filtering\_enable\_vf\_in\_tweetypie",

default = true

)

/\*\*

\* Number of max earlybird candidates

\*/

object NumberOfMaxEarlybirdInNetworkCandidatesParam

extends FSBoundedParam(

name = "frigate\_push\_max\_earlybird\_in\_network\_candidates",

default = 100,

min = 0,

max = 800

)

/\*\*

\* Number of max UserTweetEntityGraph candidates to query

\*/

object NumberOfMaxUTEGCandidatesQueriedParam

extends FSBoundedParam(

name = "frigate\_push\_max\_uteg\_candidates\_queried",

default = 30,

min = 0,

max = 300

)

/\*\*

\* Param to control the max tweet age for users

\*/

object MaxTweetAgeParam

extends FSBoundedParam[Duration](

name = "tweet\_age\_max\_hours",

default = 24.hours,

min = 1.hours,

max = 72.hours

)

with HasDurationConversion {

override val durationConversion = DurationConversion.FromHours

}

/\*\*

\* Param to control the max tweet age for modeling-based candidates

\*/

object ModelingBasedCandidateMaxTweetAgeParam

extends FSBoundedParam[Duration](

name = "tweet\_age\_candidate\_generation\_model\_max\_hours",

default = 24.hours,

min = 1.hours,

max = 72.hours

)

with HasDurationConversion {

override val durationConversion = DurationConversion.FromHours

}

/\*\*

\* Param to control the max tweet age for simcluster-based candidates

\*/

object GeoPopTweetMaxAgeInHours

extends FSBoundedParam[Duration](

name = "tweet\_age\_geo\_pop\_max\_hours",

default = 24.hours,

min = 1.hours,

max = 120.hours

)

with HasDurationConversion {

override val durationConversion = DurationConversion.FromHours

}

/\*\*

\* Param to control the max tweet age for simcluster-based candidates

\*/

object SimclusterBasedCandidateMaxTweetAgeParam

extends FSBoundedParam[Duration](

name = "tweet\_age\_simcluster\_max\_hours",

default = 24.hours,

min = 24.hours,

max = 48.hours

)

with HasDurationConversion {

override val durationConversion = DurationConversion.FromHours

}

/\*\*

\* Param to control the max tweet age for Detopic-based candidates

\*/

object DetopicBasedCandidateMaxTweetAgeParam

extends FSBoundedParam[Duration](

name = "tweet\_age\_detopic\_max\_hours",

default = 24.hours,

min = 24.hours,

max = 48.hours

)

with HasDurationConversion {

override val durationConversion = DurationConversion.FromHours

}

/\*\*

\* Param to control the max tweet age for F1 candidates

\*/

object F1CandidateMaxTweetAgeParam

extends FSBoundedParam[Duration](

name = "tweet\_age\_f1\_max\_hours",

default = 24.hours,

min = 1.hours,

max = 96.hours

)

with HasDurationConversion {

override val durationConversion = DurationConversion.FromHours

}

/\*\*

\* Param to control the max tweet age for Explore Video Tweet

\*/

object ExploreVideoTweetAgeParam

extends FSBoundedParam[Duration](

name = "explore\_video\_tweets\_age\_max\_hours",

default = 48.hours,

min = 1.hours,

max = 336.hours // Two weeks

)

with HasDurationConversion {

override val durationConversion = DurationConversion.FromHours

}

/\*\*

\* Param to no send for new user playbook push if user login for past hours

\*/

object NewUserPlaybookAllowedLastLoginHours

extends FSBoundedParam[Duration](

name = "new\_user\_playbook\_allowed\_last\_login\_hours",

default = 0.hours,

min = 0.hours,

max = 72.hours

)

with HasDurationConversion {

override val durationConversion = DurationConversion.FromHours

}

/\*\*

\* The batch size of RefreshForPushHandler's Take step

\*/

object NumberOfMaxCandidatesToBatchInRFPHTakeStep

extends FSBoundedParam(

name = "frigate\_push\_rfph\_batch\_take\_max\_size",

default = 1,

min = 1,

max = 10

)

/\*\*

\* The maximum number of candidates to batch for Importance Sampling

\*/

object NumberOfMaxCandidatesToBatchForImportanceSampling

extends FSBoundedParam(

name = "frigate\_push\_rfph\_max\_candidates\_to\_batch\_for\_importance\_sampling",

default = 65,

min = 1,

max = 500

)

/\*\*

\* Maximum number of regular MR push in 24.hours/daytime/nighttime

\*/

object MaxMrPushSends24HoursParam

extends FSBoundedParam(

name = "pushcap\_max\_sends\_24hours",

default = 5,

min = 0,

max = 12

)

/\*\*

\* Maximum number of regular MR ntab only channel in 24.hours/daytime/nighttime

\*/

object MaxMrNtabOnlySends24HoursParamV3

extends FSBoundedParam(

name = "pushcap\_max\_sends\_24hours\_ntabonly\_v3",

default = 5,

min = 0,

max = 12

)

/\*\*

\* Maximum number of regular MR ntab only in 24.hours/daytime/nighttime

\*/

object MaxMrPushSends24HoursNtabOnlyUsersParam

extends FSBoundedParam(

name = "pushcap\_max\_sends\_24hours\_ntab\_only",

default = 5,

min = 0,

max = 10

)

/\*\*

\* Customized PushCap offset (e.g., to the predicted value)

\*/

object CustomizedPushCapOffset

extends FSBoundedParam[Int](

name = "pushcap\_customized\_offset",

default = 0,

min = -2,

max = 4

)

/\*\*

\* Param to enable restricting minimum pushcap assigned with ML models

\* \*/

object EnableRestrictedMinModelPushcap

extends FSParam[Boolean](

name = "pushcap\_restricted\_model\_min\_enable",

default = false

)

/\*\*

\* Param to specify the minimum pushcap allowed to be assigned with ML models

\* \*/

object RestrictedMinModelPushcap

extends FSBoundedParam[Int](

name = "pushcap\_restricted\_model\_min\_value",

default = 1,

min = 0,

max = 9

)

object EnablePushcapRefactor

extends FSParam[Boolean](

name = "pushcap\_enable\_refactor",

default = false

)

/\*\*

\* Enables the restrict step in pushservice for a given user

\*

\* Setting this to false may cause a large number of candidates to be passed on to filtering/take

\* step in RefreshForPushHandler, increasing the service latency significantly

\*/

object EnableRestrictStep extends FSParam[Boolean]("frigate\_push\_rfph\_restrict\_step\_enable", true)

/\*\*

\* The number of candidates that are able to pass through the restrict step.

\*/

object RestrictStepSize

extends FSBoundedParam(

name = "frigate\_push\_rfph\_restrict\_step\_size",

default = 65,

min = 65,

max = 200

)

/\*\*

\* Number of max crMixer candidates to send.

\*/

object NumberOfMaxCrMixerCandidatesParam

extends FSBoundedParam(

name = "cr\_mixer\_migration\_max\_num\_of\_candidates\_to\_return",

default = 400,

min = 0,

max = 2000

)

/\*\*

\* Duration between two MR pushes

\*/

object MinDurationSincePushParam

extends FSBoundedParam[Duration](

name = "pushcap\_min\_duration\_since\_push\_hours",

default = 4.hours,

min = 0.hours,

max = 72.hours

)

with HasDurationConversion {

override val durationConversion = DurationConversion.FromHours

}

/\*\*

\* Each Phase duration to gradually ramp up MagicRecs for new users

\*/

object GraduallyRampUpPhaseDurationDays

extends FSBoundedParam[Duration](

name = "pushcap\_gradually\_ramp\_up\_phase\_duration\_days",

default = 3.days,

min = 2.days,

max = 7.days

)

with HasDurationConversion {

override val durationConversion = DurationConversion.FromDays

}

/\*\*

\* Param to specify interval for target pushcap fatigue

\*/

object TargetPushCapFatigueIntervalHours

extends FSBoundedParam[Duration](

name = "pushcap\_fatigue\_interval\_hours",

default = 24.hours,

min = 1.hour,

max = 240.hours

)

with HasDurationConversion {

override val durationConversion = DurationConversion.FromHours

}

/\*\*

\* Param to specify interval for target ntabOnly fatigue

\*/

object TargetNtabOnlyCapFatigueIntervalHours

extends FSBoundedParam[Duration](

name = "pushcap\_ntabonly\_fatigue\_interval\_hours",

default = 24.hours,

min = 1.hour,

max = 240.hours

)

with HasDurationConversion {

override val durationConversion = DurationConversion.FromHours

}

/\*\*

\* Param to use completely explicit push cap instead of LTV/modeling-based

\*/

object EnableExplicitPushCap

extends FSParam[Boolean](

name = "pushcap\_explicit\_enable",

default = false

)

/\*\*

\* Param to control explicit push cap (non-LTV)

\*/

object ExplicitPushCap

extends FSBoundedParam[Int](

name = "pushcap\_explicit\_value",

default = 1,

min = 0,

max = 20

)

/\*\*

\* Parameters for percentile thresholds of OpenOrNtabClick model in MR filtering model refreshing DDG

\*/

object PercentileThresholdCohort1

extends FSBoundedParam[Double](

name = "frigate\_push\_modeling\_percentile\_threshold\_cohort1",

default = 0.65,

min = 0.0,

max = 1.0

)

object PercentileThresholdCohort2

extends FSBoundedParam[Double](

name = "frigate\_push\_modeling\_percentile\_threshold\_cohort2",

default = 0.03,

min = 0.0,

max = 1.0

)

object PercentileThresholdCohort3

extends FSBoundedParam[Double](

name = "frigate\_push\_modeling\_percentile\_threshold\_cohort3",

default = 0.03,

min = 0.0,

max = 1.0

)

object PercentileThresholdCohort4

extends FSBoundedParam[Double](

name = "frigate\_push\_modeling\_percentile\_threshold\_cohort4",

default = 0.06,

min = 0.0,

max = 1.0

)

object PercentileThresholdCohort5

extends FSBoundedParam[Double](

name = "frigate\_push\_modeling\_percentile\_threshold\_cohort5",

default = 0.06,

min = 0.0,

max = 1.0

)

object PercentileThresholdCohort6

extends FSBoundedParam[Double](

name = "frigate\_push\_modeling\_percentile\_threshold\_cohort6",

default = 0.8,

min = 0.0,

max = 1.0

)

/\*\*

\* Parameters for percentile threshold list of OpenOrNtabCLick model in MR percentile grid search experiments

\*/

object MrPercentileGridSearchThresholdsCohort1

extends FSParam[Seq[Double]](

name = "frigate\_push\_modeling\_percentile\_grid\_search\_thresholds\_cohort1",

default = Seq(0.8, 0.75, 0.65, 0.55, 0.45, 0.35, 0.25)

)

object MrPercentileGridSearchThresholdsCohort2

extends FSParam[Seq[Double]](

name = "frigate\_push\_modeling\_percentile\_grid\_search\_thresholds\_cohort2",

default = Seq(0.15, 0.12, 0.1, 0.08, 0.06, 0.045, 0.03)

)

object MrPercentileGridSearchThresholdsCohort3

extends FSParam[Seq[Double]](

name = "frigate\_push\_modeling\_percentile\_grid\_search\_thresholds\_cohort3",

default = Seq(0.15, 0.12, 0.1, 0.08, 0.06, 0.045, 0.03)

)

object MrPercentileGridSearchThresholdsCohort4

extends FSParam[Seq[Double]](

name = "frigate\_push\_modeling\_percentile\_grid\_search\_thresholds\_cohort4",

default = Seq(0.15, 0.12, 0.1, 0.08, 0.06, 0.045, 0.03)

)

object MrPercentileGridSearchThresholdsCohort5

extends FSParam[Seq[Double]](

name = "frigate\_push\_modeling\_percentile\_grid\_search\_thresholds\_cohort5",

default = Seq(0.3, 0.2, 0.15, 0.1, 0.08, 0.06, 0.05)

)

object MrPercentileGridSearchThresholdsCohort6

extends FSParam[Seq[Double]](

name = "frigate\_push\_modeling\_percentile\_grid\_search\_thresholds\_cohort6",

default = Seq(0.8, 0.7, 0.6, 0.5, 0.4, 0.3, 0.2)

)

/\*\*

\* Parameters for threshold list of OpenOrNtabClick model in MF grid search experiments

\*/

object MfGridSearchThresholdsCohort1

extends FSParam[Seq[Double]](

name = "frigate\_push\_modeling\_mf\_grid\_search\_thresholds\_cohort1",

default = Seq(0.030, 0.040, 0.050, 0.062, 0.070, 0.080, 0.090) // default: 0.062

)

object MfGridSearchThresholdsCohort2

extends FSParam[Seq[Double]](

name = "frigate\_push\_modeling\_mf\_grid\_search\_thresholds\_cohort2",

default = Seq(0.005, 0.010, 0.015, 0.020, 0.030, 0.040, 0.050) // default: 0.020

)

object MfGridSearchThresholdsCohort3

extends FSParam[Seq[Double]](

name = "frigate\_push\_modeling\_mf\_grid\_search\_thresholds\_cohort3",

default = Seq(0.010, 0.015, 0.020, 0.025, 0.035, 0.045, 0.055) // default: 0.025

)

object MfGridSearchThresholdsCohort4

extends FSParam[Seq[Double]](

name = "frigate\_push\_modeling\_mf\_grid\_search\_thresholds\_cohort4",

default = Seq(0.015, 0.020, 0.025, 0.030, 0.040, 0.050, 0.060) // default: 0.030

)

object MfGridSearchThresholdsCohort5

extends FSParam[Seq[Double]](

name = "frigate\_push\_modeling\_mf\_grid\_search\_thresholds\_cohort5",

default = Seq(0.035, 0.040, 0.045, 0.050, 0.060, 0.070, 0.080) // default: 0.050

)

object MfGridSearchThresholdsCohort6

extends FSParam[Seq[Double]](

name = "frigate\_push\_modeling\_mf\_grid\_search\_thresholds\_cohort6",

default = Seq(0.040, 0.045, 0.050, 0.055, 0.065, 0.075, 0.085) // default: 0.055

)

/\*\*

\* Param to specify which global optout models to use to first predict the global scores for users

\*/

object GlobalOptoutModelParam

extends FSParam[Seq[OptoutModel.ModelNameType]](

name = "optout\_model\_global\_model\_ids",

default = Seq.empty[OptoutModel.ModelNameType]

)

/\*\*

\* Param to specify which optout model to use according to the experiment bucket

\*/

object BucketOptoutModelParam

extends FSParam[OptoutModel.ModelNameType](

name = "optout\_model\_bucket\_model\_id",

default = OptoutModel.D0\_has\_realtime\_features

)

/\*

\* Param to enable candidate generation model

\* \*/

object EnableCandidateGenerationModelParam

extends FSParam[Boolean](

name = "candidate\_generation\_model\_enable",

default = false

)

object EnableOverrideForSportsCandidates

extends FSParam[Boolean](name = "magicfanout\_sports\_event\_enable\_override", default = true)

object EnableEventIdBasedOverrideForSportsCandidates

extends FSParam[Boolean](

name = "magicfanout\_sports\_event\_enable\_event\_id\_based\_override",

default = true)

/\*\*

\* Param to specify the threshold to determine if a user’s optout score is high enough to enter the experiment.

\*/

object GlobalOptoutThresholdParam

extends FSParam[Seq[Double]](

name = "optout\_model\_global\_thresholds",

default = Seq(1.0, 1.0)

)

/\*\*

\* Param to specify the threshold to determine if a user’s optout score is high enough to be assigned

\* with a reduced pushcap based on the bucket membership.

\*/

object BucketOptoutThresholdParam

extends FSBoundedParam[Double](

name = "optout\_model\_bucket\_threshold",

default = 1.0,

min = 0.0,

max = 1.0

)

/\*\*

\* Param to specify the reduced pushcap value if the optout probability predicted by the bucket

\* optout model is higher than the specified bucket optout threshold.

\*/

object OptoutExptPushCapParam

extends FSBoundedParam[Int](

name = "optout\_model\_expt\_push\_cap",

default = 10,

min = 0,

max = 10

)

/\*\*

\* Param to specify the thresholds to determine which push cap slot the user should be assigned to

\* according to the optout score. For example,the slot thresholds are [0.1, 0.2, ..., 1.0], the user

\* is assigned to the second slot if the optout score is in (0.1, 0.2].

\*/

object BucketOptoutSlotThresholdParam

extends FSParam[Seq[Double]](

name = "optout\_model\_bucket\_slot\_thresholds",

default = Seq.empty[Double]

)

/\*\*

\* Param to specify the adjusted push cap of each slot. For example, if the slot push caps are [1, 2, ..., 10]

\* and the user is assigned to the 2nd slot according to the optout score, the push cap of the user

\* will be adjusted to 2.

\*/

object BucketOptoutSlotPushcapParam

extends FSParam[Seq[Int]](

name = "optout\_model\_bucket\_slot\_pushcaps",

default = Seq.empty[Int]

)

/\*\*

\* Param to specify if the optout score based push cap adjustment is enabled

\*/

object EnableOptoutAdjustedPushcap

extends FSParam[Boolean](

"optout\_model\_enable\_optout\_adjusted\_pushcap",

false

)

/\*\*

\* Param to specify which weighted open or ntab click model to use

\*/

object WeightedOpenOrNtabClickRankingModelParam

extends FSParam[WeightedOpenOrNtabClickModel.ModelNameType](

name = "frigate\_push\_modeling\_oonc\_ranking\_model\_id",

default = WeightedOpenOrNtabClickModel.Periodically\_Refreshed\_Prod\_Model

)

/\*\*

\* Param to disable heavy ranker

\*/

object DisableHeavyRankingModelFSParam

extends FSParam[Boolean](

name = "frigate\_push\_modeling\_disable\_heavy\_ranking",

default = false

)

/\*\*

\* Param to specify which weighted open or ntab click model to use for Android modelling experiment

\*/

object WeightedOpenOrNtabClickRankingModelForAndroidParam

extends FSParam[WeightedOpenOrNtabClickModel.ModelNameType](

name = "frigate\_push\_modeling\_oonc\_ranking\_model\_for\_android\_id",

default = WeightedOpenOrNtabClickModel.Periodically\_Refreshed\_Prod\_Model

)

/\*\*

\* Param to specify which weighted open or ntab click model to use for FILTERING

\*/

object WeightedOpenOrNtabClickFilteringModelParam

extends FSParam[WeightedOpenOrNtabClickModel.ModelNameType](

name = "frigate\_push\_modeling\_oonc\_filtering\_model\_id",

default = WeightedOpenOrNtabClickModel.Periodically\_Refreshed\_Prod\_Model

)

/\*\*

\* Param to specify which quality predicate to use for ML filtering

\*/

object QualityPredicateIdParam

extends FSEnumParam[QualityPredicateEnum.type](

name = "frigate\_push\_modeling\_quality\_predicate\_id",

default = QualityPredicateEnum.WeightedOpenOrNtabClick,

enum = QualityPredicateEnum

)

/\*\*

\* Param to control threshold for any quality predicates using explicit thresholds

\*/

object QualityPredicateExplicitThresholdParam

extends FSBoundedParam[Double](

name = "frigate\_push\_modeling\_quality\_predicate\_explicit\_threshold",

default = 0.1,

min = 0,

max = 1)

/\*\*

\* MagicFanout relaxed eventID fatigue interval (when we want to enable multiple updates for the same event)

\*/

object MagicFanoutRelaxedEventIdFatigueIntervalInHours

extends FSBoundedParam[Int](

name = "frigate\_push\_magicfanout\_relaxed\_event\_id\_fatigue\_interval\_in\_hours",

default = 24,

min = 0,

max = 720

)

/\*\*

\* MagicFanout DenyListed Countries

\*/

object MagicFanoutDenyListedCountries

extends FSParam[Seq[String]](

"frigate\_push\_magicfanout\_denylisted\_countries",

Seq.empty[String])

object MagicFanoutSportsEventDenyListedCountries

extends FSParam[Seq[String]](

"magicfanout\_sports\_event\_denylisted\_countries",

Seq.empty[String])

/\*\*

\* MagicFanout maximum erg rank for a given push event for non heavy users

\*/

object MagicFanoutRankErgThresholdNonHeavy

extends FSBoundedParam[Int](

name = "frigate\_push\_magicfanout\_erg\_rank\_threshold\_non\_heavy",

default = 25,

min = 1,

max = 50

)

/\*\*

\* MagicFanout maximum erg rank for a given push event for heavy users

\*/

object MagicFanoutRankErgThresholdHeavy

extends FSBoundedParam[Int](

name = "frigate\_push\_magicfanout\_erg\_rank\_threshold\_heavy",

default = 20,

min = 1,

max = 50

)

object EnablePushMixerReplacingAllSources

extends FSParam[Boolean](

name = "push\_mixer\_enable\_replacing\_all\_sources",

default = false

)

object EnablePushMixerReplacingAllSourcesWithControl

extends FSParam[Boolean](

name = "push\_mixer\_enable\_replacing\_all\_sources\_with\_control",

default = false

)

object EnablePushMixerReplacingAllSourcesWithExtra

extends FSParam[Boolean](

name = "push\_mixer\_enable\_replacing\_all\_sources\_with\_extra",

default = false

)

object EnablePushMixerSource

extends FSParam[Boolean](

name = "push\_mixer\_enable\_source",

default = false

)

object PushMixerMaxResults

extends FSBoundedParam[Int](

name = "push\_mixer\_max\_results",

default = 10,

min = 1,

max = 5000

)

/\*\*

\* Enable tweets from trends that have been annotated by curators

\*/

object EnableCuratedTrendTweets

extends FSParam[Boolean](name = "trend\_tweet\_curated\_trends\_enable", default = false)

/\*\*

\* Enable tweets from trends that haven't been annotated by curators

\*/

object EnableNonCuratedTrendTweets

extends FSParam[Boolean](name = "trend\_tweet\_non\_curated\_trends\_enable", default = false)

/\*\*

\* Maximum trend tweet notifications in fixed duration

\*/

object MaxTrendTweetNotificationsInDuration

extends FSBoundedParam[Int](

name = "trend\_tweet\_max\_notifications\_in\_duration",

min = 0,

default = 0,

max = 20)

/\*\*

\* Duration in days over which trend tweet notifications fatigue is applied

\*/

object TrendTweetNotificationsFatigueDuration

extends FSBoundedParam[Duration](

name = "trend\_tweet\_notifications\_fatigue\_in\_days",

default = 1.day,

min = Duration.Bottom,

max = Duration.Top

)

with HasDurationConversion {

override val durationConversion = DurationConversion.FromDays

}

/\*\*

\* Maximum number of trends candidates to query from event-recos endpoint

\*/

object MaxRecommendedTrendsToQuery

extends FSBoundedParam[Int](

name = "trend\_tweet\_max\_trends\_to\_query",

min = 0,

default = 0,

max = 100)

/\*\*

\* Fix missing event-associated interests in MagicFanoutNoOptoutInterestsPredicate

\*/

object MagicFanoutFixNoOptoutInterestsBugParam

extends FSParam[Boolean]("frigate\_push\_magicfanout\_fix\_no\_optout\_interests", default = true)

object EnableSimclusterOfflineAggFeatureForExpt

extends FSParam[Boolean]("frigate\_enable\_simcluster\_offline\_agg\_feature", false)

/\*\*

\* Param to enable removal of UTT domain for

\*/

object ApplyMagicFanoutBroadEntityInterestRankThresholdPredicate

extends FSParam[Boolean](

"frigate\_push\_magicfanout\_broad\_entity\_interest\_rank\_threshold\_predicate",

false

)

object HydrateEventReasonsFeatures

extends FSParam[Boolean](

name = "frigate\_push\_magicfanout\_hydrate\_event\_reasons\_features",

false

)

/\*\*

\* Param to enable online MR history features

\*/

object EnableHydratingOnlineMRHistoryFeatures

extends FSParam[Boolean](

name = "feature\_hydration\_online\_mr\_history",

default = false

)

/\*\*

\* Param to enable bold title on favorite and retweet push copy for Android in DDG 10220

\*/

object MRBoldTitleFavoriteAndRetweetParam

extends FSEnumParam[MRBoldTitleFavoriteAndRetweetExperimentEnum.type](

name = "frigate\_push\_bold\_title\_favorite\_and\_retweet\_id",

default = MRBoldTitleFavoriteAndRetweetExperimentEnum.ShortTitle,

enum = MRBoldTitleFavoriteAndRetweetExperimentEnum

)

/\*\*

\* Param to enable high priority push

\*/

object EnableHighPriorityPush

extends FSParam[Boolean]("frigate\_push\_magicfanout\_enable\_high\_priority\_push", false)

/\*\*

\* Param to redirect sports crt event to a custom url

\*/

object EnableSearchURLRedirectForSportsFanout

extends FSParam[Boolean]("magicfanout\_sports\_event\_enable\_search\_url\_redirect", false)

/\*\*

\* Param to enable score fanout notification for sports

\*/

object EnableScoreFanoutNotification

extends FSParam[Boolean]("magicfanout\_sports\_event\_enable\_score\_fanout", false)

/\*\*

\* Param to add custom search url for sports crt event

\*/

object SearchURLRedirectForSportsFanout

extends FSParam[String](

name = "magicfanout\_sports\_event\_search\_url\_redirect",

default = "https://twitter.com/explore/tabs/ipl",

)

/\*\*

\* Param to enable high priority sports push

\*/

object EnableHighPrioritySportsPush

extends FSParam[Boolean]("magicfanout\_sports\_event\_enable\_high\_priority\_push", false)

/\*\*

\* Param to control rank threshold for magicfanout user follow

\*/

object MagicFanoutRealgraphRankThreshold

extends FSBoundedParam[Int](

name = "magicfanout\_realgraph\_threshold",

default = 500,

max = 500,

min = 100

)

/\*\*

\* Topic score threshold for topic proof tweet candidates topic annotations

\* \*/

object TopicProofTweetCandidatesTopicScoreThreshold

extends FSBoundedParam[Double](

name = "topics\_as\_social\_proof\_topic\_score\_threshold",

default = 0.0,

min = 0.0,

max = 100.0

)

/\*\*

\* Enable Topic Proof Tweet Recs

\*/

object EnableTopicProofTweetRecs

extends FSParam[Boolean](name = "topics\_as\_social\_proof\_enable", default = true)

/\*\*

\* Enable health filters for topic tweet notifications

\*/

object EnableHealthFiltersForTopicProofTweet

extends FSParam[Boolean](

name = "topics\_as\_social\_proof\_enable\_health\_filters",

default = false)

/\*\*

\* Disable health filters for CrMixer candidates

\*/

object DisableHealthFiltersForCrMixerCandidates

extends FSParam[Boolean](

name = "health\_and\_quality\_filter\_disable\_for\_crmixer\_candidates",

default = false)

object EnableMagicFanoutNewsForYouNtabCopy

extends FSParam[Boolean](name = "send\_handler\_enable\_nfy\_ntab\_copy", default = false)

/\*\*

\* Param to enable semi-personalized high quality candidates in pushservice

\* \*/

object HighQualityCandidatesEnableCandidateSource

extends FSParam[Boolean](

name = "high\_quality\_candidates\_enable\_candidate\_source",

default = false

)

/\*\*

\* Param to decide semi-personalized high quality candidates

\* \*/

object HighQualityCandidatesEnableGroups

extends FSEnumSeqParam[HighQualityCandidateGroupEnum.type](

name = "high\_quality\_candidates\_enable\_groups\_ids",

default = Seq(AgeBucket, Language),

enum = HighQualityCandidateGroupEnum

)

/\*\*

\* Param to decide semi-personalized high quality candidates

\* \*/

object HighQualityCandidatesNumberOfCandidates

extends FSBoundedParam[Int](

name = "high\_quality\_candidates\_number\_of\_candidates",

default = 0,

min = 0,

max = Int.MaxValue

)

/\*\*

\* Param to enable small domain falling back to bigger domains for high quality candidates in pushservice

\* \*/

object HighQualityCandidatesEnableFallback

extends FSParam[Boolean](

name = "high\_quality\_candidates\_enable\_fallback",

default = false

)

/\*\*

\* Param to decide whether to fallback to bigger domain for high quality candidates

\* \*/

object HighQualityCandidatesMinNumOfCandidatesToFallback

extends FSBoundedParam[Int](

name = "high\_quality\_candidates\_min\_num\_of\_candidates\_to\_fallback",

default = 50,

min = 0,

max = Int.MaxValue

)

/\*\*

\* Param to specific source ids for high quality candidates

\* \*/

object HighQualityCandidatesFallbackSourceIds

extends FSParam[Seq[String]](

name = "high\_quality\_candidates\_fallback\_source\_ids",

default = Seq("HQ\_C\_COUNT\_PASS\_QUALITY\_SCORES"))

/\*\*

\* Param to decide groups for semi-personalized high quality candidates

\* \*/

object HighQualityCandidatesFallbackEnabledGroups

extends FSEnumSeqParam[HighQualityCandidateGroupEnum.type](

name = "high\_quality\_candidates\_fallback\_enabled\_groups\_ids",

default = Seq(Country),

enum = HighQualityCandidateGroupEnum

)

/\*\*

\* Param to control what heavy ranker model to use for scribing scores

\*/

object HighQualityCandidatesHeavyRankingModel

extends FSParam[String](

name = "high\_quality\_candidates\_heavy\_ranking\_model",

default = "Periodically\_Refreshed\_Prod\_Model\_V11"

)

/\*\*

\* Param to control what non personalized quality "Cnn" model to use for scribing scores

\*/

object HighQualityCandidatesNonPersonalizedQualityCnnModel

extends FSParam[String](

name = "high\_quality\_candidates\_non\_personalized\_quality\_cnn\_model",

default = "Q1\_2023\_Mr\_Tf\_Quality\_Model\_cnn"

)

/\*\*

\* Param to control what nsfw health model to use for scribing scores

\*/

object HighQualityCandidatesBqmlNsfwModel

extends FSParam[String](

name = "high\_quality\_candidates\_bqml\_nsfw\_model",

default = "Q2\_2022\_Mr\_Bqml\_Health\_Model\_NsfwV0"

)

/\*\*

\* Param to control what reportodel to use for scribing scores

\*/

object HighQualityCandidatesBqmlReportModel

extends FSParam[String](

name = "high\_quality\_candidates\_bqml\_report\_model",

default = "Q3\_2022\_15266\_Mr\_Bqml\_Non\_Personalized\_Report\_Model\_with\_Media\_Embeddings"

)

/\*\*

\* Param to specify the threshold to determine if a tweet contains nudity media

\*/

object TweetMediaSensitiveCategoryThresholdParam

extends FSBoundedParam[Double](

name = "tweet\_media\_sensitive\_category\_threshold",

default = 1.0,

min = 0.0,

max = 1.0

)

/\*\*

\* Param to boost candidates from subscription creators

\*/

object BoostCandidatesFromSubscriptionCreators

extends FSParam[Boolean](

name = "subscription\_enable\_boost\_candidates\_from\_active\_creators",

default = false

)

/\*\*

\* Param to soft rank candidates from subscription creators

\*/

object SoftRankCandidatesFromSubscriptionCreators

extends FSParam[Boolean](

name = "subscription\_enable\_soft\_rank\_candidates\_from\_active\_creators",

default = false

)

/\*\*

\* Param as factor to control how much we want to boost creator tweets

\*/

object SoftRankFactorForSubscriptionCreators

extends FSBoundedParam[Double](

name = "subscription\_soft\_rank\_factor\_for\_boost",

default = 1.0,

min = 0.0,

max = Double.MaxValue

)

/\*\*

\* Param to enable new OON copy for Push Notifications

\*/

object EnableNewMROONCopyForPush

extends FSParam[Boolean](

name = "mr\_copy\_enable\_new\_mr\_oon\_copy\_push",

default = true

)

/\*\*

\* Param to enable generated inline actions on OON Notifications

\*/

object EnableOONGeneratedInlineActions

extends FSParam[Boolean](

name = "mr\_inline\_enable\_oon\_generated\_actions",

default = false

)

/\*\*

\* Param to control dynamic inline actions for Out-of-Network copies

\*/

object OONTweetDynamicInlineActionsList

extends FSEnumSeqParam[InlineActionsEnum.type](

name = "mr\_inline\_oon\_tweet\_dynamic\_action\_ids",

default = Seq(Follow, Retweet, Favorite),

enum = InlineActionsEnum

)

object HighOONCTweetFormat

extends FSEnumParam[IbisTemplateFormatEnum.type](

name = "mr\_copy\_high\_oonc\_format\_id",

default = IbisTemplateFormatEnum.template1,

enum = IbisTemplateFormatEnum

)

object LowOONCTweetFormat

extends FSEnumParam[IbisTemplateFormatEnum.type](

name = "mr\_copy\_low\_oonc\_format\_id",

default = IbisTemplateFormatEnum.template1,

enum = IbisTemplateFormatEnum

)

/\*\*

\* Param to enable dynamic inline actions based on FSParams for Tweet copies (not OON)

\*/

object EnableTweetDynamicInlineActions

extends FSParam[Boolean](

name = "mr\_inline\_enable\_tweet\_dynamic\_actions",

default = false

)

/\*\*

\* Param to control dynamic inline actions for Tweet copies (not OON)

\*/

object TweetDynamicInlineActionsList

extends FSEnumSeqParam[InlineActionsEnum.type](

name = "mr\_inline\_tweet\_dynamic\_action\_ids",

default = Seq(Reply, Retweet, Favorite),

enum = InlineActionsEnum

)

object UseInlineActionsV1

extends FSParam[Boolean](

name = "mr\_inline\_use\_inline\_action\_v1",

default = true

)

object UseInlineActionsV2

extends FSParam[Boolean](

name = "mr\_inline\_use\_inline\_action\_v2",

default = false

)

object EnableInlineFeedbackOnPush

extends FSParam[Boolean](

name = "mr\_inline\_enable\_inline\_feedback\_on\_push",

default = false

)

object InlineFeedbackSubstitutePosition

extends FSBoundedParam[Int](

name = "mr\_inline\_feedback\_substitute\_position",

min = 0,

max = 2,

default = 2, // default to substitute or append last inline action

)

/\*\*

\* Param to control dynamic inline actions for web notifications

\*/

object EnableDynamicInlineActionsForDesktopWeb

extends FSParam[Boolean](

name = "mr\_inline\_enable\_dynamic\_actions\_for\_desktop\_web",

default = false

)

object EnableDynamicInlineActionsForMobileWeb

extends FSParam[Boolean](

name = "mr\_inline\_enable\_dynamic\_actions\_for\_mobile\_web",

default = false

)

/\*\*

\* Param to define dynamic inline action types for web notifications (both desktop web + mobile web)

\*/

object TweetDynamicInlineActionsListForWeb

extends FSEnumSeqParam[InlineActionsEnum.type](

name = "mr\_inline\_tweet\_dynamic\_action\_for\_web\_ids",

default = Seq(Retweet, Favorite),

enum = InlineActionsEnum

)

/\*\*

\* Param to enable MR Override Notifications for Android

\*/

object EnableOverrideNotificationsForAndroid

extends FSParam[Boolean](

name = "mr\_override\_enable\_override\_notifications\_for\_android",

default = false

)

/\*\*

\* Param to enable MR Override Notifications for iOS

\*/

object EnableOverrideNotificationsForIos

extends FSParam[Boolean](

name = "mr\_override\_enable\_override\_notifications\_for\_ios",

default = false

)

/\*\*

\* Param to enable gradually ramp up notification

\*/

object EnableGraduallyRampUpNotification

extends FSParam[Boolean](

name = "pushcap\_gradually\_ramp\_up\_enable",

default = false

)

/\*\*

\* Param to control the minInrerval for fatigue between consecutive MFNFY pushes

\*/

object MFMinIntervalFatigue

extends FSBoundedParam[Duration](

name = "frigate\_push\_magicfanout\_fatigue\_min\_interval\_consecutive\_pushes\_minutes",

default = 240.minutes,

min = Duration.Bottom,

max = Duration.Top)

with HasDurationConversion {

override val durationConversion = DurationConversion.FromMinutes

}

/\*\*

\* Param to control the interval for MFNFY pushes

\*/

object MFPushIntervalInHours

extends FSBoundedParam[Duration](

name = "frigate\_push\_magicfanout\_fatigue\_push\_interval\_in\_hours",

default = 24.hours,

min = Duration.Bottom,

max = Duration.Top)

with HasDurationConversion {

override val durationConversion = DurationConversion.FromHours

}

/\*\*

\* Param to control the maximum number of Sports MF pushes in a period of time

\*/

object SportsMaxNumberOfPushesInInterval

extends FSBoundedParam[Int](

name = "magicfanout\_sports\_event\_fatigue\_max\_pushes\_in\_interval",

default = 2,

min = 0,

max = 6)

/\*\*

\* Param to control the minInterval for fatigue between consecutive sports pushes

\*/

object SportsMinIntervalFatigue

extends FSBoundedParam[Duration](

name = "magicfanout\_sports\_event\_fatigue\_min\_interval\_consecutive\_pushes\_minutes",

default = 240.minutes,

min = Duration.Bottom,

max = Duration.Top)

with HasDurationConversion {

override val durationConversion = DurationConversion.FromMinutes

}

/\*\*

\* Param to control the interval for sports pushes

\*/

object SportsPushIntervalInHours

extends FSBoundedParam[Duration](

name = "magicfanout\_sports\_event\_fatigue\_push\_interval\_in\_hours",

default = 24.hours,

min = Duration.Bottom,

max = Duration.Top)

with HasDurationConversion {

override val durationConversion = DurationConversion.FromHours

}

/\*\*

\* Param to control the maximum number of same event sports MF pushes in a period of time

\*/

object SportsMaxNumberOfPushesInIntervalPerEvent

extends FSBoundedParam[Int](

name = "magicfanout\_sports\_event\_fatigue\_max\_pushes\_in\_per\_event\_interval",

default = 2,

min = 0,

max = 6)

/\*\*

\* Param to control the minInterval for fatigue between consecutive same event sports pushes

\*/

object SportsMinIntervalFatiguePerEvent

extends FSBoundedParam[Duration](

name = "magicfanout\_sports\_event\_fatigue\_min\_interval\_consecutive\_pushes\_per\_event\_minutes",

default = 240.minutes,

min = Duration.Bottom,

max = Duration.Top)

with HasDurationConversion {

override val durationConversion = DurationConversion.FromMinutes

}

/\*\*

\* Param to control the interval for same event sports pushes

\*/

object SportsPushIntervalInHoursPerEvent

extends FSBoundedParam[Duration](

name = "magicfanout\_sports\_event\_fatigue\_push\_interval\_per\_event\_in\_hours",

default = 24.hours,

min = Duration.Bottom,

max = Duration.Top)

with HasDurationConversion {

override val durationConversion = DurationConversion.FromHours

}

/\*\*

\* Param to control the maximum number of MF pushes in a period of time

\*/

object MFMaxNumberOfPushesInInterval

extends FSBoundedParam[Int](

name = "frigate\_push\_magicfanout\_fatigue\_max\_pushes\_in\_interval",

default = 2,

min = 0,

max = 6)

/\*\*

\* Param to enable custom duration for fatiguing

\*/

object GPEnableCustomMagicFanoutCricketFatigue

extends FSParam[Boolean](

name = "global\_participation\_cricket\_magicfanout\_enable\_custom\_fatigue",

default = false

)

/\*\*

\* Param to enable e2e scribing for target filtering step

\*/

object EnableMrRequestScribingForTargetFiltering

extends FSParam[Boolean](

name = "mr\_request\_scribing\_enable\_for\_target\_filtering",

default = false

)

/\*\*

\* Param to enable e2e scribing for candidate filtering step

\*/

object EnableMrRequestScribingForCandidateFiltering

extends FSParam[Boolean](

name = "mr\_request\_scribing\_enable\_for\_candidate\_filtering",

default = false

)

/\*\*

\* Param to enable e2e scribing with feature hydrating

\*/

object EnableMrRequestScribingWithFeatureHydrating

extends FSParam[Boolean](

name = "mr\_request\_scribing\_enable\_with\_feature\_hydrating",

default = false

)

/\*

\* TargetLevel Feature list for Mr request scribing

\*/

object TargetLevelFeatureListForMrRequestScribing

extends FSParam[Seq[String]](

name = "mr\_request\_scribing\_target\_level\_feature\_list",

default = Seq.empty

)

/\*\*

\* Param to enable \eps-greedy exploration for BigFiltering/LTV-based filtering

\*/

object EnableMrRequestScribingForEpsGreedyExploration

extends FSParam[Boolean](

name = "mr\_request\_scribing\_eps\_greedy\_exploration\_enable",

default = false

)

/\*\*

\* Param to control epsilon in \eps-greedy exploration for BigFiltering/LTV-based filtering

\*/

object MrRequestScribingEpsGreedyExplorationRatio

extends FSBoundedParam[Double](

name = "mr\_request\_scribing\_eps\_greedy\_exploration\_ratio",

default = 0.0,

min = 0.0,

max = 1.0

)

/\*\*

\* Param to enable scribing dismiss model score

\*/

object EnableMrRequestScribingDismissScore

extends FSParam[Boolean](

name = "mr\_request\_scribing\_dismiss\_score\_enable",

default = false

)

/\*\*

\* Param to enable scribing BigFiltering supervised model(s) score(s)

\*/

object EnableMrRequestScribingBigFilteringSupervisedScores

extends FSParam[Boolean](

name = "mr\_request\_scribing\_bigfiltering\_supervised\_scores\_enable",

default = false

)

/\*\*

\* Param to enable scribing BigFiltering RL model(s) score(s)

\*/

object EnableMrRequestScribingBigFilteringRLScores

extends FSParam[Boolean](

name = "mr\_request\_scribing\_bigfiltering\_rl\_scores\_enable",

default = false

)

/\*\*

\* Param to flatten mr request scribe

\*/

object EnableFlattenMrRequestScribing

extends FSParam[Boolean](

name = "mr\_request\_scribing\_enable\_flatten",

default = false

)

/\*\*

\* Param to enable NSFW token based filtering

\*/

object EnableNsfwTokenBasedFiltering

extends FSParam[Boolean](

name = "health\_and\_quality\_filter\_enable\_nsfw\_token\_based\_filtering",

default = false

)

object NsfwTokensParam

extends FSParam[Seq[String]](

name = "health\_and\_quality\_filter\_nsfw\_tokens",

default = Seq("nsfw", "18+", "\uD83D\uDD1E"))

object MinimumAllowedAuthorAccountAgeInHours

extends FSBoundedParam[Int](

name = "health\_and\_quality\_filter\_minimum\_allowed\_author\_account\_age\_in\_hours",

default = 0,

min = 0,

max = 168

)

/\*\*

\* Param to enable the profanity filter

\*/

object EnableProfanityFilterParam

extends FSParam[Boolean](

name = "health\_and\_quality\_filter\_enable\_profanity\_filter",

default = false

)

/\*\*

\* Param to enable query the author media representation store

\*/

object EnableQueryAuthorMediaRepresentationStore

extends FSParam[Boolean](

name = "health\_and\_quality\_filter\_enable\_query\_author\_media\_representation\_store",

default = false

)

/\*\*

\* Threshold to filter a tweet based on the author sensitive media score

\*/

object AuthorSensitiveMediaFilteringThreshold

extends FSBoundedParam[Double](

name = "health\_and\_quality\_filter\_author\_sensitive\_media\_filtering\_threshold",

default = 1.0,

min = 0.0,

max = 1.0

)

/\*\*

\* Threshold to filter a tweet based on the author sensitive media score

\*/

object AuthorSensitiveMediaFilteringThresholdForMrTwistly

extends FSBoundedParam[Double](

name = "health\_and\_quality\_filter\_author\_sensitive\_media\_filtering\_threshold\_for\_mrtwistly",

default = 1.0,

min = 0.0,

max = 1.0

)

/\*\*

\* Param to enable filtering the SimCluster tweet if it has AbuseStrike\_Top2Percent entitiy

\*/

object EnableAbuseStrikeTop2PercentFilterSimCluster

extends FSParam[Boolean](

name = "health\_signal\_store\_enable\_abuse\_strike\_top\_2\_percent\_filter\_sim\_cluster",

default = false

)

/\*\*

\* Param to enable filtering the SimCluster tweet if it has AbuseStrike\_Top1Percent entitiy

\*/

object EnableAbuseStrikeTop1PercentFilterSimCluster

extends FSParam[Boolean](

name = "health\_signal\_store\_enable\_abuse\_strike\_top\_1\_percent\_filter\_sim\_cluster",

default = false

)

/\*\*

\* Param to enable filtering the SimCluster tweet if it has AbuseStrike\_Top0.5Percent entitiy

\*/

object EnableAbuseStrikeTop05PercentFilterSimCluster

extends FSParam[Boolean](

name = "health\_signal\_store\_enable\_abuse\_strike\_top\_05\_percent\_filter\_sim\_cluster",

default = false

)

object EnableAgathaUserHealthModelPredicate

extends FSParam[Boolean](

name = "health\_signal\_store\_enable\_agatha\_user\_health\_model\_predicate",

default = false

)

/\*\*

\* Threshold to filter a tweet based on the agatha\_calibrated\_nsfw score of its author for MrTwistly

\*/

object AgathaCalibratedNSFWThresholdForMrTwistly

extends FSBoundedParam[Double](

name = "health\_signal\_store\_agatha\_calibrated\_nsfw\_threshold\_for\_mrtwistly",

default = 1.0,

min = 0.0,

max = 1.0

)

/\*\*

\* Threshold to filter a tweet based on the agatha\_calibrated\_nsfw score of its author

\*/

object AgathaCalibratedNSFWThreshold

extends FSBoundedParam[Double](

name = "health\_signal\_store\_agatha\_calibrated\_nsfw\_threshold",

default = 1.0,

min = 0.0,

max = 1.0

)

/\*\*

\* Threshold to filter a tweet based on the agatha\_nsfw\_text\_user score of its author for MrTwistly

\*/

object AgathaTextNSFWThresholdForMrTwistly

extends FSBoundedParam[Double](

name = "health\_signal\_store\_agatha\_text\_nsfw\_threshold\_for\_mrtwistly",

default = 1.0,

min = 0.0,

max = 1.0

)

/\*\*

\* Threshold to filter a tweet based on the agatha\_nsfw\_text\_user score of its author

\*/

object AgathaTextNSFWThreshold

extends FSBoundedParam[Double](

name = "health\_signal\_store\_agatha\_text\_nsfw\_threshold",

default = 1.0,

min = 0.0,

max = 1.0

)

/\*\*

\* Threshold to bucket a user based on the agatha\_calibrated\_nsfw score of the tweet author

\*/

object AgathaCalibratedNSFWBucketThreshold

extends FSBoundedParam[Double](

name = "health\_signal\_store\_agatha\_calibrated\_nsfw\_bucket\_threshold",

default = 1.0,

min = 0.0,

max = 1.0

)

/\*\*

\* Threshold to bucket a user based on the agatha\_nsfw\_text\_user score of the tweet author

\*/

object AgathaTextNSFWBucketThreshold

extends FSBoundedParam[Double](

name = "health\_signal\_store\_agatha\_text\_nsfw\_bucket\_threshold",

default = 1.0,

min = 0.0,

max = 1.0

)

/\*\*

\* Param to enable filtering using pnsfw\_text\_tweet model.

\*/

object EnableHealthSignalStorePnsfwTweetTextPredicate

extends FSParam[Boolean](

name = "health\_signal\_store\_enable\_pnsfw\_tweet\_text\_predicate",

default = false

)

/\*\*

\* Threshold score for filtering based on pnsfw\_text\_tweet Model.

\*/

object PnsfwTweetTextThreshold

extends FSBoundedParam[Double](

name = "health\_signal\_store\_pnsfw\_tweet\_text\_threshold",

default = 1.0,

min = 0.0,

max = 1.0

)

/\*\*

\* Threshold score for bucketing based on pnsfw\_text\_tweet Model.

\*/

object PnsfwTweetTextBucketingThreshold

extends FSBoundedParam[Double](

name = "health\_signal\_store\_pnsfw\_tweet\_text\_bucketing\_threshold",

default = 1.0,

min = 0.0,

max = 1.0

)

/\*\*

\* Enable filtering tweets with media based on pnsfw\_media\_tweet Model for OON tweets only.

\*/

object PnsfwTweetMediaFilterOonOnly

extends FSParam[Boolean](

name = "health\_signal\_store\_pnsfw\_tweet\_media\_filter\_oon\_only",

default = true

)

/\*\*

\* Threshold score for filtering tweets with media based on pnsfw\_media\_tweet Model.

\*/

object PnsfwTweetMediaThreshold

extends FSBoundedParam[Double](

name = "health\_signal\_store\_pnsfw\_tweet\_media\_threshold",

default = 1.0,

min = 0.0,

max = 1.0

)

/\*\*

\* Threshold score for filtering tweets with images based on pnsfw\_media\_tweet Model.

\*/

object PnsfwTweetImageThreshold

extends FSBoundedParam[Double](

name = "health\_signal\_store\_pnsfw\_tweet\_image\_threshold",

default = 1.0,

min = 0.0,

max = 1.0

)

/\*\*

\* Threshold score for filtering quote/reply tweets based on source tweet's media

\*/

object PnsfwQuoteTweetThreshold

extends FSBoundedParam[Double](

name = "health\_signal\_store\_pnsfw\_quote\_tweet\_threshold",

default = 1.0,

min = 0.0,

max = 1.0

)

/\*\*

\* Threshold score for bucketing based on pnsfw\_media\_tweet Model.

\*/

object PnsfwTweetMediaBucketingThreshold

extends FSBoundedParam[Double](

name = "health\_signal\_store\_pnsfw\_tweet\_media\_bucketing\_threshold",

default = 1.0,

min = 0.0,

max = 1.0

)

/\*\*

\* Param to enable filtering using multilingual psnfw predicate

\*/

object EnableHealthSignalStoreMultilingualPnsfwTweetTextPredicate

extends FSParam[Boolean](

name = "health\_signal\_store\_enable\_multilingual\_pnsfw\_tweet\_text\_predicate",

default = false

)

/\*\*

\* Language sequence we will query pnsfw scores for

\*/

object MultilingualPnsfwTweetTextSupportedLanguages

extends FSParam[Seq[String]](

name = "health\_signal\_store\_multilingual\_pnsfw\_tweet\_supported\_languages",

default = Seq.empty[String],

)

/\*\*

\* Threshold score per language for bucketing based on pnsfw scores.

\*/

object MultilingualPnsfwTweetTextBucketingThreshold

extends FSParam[Seq[Double]](

name = "health\_signal\_store\_multilingual\_pnsfw\_tweet\_text\_bucketing\_thresholds",

default = Seq.empty[Double],

)

/\*\*

\* Threshold score per language for filtering based on pnsfw scores.

\*/

object MultilingualPnsfwTweetTextFilteringThreshold

extends FSParam[Seq[Double]](

name = "health\_signal\_store\_multilingual\_pnsfw\_tweet\_text\_filtering\_thresholds",

default = Seq.empty[Double],

)

/\*\*

\* List of models to threshold scores for bucketing purposes

\*/

object MultilingualPnsfwTweetTextBucketingModelList

extends FSEnumSeqParam[NsfwTextDetectionModel.type](

name = "health\_signal\_store\_multilingual\_pnsfw\_tweet\_text\_bucketing\_models\_ids",

default = Seq(NsfwTextDetectionModel.ProdModel),

enum = NsfwTextDetectionModel

)

object MultilingualPnsfwTweetTextModel

extends FSEnumParam[NsfwTextDetectionModel.type](

name = "health\_signal\_store\_multilingual\_pnsfw\_tweet\_text\_model",

default = NsfwTextDetectionModel.ProdModel,

enum = NsfwTextDetectionModel

)

/\*\*

\* Param to determine media should be enabled for android

\*/

object EnableEventSquareMediaAndroid

extends FSParam[Boolean](

name = "mr\_enable\_event\_media\_square\_android",

default = false

)

/\*\*

\* Param to determine expanded media should be enabled for android

\*/

object EnableEventPrimaryMediaAndroid

extends FSParam[Boolean](

name = "mr\_enable\_event\_media\_primary\_android",

default = false

)

/\*\*

\* Param to determine media should be enabled for ios for MagicFanout

\*/

object EnableEventSquareMediaIosMagicFanoutNewsEvent

extends FSParam[Boolean](

name = "mr\_enable\_event\_media\_square\_ios\_mf",

default = false

)

/\*\*

\* Param to configure HTL Visit fatigue

\*/

object HTLVisitFatigueTime

extends FSBoundedParam[Int](

name = "frigate\_push\_htl\_visit\_fatigue\_time",

default = 20,

min = 0,

max = 72) {

// Fatigue duration for HTL visit

final val DefaultHoursToFatigueAfterHtlVisit = 20

final val OldHoursToFatigueAfterHtlVisit = 8

}

object MagicFanoutNewsUserGeneratedEventsEnable

extends FSParam[Boolean](

name = "magicfanout\_news\_user\_generated\_events\_enable",

default = false)

object MagicFanoutSkipAccountCountryPredicate

extends FSParam[Boolean]("magicfanout\_news\_skip\_account\_country\_predicate", false)

object MagicFanoutNewsEnableDescriptionCopy

extends FSParam[Boolean](name = "magicfanout\_news\_enable\_description\_copy", default = false)

/\*\*

\* Enables Custom Targeting for MagicFnaout News events in Pushservice

\*/

object MagicFanoutEnableCustomTargetingNewsEvent

extends FSParam[Boolean]("magicfanout\_news\_event\_custom\_targeting\_enable", false)

/\*\*

\* Enable Topic Copy in MF

\*/

object EnableTopicCopyForMF

extends FSParam[Boolean](

name = "magicfanout\_enable\_topic\_copy",

default = false

)

/\*\*

\* Enable Topic Copy in MF for implicit topics

\*/

object EnableTopicCopyForImplicitTopics

extends FSParam[Boolean](

name = "magicfanout\_enable\_topic\_copy\_erg\_interests",

default = false

)

/\*\*

\* Enable NewCreator push

\*/

object EnableNewCreatorPush

extends FSParam[Boolean](

name = "new\_creator\_enable\_push",

default = false

)

/\*\*

\* Enable CreatorSubscription push

\*/

object EnableCreatorSubscriptionPush

extends FSParam[Boolean](

name = "creator\_subscription\_enable\_push",

default = false

)

/\*\*

\* Featureswitch param to enable/disable push recommendations

\*/

object EnablePushRecommendationsParam

extends FSParam[Boolean](name = "push\_recommendations\_enabled", default = false)

object DisableMlInFilteringFeatureSwitchParam

extends FSParam[Boolean](

name = "frigate\_push\_modeling\_disable\_ml\_in\_filtering",

default = false

)

object EnableMinDurationModifier

extends FSParam[Boolean](

name = "min\_duration\_modifier\_enable\_hour\_modifier",

default = false

)

object EnableMinDurationModifierV2

extends FSParam[Boolean](

name = "min\_duration\_modifier\_enable\_hour\_modifier\_v2",

default = false

)

object MinDurationModifierStartHourList

extends FSParam[Seq[Int]](

name = "min\_duration\_modifier\_start\_time\_list",

default = Seq(),

)

object MinDurationModifierEndHourList

extends FSParam[Seq[Int]](

name = "min\_duration\_modifier\_start\_end\_list",

default = Seq(),

)

object MinDurationTimeModifierConst

extends FSParam[Seq[Int]](

name = "min\_duration\_modifier\_const\_list",

default = Seq(),

)

object EnableQueryUserOpenedHistory

extends FSParam[Boolean](

name = "min\_duration\_modifier\_enable\_query\_user\_opened\_history",

default = false

)

object EnableMinDurationModifierByUserHistory

extends FSParam[Boolean](

name = "min\_duration\_modifier\_enable\_hour\_modifier\_by\_user\_history",

default = false

)

object EnableRandomHourForQuickSend

extends FSParam[Boolean](

name = "min\_duration\_modifier\_enable\_random\_hour\_for\_quick\_send",

default = false

)

object SendTimeByUserHistoryMaxOpenedThreshold

extends FSBoundedParam[Int](

name = "min\_duration\_modifier\_max\_opened\_threshold",

default = 4,

min = 0,

max = 100)

object SendTimeByUserHistoryNoSendsHours

extends FSBoundedParam[Int](

name = "min\_duration\_modifier\_no\_sends\_hours",

default = 1,

min = 0,

max = 24)

object SendTimeByUserHistoryQuickSendBeforeHours

extends FSBoundedParam[Int](

name = "min\_duration\_modifier\_quick\_send\_before\_hours",

default = 0,

min = 0,

max = 24)

object SendTimeByUserHistoryQuickSendAfterHours

extends FSBoundedParam[Int](

name = "min\_duration\_modifier\_quick\_send\_after\_hours",

default = 0,

min = 0,

max = 24)

object SendTimeByUserHistoryQuickSendMinDurationInMinute

extends FSBoundedParam[Int](

name = "min\_duration\_modifier\_quick\_send\_min\_duration",

default = 0,

min = 0,

max = 1440)

object SendTimeByUserHistoryNoSendMinDuration

extends FSBoundedParam[Int](

name = "min\_duration\_modifier\_no\_send\_min\_duration",

default = 24,

min = 0,

max = 24)

object EnableMfGeoTargeting

extends FSParam[Boolean](

name = "frigate\_push\_magicfanout\_geo\_targeting\_enable",

default = false)

/\*\*

\* Enable RUX Tweet landing page for push open. When this param is enabled, user will go to RUX

\* landing page instead of Tweet details page when opening MagicRecs push.

\*/

object EnableRuxLandingPage

extends FSParam[Boolean](name = "frigate\_push\_enable\_rux\_landing\_page", default = false)

/\*\*

\* Enable RUX Tweet landing page for Ntab Click. When this param is enabled, user will go to RUX

\* landing page instead of Tweet details page when click MagicRecs entry on Ntab.

\*/

object EnableNTabRuxLandingPage

extends FSParam[Boolean](name = "frigate\_push\_enable\_ntab\_rux\_landing\_page", default = false)

/\*\*

\* Param to enable Onboarding Pushes

\*/

object EnableOnboardingPushes

extends FSParam[Boolean](

name = "onboarding\_push\_enable",

default = false

)

/\*\*

\* Param to enable Address Book Pushes

\*/

object EnableAddressBookPush

extends FSParam[Boolean](

name = "onboarding\_push\_enable\_address\_book\_push",

default = false

)

/\*\*

\* Param to enable Complete Onboarding Pushes

\*/

object EnableCompleteOnboardingPush

extends FSParam[Boolean](

name = "onboarding\_push\_enable\_complete\_onboarding\_push",

default = false

)

/\*\*

\* Param to enable Smart Push Config for MR Override Notifs on Android

\*/

object EnableOverrideNotificationsSmartPushConfigForAndroid

extends FSParam[Boolean](

name = "mr\_override\_enable\_smart\_push\_config\_for\_android",

default = false)

/\*\*

\* Param to control the min duration since last MR push for Onboarding Pushes

\*/

object MrMinDurationSincePushForOnboardingPushes

extends FSBoundedParam[Duration](

name = "onboarding\_push\_min\_duration\_since\_push\_days",

default = 4.days,

min = Duration.Bottom,

max = Duration.Top)

with HasDurationConversion {

override val durationConversion = DurationConversion.FromDays

}

/\*\*

\* Param to control the push fatigue for Onboarding Pushes

\*/

object FatigueForOnboardingPushes

extends FSBoundedParam[Duration](

name = "onboarding\_push\_fatigue\_days",

default = 30.days,

min = Duration.Bottom,

max = Duration.Top)

with HasDurationConversion {

override val durationConversion = DurationConversion.FromDays

}

/\*\*

\* Param to specify the maximum number of Onboarding Push Notifs in a specified period of time

\*/

object MaxOnboardingPushInInterval

extends FSBoundedParam[Int](

name = "onboarding\_push\_max\_in\_interval",

default = 1,

min = 0,

max = 10

)

/\*\*

\* Param to disable the Onboarding Push Notif Fatigue

\*/

object DisableOnboardingPushFatigue

extends FSParam[Boolean](

name = "onboarding\_push\_disable\_push\_fatigue",

default = false

)

/\*\*

\* Param to control the inverter for fatigue between consecutive TopTweetsByGeoPush

\*/

object TopTweetsByGeoPushInterval

extends FSBoundedParam[Duration](

name = "top\_tweets\_by\_geo\_interval\_days",

default = 0.days,

min = Duration.Bottom,

max = Duration.Top)

with HasDurationConversion {

override val durationConversion = DurationConversion.FromDays

}

/\*\*

\* Param to control the inverter for fatigue between consecutive TripTweets

\*/

object HighQualityTweetsPushInterval

extends FSBoundedParam[Duration](

name = "high\_quality\_candidates\_push\_interval\_days",

default = 1.days,

min = Duration.Bottom,

max = Duration.Top)

with HasDurationConversion {

override val durationConversion = DurationConversion.FromDays

}

/\*\*

\* Expiry TTL duration for Tweet Notification types written to history store

\*/

object FrigateHistoryTweetNotificationWriteTtl

extends FSBoundedParam[Duration](

name = "frigate\_notification\_history\_tweet\_write\_ttl\_days",

default = 60.days,

min = Duration.Bottom,

max = Duration.Top

)

with HasDurationConversion {

override val durationConversion = DurationConversion.FromDays

}

/\*\*

\* Expiry TTL duration for Notification written to history store

\*/

object FrigateHistoryOtherNotificationWriteTtl

extends FSBoundedParam[Duration](

name = "frigate\_notification\_history\_other\_write\_ttl\_days",

default = 90.days,

min = Duration.Bottom,

max = Duration.Top)

with HasDurationConversion {

override val durationConversion = DurationConversion.FromDays

}

/\*\*

\* Param to control maximum number of TopTweetsByGeoPush pushes to receive in an interval

\*/

object MaxTopTweetsByGeoPushGivenInterval

extends FSBoundedParam[Int](

name = "top\_tweets\_by\_geo\_push\_given\_interval",

default = 1,

min = 0,

max = 10

)

/\*\*

\* Param to control maximum number of HighQualityTweet pushes to receive in an interval

\*/

object MaxHighQualityTweetsPushGivenInterval

extends FSBoundedParam[Int](

name = "high\_quality\_candidates\_max\_push\_given\_interval",

default = 3,

min = 0,

max = 10

)

/\*\*

\* Param to downrank/backfill top tweets by geo candidates

\*/

object BackfillRankTopTweetsByGeoCandidates

extends FSParam[Boolean](

name = "top\_tweets\_by\_geo\_backfill\_rank",

default = false

)

/\*\*

\* Determine whether to use aggressive thresholds for Health filtering on SearchTweet

\*/

object PopGeoTweetEnableAggressiveThresholds

extends FSParam[Boolean](

name = "top\_tweets\_by\_geo\_enable\_aggressive\_health\_thresholds",

default = false

)

/\*\*

\* Param to apply different scoring functions to select top tweets by geo candidates

\*/

object ScoringFuncForTopTweetsByGeo

extends FSParam[String](

name = "top\_tweets\_by\_geo\_scoring\_function",

default = "Pop8H",

)

/\*\*

\* Param to query different stores in pop geo service.

\*/

object TopTweetsByGeoCombinationParam

extends FSEnumParam[TopTweetsForGeoCombination.type](

name = "top\_tweets\_by\_geo\_combination\_id",

default = TopTweetsForGeoCombination.Default,

enum = TopTweetsForGeoCombination

)

/\*\*

\* Param for popgeo tweet version

\*/

object PopGeoTweetVersionParam

extends FSEnumParam[PopGeoTweetVersion.type](

name = "top\_tweets\_by\_geo\_version\_id",

default = PopGeoTweetVersion.Prod,

enum = PopGeoTweetVersion

)

/\*\*

\* Param to query what length of hash for geoh store

\*/

object GeoHashLengthList

extends FSParam[Seq[Int]](

name = "top\_tweets\_by\_geo\_hash\_length\_list",

default = Seq(4),

)

/\*\*

\* Param to include country code results as back off .

\*/

object EnableCountryCodeBackoffTopTweetsByGeo

extends FSParam[Boolean](

name = "top\_tweets\_by\_geo\_enable\_country\_code\_backoff",

default = false,

)

/\*\*

\* Param to decide ranking function for fetched top tweets by geo

\*/

object RankingFunctionForTopTweetsByGeo

extends FSEnumParam[TopTweetsForGeoRankingFunction.type](

name = "top\_tweets\_by\_geo\_ranking\_function\_id",

default = TopTweetsForGeoRankingFunction.Score,

enum = TopTweetsForGeoRankingFunction

)

/\*\*

\* Param to enable top tweets by geo candidates

\*/

object EnableTopTweetsByGeoCandidates

extends FSParam[Boolean](

name = "top\_tweets\_by\_geo\_enable\_candidate\_source",

default = false

)

/\*\*

\* Param to enable top tweets by geo candidates for dormant users

\*/

object EnableTopTweetsByGeoCandidatesForDormantUsers

extends FSParam[Boolean](

name = "top\_tweets\_by\_geo\_enable\_candidate\_source\_dormant\_users",

default = false

)

/\*\*

\* Param to specify the maximum number of Top Tweets by Geo candidates to take

\*/

object MaxTopTweetsByGeoCandidatesToTake

extends FSBoundedParam[Int](

name = "top\_tweets\_by\_geo\_candidates\_to\_take",

default = 10,

min = 0,

max = 100

)

/\*\*

\* Param to min duration since last MR push for top tweets by geo pushes

\*/

object MrMinDurationSincePushForTopTweetsByGeoPushes

extends FSBoundedParam[Duration](

name = "top\_tweets\_by\_geo\_min\_duration\_since\_last\_mr\_days",

default = 3.days,

min = Duration.Bottom,

max = Duration.Top)

with HasDurationConversion {

override val durationConversion = DurationConversion.FromDays

}

/\*\*

\* Param to enable FRS candidate tweets

\*/

object EnableFrsCandidates

extends FSParam[Boolean](

name = "frs\_tweet\_candidate\_enable\_adaptor",

default = false

)

/\*\*

\* Param to enable FRSTweet candidates for topic setting users

\* \*/

object EnableFrsTweetCandidatesTopicSetting

extends FSParam[Boolean](

name = "frs\_tweet\_candidate\_enable\_adaptor\_for\_topic\_setting",

default = false

)

/\*\*

\* Param to enable topic annotations for FRSTweet candidates tweets

\* \*/

object EnableFrsTweetCandidatesTopicAnnotation

extends FSParam[Boolean](

name = "frs\_tweet\_candidate\_enable\_topic\_annotation",

default = false

)

/\*\*

\* Param to enable topic copy for FRSTweet candidates tweets

\* \*/

object EnableFrsTweetCandidatesTopicCopy

extends FSParam[Boolean](

name = "frs\_tweet\_candidate\_enable\_topic\_copy",

default = false

)

/\*\*

\* Topic score threshold for FRSTweet candidates topic annotations

\* \*/

object FrsTweetCandidatesTopicScoreThreshold

extends FSBoundedParam[Double](

name = "frs\_tweet\_candidate\_topic\_score\_threshold",

default = 0.0,

min = 0.0,

max = 100.0

)

/\*\*

\* Param to enable mr modeling-based candidates tweets

\* \*/

object EnableMrModelingBasedCandidates

extends FSParam[Boolean](

name = "candidate\_generation\_model\_enable\_adaptor",

default = false

)

/\*\*

Param to enable mr modeling-based candidates tweets for topic setting users

\* \*/

object EnableMrModelingBasedCandidatesTopicSetting

extends FSParam[Boolean](

name = "candidate\_generation\_model\_enable\_adaptor\_for\_topic\_setting",

default = false

)

/\*\*

\* Param to enable topic annotations for mr modeling-based candidates tweets

\* \*/

object EnableMrModelingBasedCandidatesTopicAnnotation

extends FSParam[Boolean](

name = "candidate\_generation\_model\_enable\_adaptor\_topic\_annotation",

default = false

)

/\*\*

\* Topic score threshold for mr modeling based candidates topic annotations

\* \*/

object MrModelingBasedCandidatesTopicScoreThreshold

extends FSBoundedParam[Double](

name = "candidate\_generation\_model\_topic\_score\_threshold",

default = 0.0,

min = 0.0,

max = 100.0

)

/\*\*

\* Param to enable topic copy for mr modeling-based candidates tweets

\* \*/

object EnableMrModelingBasedCandidatesTopicCopy

extends FSParam[Boolean](

name = "candidate\_generation\_model\_enable\_topic\_copy",

default = false

)

/\*\*

\* Number of max mr modeling based candidates

\* \*/

object NumberOfMaxMrModelingBasedCandidates

extends FSBoundedParam[Int](

name = "candidate\_generation\_model\_max\_mr\_modeling\_based\_candidates",

default = 200,

min = 0,

max = 1000

)

/\*\*

\* Enable the traffic to use fav threshold

\* \*/

object EnableThresholdOfFavMrModelingBasedCandidates

extends FSParam[Boolean](

name = "candidate\_generation\_model\_enable\_fav\_threshold",

default = false

)

/\*\*

\* Threshold of fav for mr modeling based candidates

\* \*/

object ThresholdOfFavMrModelingBasedCandidates

extends FSBoundedParam[Int](

name = "candidate\_generation\_model\_fav\_threshold",

default = 0,

min = 0,

max = 500

)

/\*\*

\* Filtered threshold for mr modeling based candidates

\* \*/

object CandidateGenerationModelCosineThreshold

extends FSBoundedParam[Double](

name = "candidate\_generation\_model\_cosine\_threshold",

default = 0.9,

min = 0.0,

max = 1.0

)

/\*

\* ANN hyparameters

\* \*/

object ANNEfQuery

extends FSBoundedParam[Int](

name = "candidate\_generation\_model\_ann\_ef\_query",

default = 300,

min = 50,

max = 1500

)

/\*\*

\* Param to do real A/B impression for FRS candidates to avoid dilution

\*/

object EnableResultFromFrsCandidates

extends FSParam[Boolean](

name = "frs\_tweet\_candidate\_enable\_returned\_result",

default = false

)

/\*\*

\* Param to enable hashspace candidate tweets

\*/

object EnableHashspaceCandidates

extends FSParam[Boolean](

name = "hashspace\_candidate\_enable\_adaptor",

default = false

)

/\*\*

\* Param to enable hashspace candidates tweets for topic setting users

\* \*/

object EnableHashspaceCandidatesTopicSetting

extends FSParam[Boolean](

name = "hashspace\_candidate\_enable\_adaptor\_for\_topic\_setting",

default = false

)

/\*\*

\* Param to enable topic annotations for hashspace candidates tweets

\* \*/

object EnableHashspaceCandidatesTopicAnnotation

extends FSParam[Boolean](

name = "hashspace\_candidate\_enable\_topic\_annotation",

default = false

)

/\*\*

\* Param to enable topic copy for hashspace candidates tweets

\* \*/

object EnableHashspaceCandidatesTopicCopy

extends FSParam[Boolean](

name = "hashspace\_candidate\_enable\_topic\_copy",

default = false

)

/\*\*

\* Topic score threshold for hashspace candidates topic annotations

\* \*/

object HashspaceCandidatesTopicScoreThreshold

extends FSBoundedParam[Double](

name = "hashspace\_candidate\_topic\_score\_threshold",

default = 0.0,

min = 0.0,

max = 100.0

)

/\*\*

\* Param to do real A/B impression for hashspace candidates to avoid dilution

\*/

object EnableResultFromHashspaceCandidates

extends FSParam[Boolean](

name = "hashspace\_candidate\_enable\_returned\_result",

default = false

)

/\*\*

\* Param to enable detopic tweet candidates in adaptor

\*/

object EnableDeTopicTweetCandidates

extends FSParam[Boolean](

name = "detopic\_tweet\_candidate\_enable\_adaptor",

default = false

)

/\*\*

\* Param to enable detopic tweet candidates results (to avoid dilution)

\*/

object EnableDeTopicTweetCandidateResults

extends FSParam[Boolean](

name = "detopic\_tweet\_candidate\_enable\_results",

default = false

)

/\*\*

\* Param to specify whether to provide a custom list of topics in request

\*/

object EnableDeTopicTweetCandidatesCustomTopics

extends FSParam[Boolean](

name = "detopic\_tweet\_candidate\_enable\_custom\_topics",

default = false

)

/\*\*

\* Param to specify whether to provide a custom language in request

\*/

object EnableDeTopicTweetCandidatesCustomLanguages

extends FSParam[Boolean](

name = "detopic\_tweet\_candidate\_enable\_custom\_languages",

default = false

)

/\*\*

\* Number of detopic tweet candidates in the request

\* \*/

object NumberOfDeTopicTweetCandidates

extends FSBoundedParam[Int](

name = "detopic\_tweet\_candidate\_num\_candidates\_in\_request",

default = 600,

min = 0,

max = 3000

)

/\*\*

\* Max Number of detopic tweet candidates returned in adaptor

\* \*/

object NumberOfMaxDeTopicTweetCandidatesReturned

extends FSBoundedParam[Int](

name = "detopic\_tweet\_candidate\_max\_num\_candidates\_returned",

default = 200,

min = 0,

max = 3000

)

/\*\*

\* Param to enable F1 from protected Authors

\*/

object EnableF1FromProtectedTweetAuthors

extends FSParam[Boolean](

"f1\_enable\_protected\_tweets",

false

)

/\*\*

\* Param to enable safe user tweet tweetypie store

\*/

object EnableSafeUserTweetTweetypieStore

extends FSParam[Boolean](

"mr\_infra\_enable\_use\_safe\_user\_tweet\_tweetypie",

false

)

/\*\*

\* Param to min duration since last MR push for top tweets by geo pushes

\*/

object EnableMrMinDurationSinceMrPushFatigue

extends FSParam[Boolean](

name = "top\_tweets\_by\_geo\_enable\_min\_duration\_since\_mr\_fatigue",

default = false

)

/\*\*

\* Param to check time since last time user logged in for geo top tweets by geo push

\*/

object TimeSinceLastLoginForGeoPopTweetPush

extends FSBoundedParam[Duration](

name = "top\_tweets\_by\_geo\_time\_since\_last\_login\_in\_days",

default = 14.days,

min = Duration.Bottom,

max = Duration.Top)

with HasDurationConversion {

override val durationConversion = DurationConversion.FromDays

}

/\*\*

\* Param to check time since last time user logged in for geo top tweets by geo push

\*/

object MinimumTimeSinceLastLoginForGeoPopTweetPush

extends FSBoundedParam[Duration](

name = "top\_tweets\_by\_geo\_minimum\_time\_since\_last\_login\_in\_days",

default = 14.days,

min = Duration.Bottom,

max = Duration.Top)

with HasDurationConversion {

override val durationConversion = DurationConversion.FromDays

}

/\*\* How long we wait after a user visited the app before sending them a space fanout rec \*/

object SpaceRecsAppFatigueDuration

extends FSBoundedParam[Duration](

name = "space\_recs\_app\_fatigue\_duration\_hours",

default = 4.hours,

min = Duration.Bottom,

max = Duration.Top)

with HasDurationConversion {

override val durationConversion = DurationConversion.FromHours

}

/\*\* The fatigue time-window for OON space fanout recs, e.g. 1 push every 3 days \*/

object OONSpaceRecsFatigueDuration

extends FSBoundedParam[Duration](

name = "space\_recs\_oon\_fatigue\_duration\_days",

default = 1.days,

min = Duration.Bottom,

max = Duration.Top)

with HasDurationConversion {

override val durationConversion = DurationConversion.FromDays

}

/\*\* The global fatigue time-window for space fanout recs, e.g. 1 push every 3 days \*/

object SpaceRecsGlobalFatigueDuration

extends FSBoundedParam[Duration](

name = "space\_recs\_global\_fatigue\_duration\_days",

default = 1.day,

min = Duration.Bottom,

max = Duration.Top)

with HasDurationConversion {

override val durationConversion = DurationConversion.FromDays

}

/\*\* The min-interval between space fanout recs.

\* After receiving a space fanout rec, they must wait a minimum of this

\* interval before eligibile for another \*/

object SpaceRecsFatigueMinIntervalDuration

extends FSBoundedParam[Duration](

name = "space\_recs\_fatigue\_mininterval\_duration\_minutes",

default = 30.minutes,

min = Duration.Bottom,

max = Duration.Top)

with HasDurationConversion {

override val durationConversion = DurationConversion.FromMinutes

}

/\*\* Space fanout user-follow rank threshold.

\* Users targeted by a follow that is above this threshold will be filtered \*/

object SpaceRecsRealgraphThreshold

extends FSBoundedParam[Int](

name = "space\_recs\_realgraph\_threshold",

default = 50,

max = 500,

min = 0

)

object EnableHydratingRealGraphTargetUserFeatures

extends FSParam[Boolean](

name = "frigate\_push\_modeling\_enable\_hydrating\_real\_graph\_target\_user\_feature",

default = true

)

/\*\* Param to reduce dillution when checking if a space is featured or not \*/

object CheckFeaturedSpaceOON

extends FSParam[Boolean](name = "space\_recs\_check\_if\_its\_featured\_space", default = false)

/\*\* Enable Featured Spaces Rules for OON spaces \*/

object EnableFeaturedSpacesOON

extends FSParam[Boolean](name = "space\_recs\_enable\_featured\_spaces\_oon", default = false)

/\*\* Enable Geo Targeting \*/

object EnableGeoTargetingForSpaces

extends FSParam[Boolean](name = "space\_recs\_enable\_geo\_targeting", default = false)

/\*\* Number of max pushes within the fatigue duration for OON Space Recs \*/

object OONSpaceRecsPushLimit

extends FSBoundedParam[Int](

name = "space\_recs\_oon\_push\_limit",

default = 1,

max = 3,

min = 0

)

/\*\* Space fanout recs, number of max pushes within the fatigue duration \*/

object SpaceRecsGlobalPushLimit

extends FSBoundedParam[Int](

name = "space\_recs\_global\_push\_limit",

default = 3,

max = 50,

min = 0

)

/\*\*

\* Param to enable score based override.

\*/

object EnableOverrideNotificationsScoreBasedOverride

extends FSParam[Boolean](

name = "mr\_override\_enable\_score\_ranking",

default = false

)

/\*\*

\* Param to determine the lookback duration when searching for override info.

\*/

object OverrideNotificationsLookbackDurationForOverrideInfo

extends FSBoundedParam[Duration](

name = "mr\_override\_lookback\_duration\_override\_info\_in\_days",

default = 30.days,

min = Duration.Bottom,

max = Duration.Top)

with HasDurationConversion {

override val durationConversion = DurationConversion.FromDays

}

/\*\*

\* Param to determine the lookback duration when searching for impression ids.

\*/

object OverrideNotificationsLookbackDurationForImpressionId

extends FSBoundedParam[Duration](

name = "mr\_override\_lookback\_duration\_impression\_id\_in\_days",

default = 30.days,

min = Duration.Bottom,

max = Duration.Top)

with HasDurationConversion {

override val durationConversion = DurationConversion.FromDays

}

/\*\*

\* Param to enable sending multiple target ids in the payload.

\*/

object EnableOverrideNotificationsMultipleTargetIds

extends FSParam[Boolean](

name = "mr\_override\_enable\_multiple\_target\_ids",

default = false

)

/\*\*

\* Param for MR Web Notifications holdback

\*/

object MRWebHoldbackParam

extends FSParam[Boolean](

name = "mr\_web\_notifications\_holdback",

default = false

)

object CommonRecommendationTypeDenyListPushHoldbacks

extends FSParam[Seq[String]](

name = "crt\_to\_exclude\_from\_holdbacks\_push\_holdbacks",

default = Seq.empty[String]

)

/\*\*

\* Param to enable sending number of slots to maintain in the payload.

\*/

object EnableOverrideNotificationsNSlots

extends FSParam[Boolean](

name = "mr\_override\_enable\_n\_slots",

default = false

)

/\*\*

\* Enable down ranking of NUPS and pop geo topic follow candidates for new user playbook.

\*/

object EnableDownRankOfNewUserPlaybookTopicFollowPush

extends FSParam[Boolean](

name = "topic\_follow\_new\_user\_playbook\_enable\_down\_rank",

default = false

)

/\*\*

\* Enable down ranking of NUPS and pop geo topic tweet candidates for new user playbook.

\*/

object EnableDownRankOfNewUserPlaybookTopicTweetPush

extends FSParam[Boolean](

name = "topic\_tweet\_new\_user\_playbook\_enable\_down\_rank",

default = false

)

/\*\*

\* Param to enable/disable employee only spaces for fanout of notifications

\*/

object EnableEmployeeOnlySpaceNotifications

extends FSParam[Boolean](name = "space\_recs\_employee\_only\_enable", default = false)

/\*\*

\* NTab spaces ttl experiments

\*/

object EnableSpacesTtlForNtab

extends FSParam[Boolean](

name = "ntab\_spaces\_ttl\_enable",

default = false

)

/\*\*

\* Param to determine the ttl duration for space notifications on NTab.

\*/

object SpaceNotificationsTTLDurationForNTab

extends FSBoundedParam[Duration](

name = "ntab\_spaces\_ttl\_hours",

default = 1.hour,

min = Duration.Bottom,

max = Duration.Top)

with HasDurationConversion {

override val durationConversion = DurationConversion.FromHours

}

/\*

\* NTab override experiments

\* see go/ntab-override experiment brief for more details

\*/

/\*\*

\* Override notifications for Spaces on lockscreen.

\*/

object EnableOverrideForSpaces

extends FSParam[Boolean](

name = "mr\_override\_spaces",

default = false

)

/\*\*

\* Param to enable storing the Generic Notification Key.

\*/

object EnableStoringNtabGenericNotifKey

extends FSParam[Boolean](

name = "ntab\_enable\_storing\_generic\_notif\_key",

default = false

)

/\*\*

\* Param to enable deleting the Target's timeline.

\*/

object EnableDeletingNtabTimeline

extends FSParam[Boolean](

name = "ntab\_enable\_delete\_timeline",

default = false

)

/\*\*

\* Param to enable sending the overrideId

\* to NTab which enables override support in NTab-api

\*/

object EnableOverrideIdNTabRequest

extends FSParam[Boolean](

name = "ntab\_enable\_override\_id\_in\_request",

default = false

)

/\*\*

\* [Override Workstream] Param to enable NTab override n-slot feature.

\*/

object EnableNslotsForOverrideOnNtab

extends FSParam[Boolean](

name = "ntab\_enable\_override\_max\_count",

default = false

)

/\*\*

\* Param to determine the lookback duration for override candidates on NTab.

\*/

object OverrideNotificationsLookbackDurationForNTab

extends FSBoundedParam[Duration](

name = "ntab\_override\_lookback\_duration\_days",

default = 30.days,

min = Duration.Bottom,

max = Duration.Top)

with HasDurationConversion {

override val durationConversion = DurationConversion.FromDays

}

/\*\*

\* Param to determine the max count for candidates on NTab.

\*/

object OverrideNotificationsMaxCountForNTab

extends FSBoundedParam[Int](

name = "ntab\_override\_limit",

min = 0,

max = Int.MaxValue,

default = 4)

//// end override experiments ////

/\*\*

\* Param to enable top tweet impressions notification

\*/

object EnableTopTweetImpressionsNotification

extends FSParam[Boolean](

name = "top\_tweet\_impressions\_notification\_enable",

default = false

)

/\*\*

\* Param to control the inverter for fatigue between consecutive TweetImpressions

\*/

object TopTweetImpressionsNotificationInterval

extends FSBoundedParam[Duration](

name = "top\_tweet\_impressions\_notification\_interval\_days",

default = 7.days,

min = Duration.Bottom,

max = Duration.Top)

with HasDurationConversion {

override val durationConversion = DurationConversion.FromDays

}

/\*\*

\* The min-interval between TweetImpressions notifications.

\* After receiving a TweetImpressions notif, they must wait a minimum of this

\* interval before being eligible for another

\*/

object TopTweetImpressionsFatigueMinIntervalDuration

extends FSBoundedParam[Duration](

name = "top\_tweet\_impressions\_fatigue\_mininterval\_duration\_days",

default = 1.days,

min = Duration.Bottom,

max = Duration.Top)

with HasDurationConversion {

override val durationConversion = DurationConversion.FromDays

}

/\*\*

\* Maximum number of top tweet impressions notifications to receive in an interval

\*/

object MaxTopTweetImpressionsNotifications

extends FSBoundedParam(

name = "top\_tweet\_impressions\_fatigue\_max\_in\_interval",

default = 0,

min = 0,

max = 10

)

/\*\*

\* Param for min number of impressions counts to be eligible for lonely\_birds\_tweet\_impressions model

\*/

object TopTweetImpressionsMinRequired

extends FSBoundedParam[Int](

name = "top\_tweet\_impressions\_min\_required",

default = 25,

min = 0,

max = Int.MaxValue

)

/\*\*

\* Param for threshold of impressions counts to notify for lonely\_birds\_tweet\_impressions model

\*/

object TopTweetImpressionsThreshold

extends FSBoundedParam[Int](

name = "top\_tweet\_impressions\_threshold",

default = 25,

min = 0,

max = Int.MaxValue

)

/\*\*

\* Param for the number of days to search up to for a user's original tweets

\*/

object TopTweetImpressionsOriginalTweetsNumDaysSearch

extends FSBoundedParam[Int](

name = "top\_tweet\_impressions\_original\_tweets\_num\_days\_search",

default = 3,

min = 0,

max = 21

)

/\*\*

\* Param for the minimum number of original tweets a user needs to be considered an original author

\*/

object TopTweetImpressionsMinNumOriginalTweets

extends FSBoundedParam[Int](

name = "top\_tweet\_impressions\_num\_original\_tweets",

default = 3,

min = 0,

max = Int.MaxValue

)

/\*\*

\* Param for the max number of favorites any original Tweet can have

\*/

object TopTweetImpressionsMaxFavoritesPerTweet

extends FSBoundedParam[Int](

name = "top\_tweet\_impressions\_max\_favorites\_per\_tweet",

default = 3,

min = 0,

max = Int.MaxValue

)

/\*\*

\* Param for the max number of total inbound favorites for a user's tweets

\*/

object TopTweetImpressionsTotalInboundFavoritesLimit

extends FSBoundedParam[Int](

name = "top\_tweet\_impressions\_total\_inbound\_favorites\_limit",

default = 60,

min = 0,

max = Int.MaxValue

)

/\*\*

\* Param for the number of days to search for tweets to count the total inbound favorites

\*/

object TopTweetImpressionsTotalFavoritesLimitNumDaysSearch

extends FSBoundedParam[Int](

name = "top\_tweet\_impressions\_total\_favorites\_limit\_num\_days\_search",

default = 7,

min = 0,

max = 21

)

/\*\*

\* Param for the max number of recent tweets Tflock should return

\*/

object TopTweetImpressionsRecentTweetsByAuthorStoreMaxResults

extends FSBoundedParam[Int](

name = "top\_tweet\_impressions\_recent\_tweets\_by\_author\_store\_max\_results",

default = 50,

min = 0,

max = 1000

)

/\*

\* Param to represent the max number of slots to maintain for Override Notifications

\*/

object OverrideNotificationsMaxNumOfSlots

extends FSBoundedParam[Int](

name = "mr\_override\_max\_num\_slots",

default = 1,

max = 10,

min = 1

)

object EnableOverrideMaxSlotFn

extends FSParam[Boolean](

name = "mr\_override\_enable\_max\_num\_slots\_fn",

default = false

)

object OverrideMaxSlotFnPushCapKnobs

extends FSParam[Seq[Double]]("mr\_override\_fn\_pushcap\_knobs", default = Seq.empty[Double])

object OverrideMaxSlotFnNSlotKnobs

extends FSParam[Seq[Double]]("mr\_override\_fn\_nslot\_knobs", default = Seq.empty[Double])

object OverrideMaxSlotFnPowerKnobs

extends FSParam[Seq[Double]]("mr\_override\_fn\_power\_knobs", default = Seq.empty[Double])

object OverrideMaxSlotFnWeight

extends FSBoundedParam[Double](

"mr\_override\_fn\_weight",

default = 1.0,

min = 0.0,

max = Double.MaxValue)

/\*\*

\* Use to enable sending target ids in the Smart Push Payload

\*/

object EnableTargetIdsInSmartPushPayload

extends FSParam[Boolean](name = "mr\_override\_enable\_target\_ids", default = true)

/\*\*

\* Param to enable override by target id for MagicFanoutSportsEvent candidates

\*/

object EnableTargetIdInSmartPushPayloadForMagicFanoutSportsEvent

extends FSParam[Boolean](

name = "mr\_override\_enable\_target\_id\_for\_magic\_fanout\_sports\_event",

default = true)

/\*\*

\* Param to enable secondary account predicate on MF NFY

\*/

object EnableSecondaryAccountPredicateMF

extends FSParam[Boolean](

name = "frigate\_push\_magicfanout\_secondary\_account\_predicate",

default = false

)

/\*\*

\* Enables showing our customers videos on their notifications

\*/

object EnableInlineVideo

extends FSParam[Boolean](name = "mr\_inline\_enable\_inline\_video", default = false)

/\*\*

\* Enables autoplay for inline videos

\*/

object EnableAutoplayForInlineVideo

extends FSParam[Boolean](name = "mr\_inline\_enable\_autoplay\_for\_inline\_video", default = false)

/\*\*

\* Enable OON filtering based on MentionFilter.

\*/

object EnableOONFilteringBasedOnUserSettings

extends FSParam[Boolean](name = "oon\_filtering\_enable\_based\_on\_user\_settings", false)

/\*\*

\* Enables Custom Thread Ids which is used to ungroup notifications for N-slots on iOS

\*/

object EnableCustomThreadIdForOverride

extends FSParam[Boolean](name = "mr\_override\_enable\_custom\_thread\_id", default = false)

/\*\*

\* Enables showing verified symbol in the push presentation

\*/

object EnablePushPresentationVerifiedSymbol

extends FSParam[Boolean](name = "push\_presentation\_enable\_verified\_symbol", default = false)

/\*\*

\* Decide subtext in Android push header

\*/

object SubtextInAndroidPushHeaderParam

extends FSEnumParam[SubtextForAndroidPushHeader.type](

name = "push\_presentation\_subtext\_in\_android\_push\_header\_id",

default = SubtextForAndroidPushHeader.None,

enum = SubtextForAndroidPushHeader)

/\*\*

\* Enable SimClusters Targeting For Spaces. If false we just drop all candidates with such targeting reason

\*/

object EnableSimClusterTargetingSpaces

extends FSParam[Boolean](name = "space\_recs\_send\_simcluster\_recommendations", default = false)

/\*\*

\* Param to control threshold for dot product of simcluster based targeting on Spaces

\*/

object SpacesTargetingSimClusterDotProductThreshold

extends FSBoundedParam[Double](

"space\_recs\_simclusters\_dot\_product\_threshold",

default = 0.0,

min = 0.0,

max = 10.0)

/\*\*

\* Param to control top-k clusters simcluster based targeting on Spaces

\*/

object SpacesTopKSimClusterCount

extends FSBoundedParam[Int](

"space\_recs\_simclusters\_top\_k\_count",

default = 1,

min = 1,

max = 50)

/\*\* SimCluster users host/speaker must meet this follower count minimum threshold to be considered for sends \*/

object SpaceRecsSimClusterUserMinimumFollowerCount

extends FSBoundedParam[Int](

name = "space\_recs\_simcluster\_user\_min\_follower\_count",

default = 5000,

max = Int.MaxValue,

min = 0

)

/\*\*

\* Target has been bucketed into the Inline Action App Visit Fatigue Experiment

\*/

object TargetInInlineActionAppVisitFatigue

extends FSParam[Boolean](name = "inline\_action\_target\_in\_app\_visit\_fatigue", default = false)

/\*\*

\* Enables Inline Action App Visit Fatigue

\*/

object EnableInlineActionAppVisitFatigue

extends FSParam[Boolean](name = "inline\_action\_enable\_app\_visit\_fatigue", default = false)

/\*\*

\* Determines the fatigue that we should apply when the target user has performed an inline action

\*/

object InlineActionAppVisitFatigue

extends FSBoundedParam[Duration](

name = "inline\_action\_app\_visit\_fatigue\_hours",

default = 8.hours,

min = 1.hour,

max = 48.hours)

with HasDurationConversion {

override val durationConversion = DurationConversion.FromHours

}

/\*\*

\* Weight for reranking(oonc - weight \* nudityRate)

\*/

object AuthorSensitiveScoreWeightInReranking

extends FSBoundedParam[Double](

name = "rerank\_candidates\_author\_sensitive\_score\_weight\_in\_reranking",

default = 0.0,

min = -100.0,

max = 100.0

)

/\*\*

\* Param to control the last active space listener threshold to filter out based on that

\*/

object SpaceParticipantHistoryLastActiveThreshold

extends FSBoundedParam[Duration](

name = "space\_recs\_last\_active\_space\_listener\_threshold\_in\_hours",

default = 0.hours,

min = Duration.Bottom,

max = Duration.Top)

with HasDurationConversion {

override val durationConversion = DurationConversion.FromHours

}

/\*

\* Param to enable mr user simcluster feature set (v2020) hydration for modeling-based candidate generation

\* \*/

object HydrateMrUserSimclusterV2020InModelingBasedCG

extends FSParam[Boolean](

name = "candidate\_generation\_model\_hydrate\_mr\_user\_simcluster\_v2020",

default = false)

/\*

\* Param to enable mr semantic core feature set hydration for modeling-based candidate generation

\* \*/

object HydrateMrUserSemanticCoreInModelingBasedCG

extends FSParam[Boolean](

name = "candidate\_generation\_model\_hydrate\_mr\_user\_semantic\_core",

default = false)

/\*

\* Param to enable mr semantic core feature set hydration for modeling-based candidate generation

\* \*/

object HydrateOnboardingInModelingBasedCG

extends FSParam[Boolean](

name = "candidate\_generation\_model\_hydrate\_onboarding",

default = false)

/\*

\* Param to enable mr topic follow feature set hydration for modeling-based candidate generation

\* \*/

object HydrateTopicFollowInModelingBasedCG

extends FSParam[Boolean](

name = "candidate\_generation\_model\_hydrate\_topic\_follow",

default = false)

/\*

\* Param to enable mr user topic feature set hydration for modeling-based candidate generation

\* \*/

object HydrateMrUserTopicInModelingBasedCG

extends FSParam[Boolean](

name = "candidate\_generation\_model\_hydrate\_mr\_user\_topic",

default = false)

/\*

\* Param to enable mr user topic feature set hydration for modeling-based candidate generation

\* \*/

object HydrateMrUserAuthorInModelingBasedCG

extends FSParam[Boolean](

name = "candidate\_generation\_model\_hydrate\_mr\_user\_author",

default = false)

/\*

\* Param to enable user penguin language feature set hydration for modeling-based candidate generation

\* \*/

object HydrateUserPenguinLanguageInModelingBasedCG

extends FSParam[Boolean](

name = "candidate\_generation\_model\_hydrate\_user\_penguin\_language",

default = false)

/\*

\* Param to enable user geo feature set hydration for modeling-based candidate generation

\* \*/

object HydrateUseGeoInModelingBasedCG

extends FSParam[Boolean](

name = "candidate\_generation\_model\_hydrate\_user\_geo",

default = false)

/\*

\* Param to enable mr user hashspace embedding feature set hydration for modeling-based candidate generation

\* \*/

object HydrateMrUserHashspaceEmbeddingInModelingBasedCG

extends FSParam[Boolean](

name = "candidate\_generation\_model\_hydrate\_mr\_user\_hashspace\_embedding",

default = false)

/\*

\* Param to enable user tweet text feature hydration

\* \*/

object EnableMrUserEngagedTweetTokensFeature

extends FSParam[Boolean](

name = "feature\_hydration\_mr\_user\_engaged\_tweet\_tokens",

default = false)

/\*\*

\* Params for CRT based see less often fatigue rules

\*/

object EnableF1TriggerSeeLessOftenFatigue

extends FSParam[Boolean](

name = "seelessoften\_enable\_f1\_trigger\_fatigue",

default = false

)

object EnableNonF1TriggerSeeLessOftenFatigue

extends FSParam[Boolean](

name = "seelessoften\_enable\_nonf1\_trigger\_fatigue",

default = false

)

/\*\*

\* Adjust the NtabCaretClickFatigue for candidates if it is triggered by

\* TripHqTweet candidates

\*/

object AdjustTripHqTweetTriggeredNtabCaretClickFatigue

extends FSParam[Boolean](

name = "seelessoften\_adjust\_trip\_hq\_tweet\_triggered\_fatigue",

default = false

)

object NumberOfDaysToFilterForSeeLessOftenForF1TriggerF1

extends FSBoundedParam[Duration](

name = "seelessoften\_for\_f1\_trigger\_f1\_tofiltermr\_days",

default = 7.days,

min = Duration.Bottom,

max = Duration.Top)

with HasDurationConversion {

override val durationConversion = DurationConversion.FromDays

}

object NumberOfDaysToReducePushCapForSeeLessOftenForF1TriggerF1

extends FSBoundedParam[Duration](

name = "seelessoften\_for\_f1\_trigger\_f1\_toreduce\_pushcap\_days",

default = 30.days,

min = Duration.Bottom,

max = Duration.Top)

with HasDurationConversion {

override val durationConversion = DurationConversion.FromDays

}

object NumberOfDaysToFilterForSeeLessOftenForF1TriggerNonF1

extends FSBoundedParam[Duration](

name = "seelessoften\_for\_f1\_trigger\_nonf1\_tofiltermr\_days",

default = 7.days,

min = Duration.Bottom,

max = Duration.Top)

with HasDurationConversion {

override val durationConversion = DurationConversion.FromDays

}

object NumberOfDaysToReducePushCapForSeeLessOftenForF1TriggerNonF1

extends FSBoundedParam[Duration](

name = "seelessoften\_for\_f1\_trigger\_non\_f1\_toreduce\_pushcap\_days",

default = 30.days,

min = Duration.Bottom,

max = Duration.Top)

with HasDurationConversion {

override val durationConversion = DurationConversion.FromDays

}

object NumberOfDaysToFilterForSeeLessOftenForNonF1TriggerF1

extends FSBoundedParam[Duration](

name = "seelessoften\_for\_nonf1\_trigger\_f1\_tofiltermr\_days",

default = 7.days,

min = Duration.Bottom,

max = Duration.Top)

with HasDurationConversion {

override val durationConversion = DurationConversion.FromDays

}

object NumberOfDaysToReducePushCapForSeeLessOftenForNonF1TriggerF1

extends FSBoundedParam[Duration](

name = "seelessoften\_for\_nonf1\_trigger\_f1\_toreduce\_pushcap\_days",

default = 30.days,

min = Duration.Bottom,

max = Duration.Top)

with HasDurationConversion {

override val durationConversion = DurationConversion.FromDays

}

object NumberOfDaysToFilterForSeeLessOftenForNonF1TriggerNonF1

extends FSBoundedParam[Duration](

name = "seelessoften\_for\_nonf1\_trigger\_nonf1\_tofiltermr\_days",

default = 7.days,

min = Duration.Bottom,

max = Duration.Top)

with HasDurationConversion {

override val durationConversion = DurationConversion.FromDays

}

object NumberOfDaysToReducePushCapForSeeLessOftenForNonF1TriggerNonF1

extends FSBoundedParam[Duration](

name = "seelessoften\_for\_nonf1\_trigger\_nonf1\_toreduce\_pushcap\_days",

default = 30.days,

min = Duration.Bottom,

max = Duration.Top)

with HasDurationConversion {

override val durationConversion = DurationConversion.FromDays

}

object EnableContFnF1TriggerSeeLessOftenFatigue

extends FSParam[Boolean](

name = "seelessoften\_fn\_enable\_f1\_trigger\_fatigue",

default = false

)

object EnableContFnNonF1TriggerSeeLessOftenFatigue

extends FSParam[Boolean](

name = "seelessoften\_fn\_enable\_nonf1\_trigger\_fatigue",

default = false

)

object SeeLessOftenListOfDayKnobs

extends FSParam[Seq[Double]]("seelessoften\_fn\_day\_knobs", default = Seq.empty[Double])

object SeeLessOftenListOfPushCapWeightKnobs

extends FSParam[Seq[Double]]("seelessoften\_fn\_pushcap\_knobs", default = Seq.empty[Double])

object SeeLessOftenListOfPowerKnobs

extends FSParam[Seq[Double]]("seelessoften\_fn\_power\_knobs", default = Seq.empty[Double])

object SeeLessOftenF1TriggerF1PushCapWeight

extends FSBoundedParam[Double](

"seelessoften\_fn\_f1\_trigger\_f1\_weight",

default = 1.0,

min = 0.0,

max = 10000000.0)

object SeeLessOftenF1TriggerNonF1PushCapWeight

extends FSBoundedParam[Double](

"seelessoften\_fn\_f1\_trigger\_nonf1\_weight",

default = 1.0,

min = 0.0,

max = 10000000.0)

object SeeLessOftenNonF1TriggerF1PushCapWeight

extends FSBoundedParam[Double](

"seelessoften\_fn\_nonf1\_trigger\_f1\_weight",

default = 1.0,

min = 0.0,

max = 10000000.0)

object SeeLessOftenNonF1TriggerNonF1PushCapWeight

extends FSBoundedParam[Double](

"seelessoften\_fn\_nonf1\_trigger\_nonf1\_weight",

default = 1.0,

min = 0.0,

max = 10000000.0)

object SeeLessOftenTripHqTweetTriggerF1PushCapWeight

extends FSBoundedParam[Double](

"seelessoften\_fn\_trip\_hq\_tweet\_trigger\_f1\_weight",

default = 1.0,

min = 0.0,

max = 10000000.0)

object SeeLessOftenTripHqTweetTriggerNonF1PushCapWeight

extends FSBoundedParam[Double](

"seelessoften\_fn\_trip\_hq\_tweet\_trigger\_nonf1\_weight",

default = 1.0,

min = 0.0,

max = 10000000.0)

object SeeLessOftenTripHqTweetTriggerTripHqTweetPushCapWeight

extends FSBoundedParam[Double](

"seelessoften\_fn\_trip\_hq\_tweet\_trigger\_trip\_hq\_tweet\_weight",

default = 1.0,

min = 0.0,

max = 10000000.0)

object SeeLessOftenTopicTriggerTopicPushCapWeight

extends FSBoundedParam[Double](

"seelessoften\_fn\_topic\_trigger\_topic\_weight",

default = 1.0,

min = 0.0,

max = Double.MaxValue)

object SeeLessOftenTopicTriggerF1PushCapWeight

extends FSBoundedParam[Double](

"seelessoften\_fn\_topic\_trigger\_f1\_weight",

default = 100000.0,

min = 0.0,

max = Double.MaxValue)

object SeeLessOftenTopicTriggerOONPushCapWeight

extends FSBoundedParam[Double](

"seelessoften\_fn\_topic\_trigger\_oon\_weight",

default = 100000.0,

min = 0.0,

max = Double.MaxValue)

object SeeLessOftenF1TriggerTopicPushCapWeight

extends FSBoundedParam[Double](

"seelessoften\_fn\_f1\_trigger\_topic\_weight",

default = 100000.0,

min = 0.0,

max = Double.MaxValue)

object SeeLessOftenOONTriggerTopicPushCapWeight

extends FSBoundedParam[Double](

"seelessoften\_fn\_oon\_trigger\_topic\_weight",

default = 1.0,

min = 0.0,

max = Double.MaxValue)

object SeeLessOftenDefaultPushCapWeight

extends FSBoundedParam[Double](

"seelessoften\_fn\_default\_weight",

default = 100000.0,

min = 0.0,

max = Double.MaxValue)

object SeeLessOftenNtabOnlyNotifUserPushCapWeight

extends FSBoundedParam[Double](

"seelessoften\_fn\_ntab\_only\_user\_weight",

default = 1.0,

min = 0.0,

max = Double.MaxValue)

// Params for inline feedback fatigue

object EnableContFnF1TriggerInlineFeedbackFatigue

extends FSParam[Boolean](

name = "feedback\_inline\_fn\_enable\_f1\_trigger\_fatigue",

default = false

)

object EnableContFnNonF1TriggerInlineFeedbackFatigue

extends FSParam[Boolean](

name = "feedback\_inline\_fn\_enable\_nonf1\_trigger\_fatigue",

default = false

)

object UseInlineDislikeForFatigue

extends FSParam[Boolean](

name = "feedback\_inline\_fn\_use\_dislike",

default = true

)

object UseInlineDismissForFatigue

extends FSParam[Boolean](

name = "feedback\_inline\_fn\_use\_dismiss",

default = false

)

object UseInlineSeeLessForFatigue

extends FSParam[Boolean](

name = "feedback\_inline\_fn\_use\_see\_less",

default = false

)

object UseInlineNotRelevantForFatigue

extends FSParam[Boolean](

name = "feedback\_inline\_fn\_use\_not\_relevant",

default = false

)

object InlineFeedbackListOfDayKnobs

extends FSParam[Seq[Double]]("feedback\_inline\_fn\_day\_knobs", default = Seq.empty[Double])

object InlineFeedbackListOfPushCapWeightKnobs

extends FSParam[Seq[Double]]("feedback\_inline\_fn\_pushcap\_knobs", default = Seq.empty[Double])

object InlineFeedbackListOfPowerKnobs

extends FSParam[Seq[Double]]("feedback\_inline\_fn\_power\_knobs", default = Seq.empty[Double])

object InlineFeedbackF1TriggerF1PushCapWeight

extends FSBoundedParam[Double](

"feedback\_inline\_fn\_f1\_trigger\_f1\_weight",

default = 1.0,

min = 0.0,

max = 10000000.0)

object InlineFeedbackF1TriggerNonF1PushCapWeight

extends FSBoundedParam[Double](

"feedback\_inline\_fn\_f1\_trigger\_nonf1\_weight",

default = 1.0,

min = 0.0,

max = 10000000.0)

object InlineFeedbackNonF1TriggerF1PushCapWeight

extends FSBoundedParam[Double](

"feedback\_inline\_fn\_nonf1\_trigger\_f1\_weight",

default = 1.0,

min = 0.0,

max = 10000000.0)

object InlineFeedbackNonF1TriggerNonF1PushCapWeight

extends FSBoundedParam[Double](

"feedback\_inline\_fn\_nonf1\_trigger\_nonf1\_weight",

default = 1.0,

min = 0.0,

max = 10000000.0)

// Params for prompt feedback

object EnableContFnF1TriggerPromptFeedbackFatigue

extends FSParam[Boolean](

name = "feedback\_prompt\_fn\_enable\_f1\_trigger\_fatigue",

default = false

)

object EnableContFnNonF1TriggerPromptFeedbackFatigue

extends FSParam[Boolean](

name = "feedback\_prompt\_fn\_enable\_nonf1\_trigger\_fatigue",

default = false

)

object PromptFeedbackListOfDayKnobs

extends FSParam[Seq[Double]]("feedback\_prompt\_fn\_day\_knobs", default = Seq.empty[Double])

object PromptFeedbackListOfPushCapWeightKnobs

extends FSParam[Seq[Double]]("feedback\_prompt\_fn\_pushcap\_knobs", default = Seq.empty[Double])

object PromptFeedbackListOfPowerKnobs

extends FSParam[Seq[Double]]("feedback\_prompt\_fn\_power\_knobs", default = Seq.empty[Double])

object PromptFeedbackF1TriggerF1PushCapWeight

extends FSBoundedParam[Double](

"feedback\_prompt\_fn\_f1\_trigger\_f1\_weight",

default = 1.0,

min = 0.0,

max = 10000000.0)

object PromptFeedbackF1TriggerNonF1PushCapWeight

extends FSBoundedParam[Double](

"feedback\_prompt\_fn\_f1\_trigger\_nonf1\_weight",

default = 1.0,

min = 0.0,

max = 10000000.0)

object PromptFeedbackNonF1TriggerF1PushCapWeight

extends FSBoundedParam[Double](

"feedback\_prompt\_fn\_nonf1\_trigger\_f1\_weight",

default = 1.0,

min = 0.0,

max = 10000000.0)

object PromptFeedbackNonF1TriggerNonF1PushCapWeight

extends FSBoundedParam[Double](

"feedback\_prompt\_fn\_nonf1\_trigger\_nonf1\_weight",

default = 1.0,

min = 0.0,

max = 10000000.0)

/\*

\* Param to enable cohost join event notif

\*/

object EnableSpaceCohostJoinEvent

extends FSParam[Boolean](name = "space\_recs\_cohost\_join\_enable", default = true)

/\*

\* Param to bypass global push cap when target is device following host/speaker.

\*/

object BypassGlobalSpacePushCapForSoftDeviceFollow

extends FSParam[Boolean](name = "space\_recs\_bypass\_global\_pushcap\_for\_soft\_follow", false)

/\*

\* Param to bypass active listener predicate when target is device following host/speaker.

\*/

object CheckActiveListenerPredicateForSoftDeviceFollow

extends FSParam[Boolean](name = "space\_recs\_check\_active\_listener\_for\_soft\_follow", false)

object SpreadControlRatioParam

extends FSBoundedParam[Double](

name = "oon\_spread\_control\_ratio",

default = 1000.0,

min = 0.0,

max = 100000.0

)

object FavOverSendThresholdParam

extends FSBoundedParam[Double](

name = "oon\_spread\_control\_fav\_over\_send\_threshold",

default = 0.14,

min = 0.0,

max = 1000.0

)

object AuthorReportRateThresholdParam

extends FSBoundedParam[Double](

name = "oon\_spread\_control\_author\_report\_rate\_threshold",

default = 7.4e-6,

min = 0.0,

max = 1000.0

)

object AuthorDislikeRateThresholdParam

extends FSBoundedParam[Double](

name = "oon\_spread\_control\_author\_dislike\_rate\_threshold",

default = 1.0,

min = 0.0,

max = 1000.0

)

object MinTweetSendsThresholdParam

extends FSBoundedParam[Double](

name = "oon\_spread\_control\_min\_tweet\_sends\_threshold",

default = 10000000000.0,

min = 0.0,

max = 10000000000.0

)

object MinAuthorSendsThresholdParam

extends FSBoundedParam[Double](

name = "oon\_spread\_control\_min\_author\_sends\_threshold",

default = 10000000000.0,

min = 0.0,

max = 10000000000.0

)

/\*

\* Tweet Ntab-dislike predicate related params

\*/

object TweetNtabDislikeCountThresholdParam

extends FSBoundedParam[Double](

name = "oon\_tweet\_ntab\_dislike\_count\_threshold",

default = 10000.0,

min = 0.0,

max = 10000.0

)

object TweetNtabDislikeRateThresholdParam

extends FSBoundedParam[Double](

name = "oon\_tweet\_ntab\_dislike\_rate\_threshold",

default = 1.0,

min = 0.0,

max = 1.0

)

/\*\*

\* Param for tweet language feature name

\*/

object TweetLanguageFeatureNameParam

extends FSParam[String](

name = "language\_tweet\_language\_feature\_name",

default = "tweet.language.tweet.identified")

/\*\*

\* Threshold for user inferred language filtering

\*/

object UserInferredLanguageThresholdParam

extends FSBoundedParam[Double](

name = "language\_user\_inferred\_language\_threshold",

default = 0.0,

min = 0.0,

max = 1.0

)

/\*\*

\* Threshold for user device language filtering

\*/

object UserDeviceLanguageThresholdParam

extends FSBoundedParam[Double](

name = "language\_user\_device\_language\_threshold",

default = 0.0,

min = 0.0,

max = 1.0

)

/\*\*

\* Param to enable/disable tweet language filter

\*/

object EnableTweetLanguageFilter

extends FSParam[Boolean](

name = "language\_enable\_tweet\_language\_filter",

default = false

)

/\*\*

\* Param to skip language filter for media tweets

\*/

object SkipLanguageFilterForMediaTweets

extends FSParam[Boolean](

name = "language\_skip\_language\_filter\_for\_media\_tweets",

default = false

)

/\*

\* Tweet Ntab-dislike predicate related params for MrTwistly

\*/

object TweetNtabDislikeCountThresholdForMrTwistlyParam

extends FSBoundedParam[Double](

name = "oon\_tweet\_ntab\_dislike\_count\_threshold\_for\_mrtwistly",

default = 10000.0,

min = 0.0,

max = 10000.0

)

object TweetNtabDislikeRateThresholdForMrTwistlyParam

extends FSBoundedParam[Double](

name = "oon\_tweet\_ntab\_dislike\_rate\_threshold\_for\_mrtwistly",

default = 1.0,

min = 0.0,

max = 1.0

)

object TweetNtabDislikeCountBucketThresholdParam

extends FSBoundedParam[Double](

name = "oon\_tweet\_ntab\_dislike\_count\_bucket\_threshold",

default = 10.0,

min = 0.0,

max = 10000.0

)

/\*

\* Tweet engagement ratio predicate related params

\*/

object TweetQTtoNtabClickRatioThresholdParam

extends FSBoundedParam[Double](

name = "oon\_tweet\_engagement\_filter\_qt\_to\_ntabclick\_ratio\_threshold",

default = 0.0,

min = 0.0,

max = 100000.0

)

/\*\*

\* Lower bound threshold to filter a tweet based on its reply to like ratio

\*/

object TweetReplytoLikeRatioThresholdLowerBound

extends FSBoundedParam[Double](

name = "oon\_tweet\_engagement\_filter\_reply\_to\_like\_ratio\_threshold\_lower\_bound",

default = Double.MaxValue,

min = 0.0,

max = Double.MaxValue

)

/\*\*

\* Upper bound threshold to filter a tweet based on its reply to like ratio

\*/

object TweetReplytoLikeRatioThresholdUpperBound

extends FSBoundedParam[Double](

name = "oon\_tweet\_engagement\_filter\_reply\_to\_like\_ratio\_threshold\_upper\_bound",

default = 0.0,

min = 0.0,

max = Double.MaxValue

)

/\*\*

\* Upper bound threshold to filter a tweet based on its reply to like ratio

\*/

object TweetReplytoLikeRatioReplyCountThreshold

extends FSBoundedParam[Int](

name = "oon\_tweet\_engagement\_filter\_reply\_count\_threshold",

default = Int.MaxValue,

min = 0,

max = Int.MaxValue

)

/\*

\* oonTweetLengthBasedPrerankingPredicate related params

\*/

object OonTweetLengthPredicateUpdatedMediaLogic

extends FSParam[Boolean](

name = "oon\_quality\_filter\_tweet\_length\_updated\_media\_logic",

default = false

)

object OonTweetLengthPredicateUpdatedQuoteTweetLogic

extends FSParam[Boolean](

name = "oon\_quality\_filter\_tweet\_length\_updated\_quote\_tweet\_logic",

default = false

)

object OonTweetLengthPredicateMoreStrictForUndefinedLanguages

extends FSParam[Boolean](

name = "oon\_quality\_filter\_tweet\_length\_more\_strict\_for\_undefined\_languages",

default = false

)

object EnablePrerankingTweetLengthPredicate

extends FSParam[Boolean](

name = "oon\_quality\_filter\_enable\_preranking\_filter",

default = false

)

/\*

\* LengthLanguageBasedOONTweetCandidatesQualityPredicate related params

\*/

object SautOonWithMediaTweetLengthThresholdParam

extends FSBoundedParam[Double](

name = "oon\_quality\_filter\_tweet\_length\_threshold\_for\_saut\_oon\_with\_media",

default = 0.0,

min = 0.0,

max = 70.0

)

object NonSautOonWithMediaTweetLengthThresholdParam

extends FSBoundedParam[Double](

name = "oon\_quality\_filter\_tweet\_length\_threshold\_for\_non\_saut\_oon\_with\_media",

default = 0.0,

min = 0.0,

max = 70.0

)

object SautOonWithoutMediaTweetLengthThresholdParam

extends FSBoundedParam[Double](

name = "oon\_quality\_filter\_tweet\_length\_threshold\_for\_saut\_oon\_without\_media",

default = 0.0,

min = 0.0,

max = 70.0

)

object NonSautOonWithoutMediaTweetLengthThresholdParam

extends FSBoundedParam[Double](

name = "oon\_quality\_filter\_tweet\_length\_threshold\_for\_non\_saut\_oon\_without\_media",

default = 0.0,

min = 0.0,

max = 70.0

)

object ArgfOonWithMediaTweetWordLengthThresholdParam

extends FSBoundedParam[Double](

name = "oon\_quality\_filter\_tweet\_word\_length\_threshold\_for\_argf\_oon\_with\_media",

default = 0.0,

min = 0.0,

max = 18.0

)

object EsfthOonWithMediaTweetWordLengthThresholdParam

extends FSBoundedParam[Double](

name = "oon\_quality\_filter\_tweet\_word\_length\_threshold\_for\_esfth\_oon\_with\_media",

default = 0.0,

min = 0.0,

max = 10.0

)

/\*\*

\* Param to enable/disable sentiment feature hydration

\*/

object EnableMrTweetSentimentFeatureHydrationFS

extends FSParam[Boolean](

name = "feature\_hydration\_enable\_mr\_tweet\_sentiment\_feature",

default = false

)

/\*\*

\* Param to enable/disable feature map scribing for staging test log

\*/

object EnableMrScribingMLFeaturesAsFeatureMapForStaging

extends FSParam[Boolean](

name = "frigate\_pushservice\_enable\_scribing\_ml\_features\_as\_featuremap\_for\_staging",

default = false

)

/\*\*

\* Param to enable timeline health signal hydration

\* \*/

object EnableTimelineHealthSignalHydration

extends FSParam[Boolean](

name = "timeline\_health\_signal\_hydration",

default = false

)

/\*\*

\* Param to enable timeline health signal hydration for model training

\* \*/

object EnableTimelineHealthSignalHydrationForModelTraining

extends FSParam[Boolean](

name = "timeline\_health\_signal\_hydration\_for\_model\_training",

default = false

)

/\*\*

\* Param to enable/disable mr user social context agg feature hydration

\*/

object EnableMrUserSocialContextAggregateFeatureHydration

extends FSParam[Boolean](

name = "frigate\_push\_modeling\_hydrate\_mr\_user\_social\_context\_agg\_feature",

default = true

)

/\*\*

\* Param to enable/disable mr user semantic core agg feature hydration

\*/

object EnableMrUserSemanticCoreAggregateFeatureHydration

extends FSParam[Boolean](

name = "frigate\_push\_modeling\_hydrate\_mr\_user\_semantic\_core\_agg\_feature",

default = true

)

/\*\*

\* Param to enable/disable mr user candidate sparse agg feature hydration

\*/

object EnableMrUserCandidateSparseOfflineAggregateFeatureHydration

extends FSParam[Boolean](

name = "frigate\_push\_modeling\_hydrate\_mr\_user\_candidate\_sparse\_agg\_feature",

default = true

)

/\*\*

\* Param to enable/disable mr user candidate agg feature hydration

\*/

object EnableMrUserCandidateOfflineAggregateFeatureHydration

extends FSParam[Boolean](

name = "frigate\_push\_modeling\_hydrate\_mr\_user\_candidate\_agg\_feature",

default = true

)

/\*\*

\* Param to enable/disable mr user candidate compact agg feature hydration

\*/

object EnableMrUserCandidateOfflineCompactAggregateFeatureHydration

extends FSParam[Boolean](

name = "frigate\_push\_modeling\_hydrate\_mr\_user\_candidate\_compact\_agg\_feature",

default = false

)

/\*\*

\* Param to enable/disable mr real graph user-author/social-context feature hydration

\*/

object EnableRealGraphUserAuthorAndSocialContxtFeatureHydration

extends FSParam[Boolean](

name = "frigate\_push\_modeling\_hydrate\_real\_graph\_user\_social\_feature",

default = true

)

/\*\*

\* Param to enable/disable mr user author agg feature hydration

\*/

object EnableMrUserAuthorOfflineAggregateFeatureHydration

extends FSParam[Boolean](

name = "frigate\_push\_modeling\_hydrate\_mr\_user\_author\_agg\_feature",

default = true

)

/\*\*

\* Param to enable/disable mr user author compact agg feature hydration

\*/

object EnableMrUserAuthorOfflineCompactAggregateFeatureHydration

extends FSParam[Boolean](

name = "frigate\_push\_modeling\_hydrate\_mr\_user\_author\_compact\_agg\_feature",

default = false

)

/\*\*

\* Param to enable/disable mr user compact agg feature hydration

\*/

object EnableMrUserOfflineCompactAggregateFeatureHydration

extends FSParam[Boolean](

name = "frigate\_push\_modeling\_hydrate\_mr\_user\_compact\_agg\_feature",

default = false

)

/\*\*

\* Param to enable/disable mr user simcluster agg feature hydration

\*/

object EnableMrUserSimcluster2020AggregateFeatureHydration

extends FSParam[Boolean](

name = "frigate\_push\_modeling\_hydrate\_mr\_user\_simcluster\_agg\_feature",

default = true

)

/\*\*

\* Param to enable/disable mr user agg feature hydration

\*/

object EnableMrUserOfflineAggregateFeatureHydration

extends FSParam[Boolean](

name = "frigate\_push\_modeling\_hydrate\_mr\_user\_agg\_feature",

default = true

)

/\*\*

\* Param to enable/disable topic engagement RTA in the ranking model

\*/

object EnableTopicEngagementRealTimeAggregatesFS

extends FSParam[Boolean](

"feature\_hydration\_enable\_htl\_topic\_engagement\_real\_time\_agg\_feature",

false)

/\*

\* Param to enable mr user semantic core feature hydration for heavy ranker

\* \*/

object EnableMrUserSemanticCoreFeatureForExpt

extends FSParam[Boolean](

name = "frigate\_push\_modeling\_hydrate\_mr\_user\_semantic\_core",

default = false)

/\*\*

\* Param to enable hydrating user duration since last visit features

\*/

object EnableHydratingUserDurationSinceLastVisitFeatures

extends FSParam[Boolean](

name = "feature\_hydration\_user\_duration\_since\_last\_visit",

default = false)

/\*\*

Param to enable/disable user-topic aggregates in the ranking model

\*/

object EnableUserTopicAggregatesFS

extends FSParam[Boolean]("feature\_hydration\_enable\_htl\_topic\_user\_agg\_feature", false)

/\*

\* PNegMultimodalPredicate related params

\*/

object EnablePNegMultimodalPredicateParam

extends FSParam[Boolean](

name = "pneg\_multimodal\_filter\_enable\_param",

default = false

)

object PNegMultimodalPredicateModelThresholdParam

extends FSBoundedParam[Double](

name = "pneg\_multimodal\_filter\_model\_threshold\_param",

default = 1.0,

min = 0.0,

max = 1.0

)

object PNegMultimodalPredicateBucketThresholdParam

extends FSBoundedParam[Double](

name = "pneg\_multimodal\_filter\_bucket\_threshold\_param",

default = 0.4,

min = 0.0,

max = 1.0

)

/\*

\* NegativeKeywordsPredicate related params

\*/

object EnableNegativeKeywordsPredicateParam

extends FSParam[Boolean](

name = "negative\_keywords\_filter\_enable\_param",

default = false

)

object NegativeKeywordsPredicateDenylist

extends FSParam[Seq[String]](

name = "negative\_keywords\_filter\_denylist",

default = Seq.empty[String]

)

/\*

\* LightRanking related params

\*/

object EnableLightRankingParam

extends FSParam[Boolean](

name = "light\_ranking\_enable\_param",

default = false

)

object LightRankingNumberOfCandidatesParam

extends FSBoundedParam[Int](

name = "light\_ranking\_number\_of\_candidates\_param",

default = 100,

min = 0,

max = 1000

)

object LightRankingModelTypeParam

extends FSParam[String](

name = "light\_ranking\_model\_type\_param",

default = "WeightedOpenOrNtabClickProbability\_Q4\_2021\_13172\_Mr\_Light\_Ranker\_Dbv2\_Top3")

object EnableRandomBaselineLightRankingParam

extends FSParam[Boolean](

name = "light\_ranking\_random\_baseline\_enable\_param",

default = false

)

object LightRankingScribeCandidatesDownSamplingParam

extends FSBoundedParam[Double](

name = "light\_ranking\_scribe\_candidates\_down\_sampling\_param",

default = 1.0,

min = 0.0,

max = 1.0

)

/\*

\* Quality Upranking related params

\*/

object EnableProducersQualityBoostingForHeavyRankingParam

extends FSParam[Boolean](

name = "quality\_upranking\_enable\_producers\_quality\_boosting\_for\_heavy\_ranking\_param",

default = false

)

object QualityUprankingBoostForHighQualityProducersParam

extends FSBoundedParam[Double](

name = "quality\_upranking\_boost\_for\_high\_quality\_producers\_param",

default = 1.0,

min = 0.0,

max = 10000.0

)

object QualityUprankingDownboostForLowQualityProducersParam

extends FSBoundedParam[Double](

name = "quality\_upranking\_downboost\_for\_low\_quality\_producers\_param",

default = 1.0,

min = 0.0,

max = 1.0

)

object EnableQualityUprankingForHeavyRankingParam

extends FSParam[Boolean](

name = "quality\_upranking\_enable\_for\_heavy\_ranking\_param",

default = false

)

object QualityUprankingModelTypeParam

extends FSParam[WeightedOpenOrNtabClickModel.ModelNameType](

name = "quality\_upranking\_model\_id",

default = "Q4\_2022\_Mr\_Bqml\_Quality\_Model\_wALL"

)

object QualityUprankingTransformTypeParam

extends FSEnumParam[MrQualityUprankingTransformTypeEnum.type](

name = "quality\_upranking\_transform\_id",

default = MrQualityUprankingTransformTypeEnum.Sigmoid,

enum = MrQualityUprankingTransformTypeEnum

)

object QualityUprankingBoostForHeavyRankingParam

extends FSBoundedParam[Double](

name = "quality\_upranking\_boost\_for\_heavy\_ranking\_param",

default = 1.0,

min = -10.0,

max = 10.0

)

object QualityUprankingSigmoidBiasForHeavyRankingParam

extends FSBoundedParam[Double](

name = "quality\_upranking\_sigmoid\_bias\_for\_heavy\_ranking\_param",

default = 0.0,

min = -10.0,

max = 10.0

)

object QualityUprankingSigmoidWeightForHeavyRankingParam

extends FSBoundedParam[Double](

name = "quality\_upranking\_sigmoid\_weight\_for\_heavy\_ranking\_param",

default = 1.0,

min = -10.0,

max = 10.0

)

object QualityUprankingLinearBarForHeavyRankingParam

extends FSBoundedParam[Double](

name = "quality\_upranking\_linear\_bar\_for\_heavy\_ranking\_param",

default = 1.0,

min = 0.0,

max = 10.0

)

object EnableQualityUprankingCrtScoreStatsForHeavyRankingParam

extends FSParam[Boolean](

name = "quality\_upranking\_enable\_crt\_score\_stats\_for\_heavy\_ranking\_param",

default = false

)

/\*

\* BQML Health Model related params

\*/

object EnableBqmlHealthModelPredicateParam

extends FSParam[Boolean](

name = "bqml\_health\_model\_filter\_enable\_param",

default = false

)

object EnableBqmlHealthModelPredictionForInNetworkCandidatesParam

extends FSParam[Boolean](

name = "bqml\_health\_model\_enable\_prediction\_for\_in\_network\_candidates\_param",

default = false

)

object BqmlHealthModelTypeParam

extends FSParam[HealthNsfwModel.ModelNameType](

name = "bqml\_health\_model\_id",

default = HealthNsfwModel.Q2\_2022\_Mr\_Bqml\_Health\_Model\_NsfwV0

)

object BqmlHealthModelPredicateFilterThresholdParam

extends FSBoundedParam[Double](

name = "bqml\_health\_model\_filter\_threshold\_param",

default = 1.0,

min = 0.0,

max = 1.0

)

object BqmlHealthModelPredicateBucketThresholdParam

extends FSBoundedParam[Double](

name = "bqml\_health\_model\_bucket\_threshold\_param",

default = 0.005,

min = 0.0,

max = 1.0

)

object EnableBqmlHealthModelScoreHistogramParam

extends FSParam[Boolean](

name = "bqml\_health\_model\_score\_histogram\_enable\_param",

default = false

)

/\*

\* BQML Quality Model related params

\*/

object EnableBqmlQualityModelPredicateParam

extends FSParam[Boolean](

name = "bqml\_quality\_model\_filter\_enable\_param",

default = false

)

object EnableBqmlQualityModelScoreHistogramParam

extends FSParam[Boolean](

name = "bqml\_quality\_model\_score\_histogram\_enable\_param",

default = false

)

object BqmlQualityModelTypeParam

extends FSParam[WeightedOpenOrNtabClickModel.ModelNameType](

name = "bqml\_quality\_model\_id",

default = "Q1\_2022\_13562\_Mr\_Bqml\_Quality\_Model\_V2"

)

/\*\*

\* Param to specify which quality models to use to get the scores for determining

\* whether to bucket a user for the DDG

\*/

object BqmlQualityModelBucketModelIdListParam

extends FSParam[Seq[WeightedOpenOrNtabClickModel.ModelNameType]](

name = "bqml\_quality\_model\_bucket\_model\_id\_list",

default = Seq(

"Q1\_2022\_13562\_Mr\_Bqml\_Quality\_Model\_V2",

"Q2\_2022\_DDG14146\_Mr\_Personalised\_BQML\_Quality\_Model",

"Q2\_2022\_DDG14146\_Mr\_NonPersonalised\_BQML\_Quality\_Model"

)

)

object BqmlQualityModelPredicateThresholdParam

extends FSBoundedParam[Double](

name = "bqml\_quality\_model\_filter\_threshold\_param",

default = 1.0,

min = 0.0,

max = 1.0

)

/\*\*

\* Param to specify the threshold to determine if a user’s quality score is high enough to enter the experiment.

\*/

object BqmlQualityModelBucketThresholdListParam

extends FSParam[Seq[Double]](

name = "bqml\_quality\_model\_bucket\_threshold\_list",

default = Seq(0.7, 0.7, 0.7)

)

/\*

\* TweetAuthorAggregates related params

\*/

object EnableTweetAuthorAggregatesFeatureHydrationParam

extends FSParam[Boolean](

name = "tweet\_author\_aggregates\_feature\_hydration\_enable\_param",

default = false

)

/\*\*

\* Param to determine if we should include the relevancy score of candidates in the Ibis payload

\*/

object IncludeRelevanceScoreInIbis2Payload

extends FSParam[Boolean](

name = "relevance\_score\_include\_in\_ibis2\_payload",

default = false

)

/\*\*

\* Param to specify supervised model to predict score by sending the notification

\*/

object BigFilteringSupervisedSendingModelParam

extends FSParam[BigFilteringSupervisedModel.ModelNameType](

name = "ltv\_filtering\_bigfiltering\_supervised\_sending\_model\_param",

default = BigFilteringSupervisedModel.V0\_0\_BigFiltering\_Supervised\_Sending\_Model

)

/\*\*

\* Param to specify supervised model to predict score by not sending the notification

\*/

object BigFilteringSupervisedWithoutSendingModelParam

extends FSParam[BigFilteringSupervisedModel.ModelNameType](

name = "ltv\_filtering\_bigfiltering\_supervised\_without\_sending\_model\_param",

default = BigFilteringSupervisedModel.V0\_0\_BigFiltering\_Supervised\_Without\_Sending\_Model

)

/\*\*

\* Param to specify RL model to predict score by sending the notification

\*/

object BigFilteringRLSendingModelParam

extends FSParam[BigFilteringSupervisedModel.ModelNameType](

name = "ltv\_filtering\_bigfiltering\_rl\_sending\_model\_param",

default = BigFilteringRLModel.V0\_0\_BigFiltering\_Rl\_Sending\_Model

)

/\*\*

\* Param to specify RL model to predict score by not sending the notification

\*/

object BigFilteringRLWithoutSendingModelParam

extends FSParam[BigFilteringSupervisedModel.ModelNameType](

name = "ltv\_filtering\_bigfiltering\_rl\_without\_sending\_model\_param",

default = BigFilteringRLModel.V0\_0\_BigFiltering\_Rl\_Without\_Sending\_Model

)

/\*\*

\* Param to specify the threshold (send notification if score >= threshold)

\*/

object BigFilteringThresholdParam

extends FSBoundedParam[Double](

name = "ltv\_filtering\_bigfiltering\_threshold\_param",

default = 0.0,

min = Double.MinValue,

max = Double.MaxValue

)

/\*\*

\* Param to specify normalization used for BigFiltering

\*/

object BigFilteringNormalizationTypeIdParam

extends FSEnumParam[BigFilteringNormalizationEnum.type](

name = "ltv\_filtering\_bigfiltering\_normalization\_type\_id",

default = BigFilteringNormalizationEnum.NormalizationDisabled,

enum = BigFilteringNormalizationEnum

)

/\*\*

\* Param to specify histograms of model scores in BigFiltering

\*/

object BigFilteringEnableHistogramsParam

extends FSParam[Boolean](

name = "ltv\_filtering\_bigfiltering\_enable\_histograms\_param",

default = false

)

/\*

\* Param to enable sending requests to Ins Sender

\*/

object EnableInsSender extends FSParam[Boolean](name = "ins\_enable\_dark\_traffic", default = false)

/\*\*

\* Param to specify the range of relevance scores for MagicFanout types.

\*/

object MagicFanoutRelevanceScoreRange

extends FSParam[Seq[Double]](

name = "relevance\_score\_mf\_range",

default = Seq(0.75, 1.0)

)

/\*\*

\* Param to specify the range of relevance scores for MR types.

\*/

object MagicRecsRelevanceScoreRange

extends FSParam[Seq[Double]](

name = "relevance\_score\_mr\_range",

default = Seq(0.25, 0.5)

)

/\*\*

\* Param to enable backfilling OON candidates if number of F1 candidates is greater than a threshold K.

\*/

object EnableOONBackfillBasedOnF1Candidates

extends FSParam[Boolean](name = "oon\_enable\_backfill\_based\_on\_f1", default = false)

/\*\*

\* Threshold for the minimum number of F1 candidates required to enable backfill of OON candidates.

\*/

object NumberOfF1CandidatesThresholdForOONBackfill

extends FSBoundedParam[Int](

name = "oon\_enable\_backfill\_f1\_threshold",

min = 0,

default = 5000,

max = 5000)

/\*\*

\* Event ID allowlist to skip account country predicate

\*/

object MagicFanoutEventAllowlistToSkipAccountCountryPredicate

extends FSParam[Seq[Long]](

name = "magicfanout\_event\_allowlist\_skip\_account\_country\_predicate",

default = Seq.empty[Long]

)

/\*\*

\* MagicFanout Event Semantic Core Domain Ids

\*/

object ListOfEventSemanticCoreDomainIds

extends FSParam[Seq[Long]](

name = "magicfanout\_automated\_events\_semantic\_core\_domain\_ids",

default = Seq())

/\*\*

\* Adhoc id for detailed rank flow stats

\*/

object ListOfAdhocIdsForStatsTracking

extends FSParam[Set[Long]](

name = "stats\_enable\_detailed\_stats\_tracking\_ids",

default = Set.empty[Long]

)

object EnableGenericCRTBasedFatiguePredicate

extends FSParam[Boolean](

name = "seelessoften\_enable\_generic\_crt\_based\_fatigue\_predicate",

default = false)

/\*\*

\* Param to enable copy features such as Emojis and Target Name

\*/

object EnableCopyFeaturesForF1

extends FSParam[Boolean](name = "mr\_copy\_enable\_features\_f1", default = false)

/\*\*

\* Param to enable copy features such as Emojis and Target Name

\*/

object EnableCopyFeaturesForOon

extends FSParam[Boolean](name = "mr\_copy\_enable\_features\_oon", default = false)

/\*\*

\* Param to enable Emoji in F1 Copy

\*/

object EnableEmojiInF1Copy

extends FSParam[Boolean](name = "mr\_copy\_enable\_f1\_emoji", default = false)

/\*\*

\* Param to enable Target in F1 Copy

\*/

object EnableTargetInF1Copy

extends FSParam[Boolean](name = "mr\_copy\_enable\_f1\_target", default = false)

/\*\*

\* Param to enable Emoji in OON Copy

\*/

object EnableEmojiInOonCopy

extends FSParam[Boolean](name = "mr\_copy\_enable\_oon\_emoji", default = false)

/\*\*

\* Param to enable Target in OON Copy

\*/

object EnableTargetInOonCopy

extends FSParam[Boolean](name = "mr\_copy\_enable\_oon\_target", default = false)

/\*\*

\* Param to enable split fatigue for Target and Emoji copy for OON and F1

\*/

object EnableTargetAndEmojiSplitFatigue

extends FSParam[Boolean](name = "mr\_copy\_enable\_target\_emoji\_split\_fatigue", default = false)

/\*\*

\* Param to enable experimenting string on the body

\*/

object EnableF1CopyBody extends FSParam[Boolean](name = "mr\_copy\_f1\_enable\_body", default = false)

object EnableOONCopyBody

extends FSParam[Boolean](name = "mr\_copy\_oon\_enable\_body", default = false)

object EnableIosCopyBodyTruncate

extends FSParam[Boolean](name = "mr\_copy\_enable\_body\_truncate", default = false)

object EnableNsfwCopy extends FSParam[Boolean](name = "mr\_copy\_enable\_nsfw", default = false)

/\*\*

\* Param to determine F1 candidate nsfw score threshold

\*/

object NsfwScoreThresholdForF1Copy

extends FSBoundedParam[Double](

name = "mr\_copy\_nsfw\_threshold\_f1",

default = 0.3,

min = 0.0,

max = 1.0

)

/\*\*

\* Param to determine OON candidate nsfw score threshold

\*/

object NsfwScoreThresholdForOONCopy

extends FSBoundedParam[Double](

name = "mr\_copy\_nsfw\_threshold\_oon",

default = 0.2,

min = 0.0,

max = 1.0

)

/\*\*

\* Param to determine the lookback duration when searching for prev copy features.

\*/

object CopyFeaturesHistoryLookbackDuration

extends FSBoundedParam[Duration](

name = "mr\_copy\_history\_lookback\_duration\_in\_days",

default = 30.days,

min = Duration.Bottom,

max = Duration.Top)

with HasDurationConversion {

override val durationConversion = DurationConversion.FromDays

}

/\*\*

\* Param to determine the F1 emoji copy fatigue in # of hours.

\*/

object F1EmojiCopyFatigueDuration

extends FSBoundedParam[Duration](

name = "mr\_copy\_f1\_emoji\_copy\_fatigue\_in\_hours",

default = 24.hours,

min = 0.hours,

max = Duration.Top)

with HasDurationConversion {

override val durationConversion = DurationConversion.FromHours

}

/\*\*

\* Param to determine the F1 target copy fatigue in # of hours.

\*/

object F1TargetCopyFatigueDuration

extends FSBoundedParam[Duration](

name = "mr\_copy\_f1\_target\_copy\_fatigue\_in\_hours",

default = 24.hours,

min = 0.hours,

max = Duration.Top)

with HasDurationConversion {

override val durationConversion = DurationConversion.FromHours

}

/\*\*

\* Param to determine the OON emoji copy fatigue in # of hours.

\*/

object OonEmojiCopyFatigueDuration

extends FSBoundedParam[Duration](

name = "mr\_copy\_oon\_emoji\_copy\_fatigue\_in\_hours",

default = 24.hours,

min = 0.hours,

max = Duration.Top)

with HasDurationConversion {

override val durationConversion = DurationConversion.FromHours

}

/\*\*

\* Param to determine the OON target copy fatigue in # of hours.

\*/

object OonTargetCopyFatigueDuration

extends FSBoundedParam[Duration](

name = "mr\_copy\_oon\_target\_copy\_fatigue\_in\_hours",

default = 24.hours,

min = 0.hours,

max = Duration.Top)

with HasDurationConversion {

override val durationConversion = DurationConversion.FromHours

}

/\*\*

\* Param to turn on/off home timeline based fatigue rule, where once last home timeline visit

\* is larger than the specified will evalute to not fatigue

\*/

object EnableHTLBasedFatigueBasicRule

extends FSParam[Boolean](

name = "mr\_copy\_enable\_htl\_based\_fatigue\_basic\_rule",

default = false)

/\*\*

\* Param to determine f1 emoji copy fatigue in # of pushes

\*/

object F1EmojiCopyNumOfPushesFatigue

extends FSBoundedParam[Int](

name = "mr\_copy\_f1\_emoji\_copy\_number\_of\_pushes\_fatigue",

default = 0,

min = 0,

max = 200

)

/\*\*

\* Param to determine oon emoji copy fatigue in # of pushes

\*/

object OonEmojiCopyNumOfPushesFatigue

extends FSBoundedParam[Int](

name = "mr\_copy\_oon\_emoji\_copy\_number\_of\_pushes\_fatigue",

default = 0,

min = 0,

max = 200

)

/\*\*

\* If user haven't visited home timeline for certain duration, we will

\* exempt user from feature copy fatigue. This param is used to control

\* how long it is before we enter exemption.

\*/

object MinFatigueDurationSinceLastHTLVisit

extends FSBoundedParam[Duration](

name = "mr\_copy\_min\_duration\_since\_last\_htl\_visit\_hours",

default = Duration.Top,

min = 0.hour,

max = Duration.Top,

)

with HasDurationConversion {

override val durationConversion = DurationConversion.FromHours

}

/\*\*

\* If a user haven't visit home timeline very long, the user will return

\* to fatigue state under the home timeline based fatigue rule. There will

\* only be a window, where the user is out of fatigue state under the rule.

\* This param control the length of the non fatigue period.

\*/

object LastHTLVisitBasedNonFatigueWindow

extends FSBoundedParam[Duration](

name = "mr\_copy\_last\_htl\_visit\_based\_non\_fatigue\_window\_hours",

default = 48.hours,

min = 0.hour,

max = Duration.Top,

)

with HasDurationConversion {

override val durationConversion = DurationConversion.FromHours

}

object EnableOONCBasedCopy

extends FSParam[Boolean](

name = "mr\_copy\_enable\_oonc\_based\_copy",

default = false

)

object HighOONCThresholdForCopy

extends FSBoundedParam[Double](

name = "mr\_copy\_high\_oonc\_threshold\_for\_copy",

default = 1.0,

min = 0.0,

max = 1.0

)

object LowOONCThresholdForCopy

extends FSBoundedParam[Double](

name = "mr\_copy\_low\_oonc\_threshold\_for\_copy",

default = 0.0,

min = 0.0,

max = 1.0

)

object EnableTweetTranslation

extends FSParam[Boolean](name = "tweet\_translation\_enable", default = false)

object TripTweetCandidateReturnEnable

extends FSParam[Boolean](name = "trip\_tweet\_candidate\_enable", default = false)

object TripTweetCandidateSourceIds

extends FSParam[Seq[String]](

name = "trip\_tweet\_candidate\_source\_ids",

default = Seq("TOP\_GEO\_V3"))

object TripTweetMaxTotalCandidates

extends FSBoundedParam[Int](

name = "trip\_tweet\_max\_total\_candidates",

default = 500,

min = 10,

max = 1000)

object EnableEmptyBody

extends FSParam[Boolean](name = "push\_presentation\_enable\_empty\_body", default = false)

object EnableSocialContextForRetweet

extends FSParam[Boolean](name = "push\_presentation\_social\_context\_retweet", default = false)

/\*\*

\* Param to enable/disable simcluster feature hydration

\*/

object EnableMrTweetSimClusterFeatureHydrationFS

extends FSParam[Boolean](

name = "feature\_hydration\_enable\_mr\_tweet\_simcluster\_feature",

default = false

)

/\*\*

\* Param to disable OON candidates based on tweetAuthor

\*/

object DisableOutNetworkTweetCandidatesFS

extends FSParam[Boolean](name = "oon\_filtering\_disable\_oon\_candidates", default = false)

/\*\*

\* Param to enable Local Viral Tweets

\*/

object EnableLocalViralTweets

extends FSParam[Boolean](name = "local\_viral\_tweets\_enable", default = true)

/\*\*

\* Param to enable Explore Video Tweets

\*/

object EnableExploreVideoTweets

extends FSParam[Boolean](name = "explore\_video\_tweets\_enable", default = false)

/\*\*

\* Param to enable List Recommendations

\*/

object EnableListRecommendations

extends FSParam[Boolean](name = "list\_recommendations\_enable", default = false)

/\*\*

\* Param to enable IDS List Recommendations

\*/

object EnableIDSListRecommendations

extends FSParam[Boolean](name = "list\_recommendations\_ids\_enable", default = false)

/\*\*

\* Param to enable PopGeo List Recommendations

\*/

object EnablePopGeoListRecommendations

extends FSParam[Boolean](name = "list\_recommendations\_pop\_geo\_enable", default = false)

/\*\*

\* Param to control the inverter for fatigue between consecutive ListRecommendations

\*/

object ListRecommendationsPushInterval

extends FSBoundedParam[Duration](

name = "list\_recommendations\_interval\_days",

default = 24.hours,

min = Duration.Bottom,

max = Duration.Top)

with HasDurationConversion {

override val durationConversion = DurationConversion.FromDays

}

/\*\*

\* Param to control the granularity of GeoHash for ListRecommendations

\*/

object ListRecommendationsGeoHashLength

extends FSBoundedParam[Int](

name = "list\_recommendations\_geo\_hash\_length",

default = 5,

min = 3,

max = 5)

/\*\*

\* Param to control maximum number of ListRecommendation pushes to receive in an interval

\*/

object MaxListRecommendationsPushGivenInterval

extends FSBoundedParam[Int](

name = "list\_recommendations\_push\_given\_interval",

default = 1,

min = 0,

max = 10

)

/\*\*

\* Param to control the subscriber count for list recommendation

\*/

object ListRecommendationsSubscriberCount

extends FSBoundedParam[Int](

name = "list\_recommendations\_subscriber\_count",

default = 0,

min = 0,

max = Integer.MAX\_VALUE)

/\*\*

\* Param to define dynamic inline action types for web notifications (both desktop web + mobile web)

\*/

object LocalViralTweetsBucket

extends FSParam[String](

name = "local\_viral\_tweets\_bucket",

default = "high",

)

/\*\*

\* List of CrTags to disable

\*/

object OONCandidatesDisabledCrTagParam

extends FSParam[Seq[String]](

name = "oon\_enable\_oon\_candidates\_disabled\_crtag",

default = Seq.empty[String]

)

/\*\*

\* List of Crt groups to disable

\*/

object OONCandidatesDisabledCrtGroupParam

extends FSEnumSeqParam[CrtGroupEnum.type](

name = "oon\_enable\_oon\_candidates\_disabled\_crt\_group\_ids",

default = Seq.empty[CrtGroupEnum.Value],

enum = CrtGroupEnum

)

/\*\*

\* Param to enable launching video tweets in the Immersive Explore timeline

\*/

object EnableLaunchVideosInImmersiveExplore

extends FSParam[Boolean](name = "launch\_videos\_in\_immersive\_explore", default = false)

/\*\*

\* Param to enable Ntab Entries for Sports Event Notifications

\*/

object EnableNTabEntriesForSportsEventNotifications

extends FSParam[Boolean](

name = "magicfanout\_sports\_event\_enable\_ntab\_entries",

default = false)

/\*\*

\* Param to enable Ntab Facepiles for teams in Sport Notifs

\*/

object EnableNTabFacePileForSportsEventNotifications

extends FSParam[Boolean](

name = "magicfanout\_sports\_event\_enable\_ntab\_facepiles",

default = false)

/\*\*

\* Param to enable Ntab Override for Sports Event Notifications

\*/

object EnableNTabOverrideForSportsEventNotifications

extends FSParam[Boolean](

name = "magicfanout\_sports\_event\_enable\_ntab\_override",

default = false)

/\*\*

\* Param to control the interval for MF Product Launch Notifs

\*/

object ProductLaunchPushIntervalInHours

extends FSBoundedParam[Duration](

name = "product\_launch\_fatigue\_push\_interval\_in\_hours",

default = 24.hours,

min = Duration.Bottom,

max = Duration.Top)

with HasDurationConversion {

override val durationConversion = DurationConversion.FromHours

}

/\*\*

\* Param to control the maximum number of MF Product Launch Notifs in a period of time

\*/

object ProductLaunchMaxNumberOfPushesInInterval

extends FSBoundedParam[Int](

name = "product\_launch\_fatigue\_max\_pushes\_in\_interval",

default = 1,

min = 0,

max = 10)

/\*\*

\* Param to control the minInterval for fatigue between consecutive MF Product Launch Notifs

\*/

object ProductLaunchMinIntervalFatigue

extends FSBoundedParam[Duration](

name = "product\_launch\_fatigue\_min\_interval\_consecutive\_pushes\_in\_hours",

default = 24.hours,

min = Duration.Bottom,

max = Duration.Top)

with HasDurationConversion {

override val durationConversion = DurationConversion.FromHours

}

/\*\*

\* Param to control the interval for MF New Creator Notifs

\*/

object NewCreatorPushIntervalInHours

extends FSBoundedParam[Duration](

name = "new\_creator\_fatigue\_push\_interval\_in\_hours",

default = 24.hours,

min = Duration.Bottom,

max = Duration.Top)

with HasDurationConversion {

override val durationConversion = DurationConversion.FromHours

}

/\*\*

\* Param to control the maximum number of MF New Creator Notifs in a period of time

\*/

object NewCreatorPushMaxNumberOfPushesInInterval

extends FSBoundedParam[Int](

name = "new\_creator\_fatigue\_max\_pushes\_in\_interval",

default = 1,

min = 0,

max = 10)

/\*\*

\* Param to control the minInterval for fatigue between consecutive MF New Creator Notifs

\*/

object NewCreatorPushMinIntervalFatigue

extends FSBoundedParam[Duration](

name = "new\_creator\_fatigue\_min\_interval\_consecutive\_pushes\_in\_hours",

default = 24.hours,

min = Duration.Bottom,

max = Duration.Top)

with HasDurationConversion {

override val durationConversion = DurationConversion.FromHours

}

/\*\*

\* Param to control the interval for MF New Creator Notifs

\*/

object CreatorSubscriptionPushIntervalInHours

extends FSBoundedParam[Duration](

name = "creator\_subscription\_fatigue\_push\_interval\_in\_hours",

default = 24.hours,

min = Duration.Bottom,

max = Duration.Top)

with HasDurationConversion {

override val durationConversion = DurationConversion.FromHours

}

/\*\*

\* Param to control the maximum number of MF New Creator Notifs in a period of time

\*/

object CreatorSubscriptionPushMaxNumberOfPushesInInterval

extends FSBoundedParam[Int](

name = "creator\_subscription\_fatigue\_max\_pushes\_in\_interval",

default = 1,

min = 0,

max = 10)

/\*\*

\* Param to control the minInterval for fatigue between consecutive MF New Creator Notifs

\*/

object CreatorSubscriptionPushhMinIntervalFatigue

extends FSBoundedParam[Duration](

name = "creator\_subscription\_fatigue\_min\_interval\_consecutive\_pushes\_in\_hours",

default = 24.hours,

min = Duration.Bottom,

max = Duration.Top)

with HasDurationConversion {

override val durationConversion = DurationConversion.FromHours

}

/\*\*

\* Param to define the landing page deeplink of product launch notifications

\*/

object ProductLaunchLandingPageDeepLink

extends FSParam[String](

name = "product\_launch\_landing\_page\_deeplink",

default = ""

)

/\*\*

\* Param to define the tap through of product launch notifications

\*/

object ProductLaunchTapThrough

extends FSParam[String](

name = "product\_launch\_tap\_through",

default = ""

)

/\*\*

\* Param to skip checking isTargetBlueVerified

\*/

object DisableIsTargetBlueVerifiedPredicate

extends FSParam[Boolean](

name = "product\_launch\_disable\_is\_target\_blue\_verified\_predicate",

default = false

)

/\*\*

\* Param to enable Ntab Entries for Sports Event Notifications

\*/

object EnableNTabEntriesForProductLaunchNotifications

extends FSParam[Boolean](name = "product\_launch\_enable\_ntab\_entry", default = true)

/\*\*

\* Param to skip checking isTargetLegacyVerified

\*/

object DisableIsTargetLegacyVerifiedPredicate

extends FSParam[Boolean](

name = "product\_launch\_disable\_is\_target\_legacy\_verified\_predicate",

default = false

)

/\*\*

\* Param to enable checking isTargetSuperFollowCreator

\*/

object EnableIsTargetSuperFollowCreatorPredicate

extends FSParam[Boolean](

name = "product\_launch\_is\_target\_super\_follow\_creator\_predicate\_enabled",

default = false

)

/\*\*

\* Param to enable Spammy Tweet filter

\*/

object EnableSpammyTweetFilter

extends FSParam[Boolean](

name = "health\_signal\_store\_enable\_spammy\_tweet\_filter",

default = false)

/\*\*

\* Param to enable Push to Home Android

\*/

object EnableTweetPushToHomeAndroid

extends FSParam[Boolean](name = "push\_to\_home\_tweet\_recs\_android", default = false)

/\*\*

\* Param to enable Push to Home iOS

\*/

object EnableTweetPushToHomeiOS

extends FSParam[Boolean](name = "push\_to\_home\_tweet\_recs\_iOS", default = false)

/\*\*

\* Param to set Spammy Tweet score threshold for OON candidates

\*/

object SpammyTweetOonThreshold

extends FSBoundedParam[Double](

name = "health\_signal\_store\_spammy\_tweet\_oon\_threshold",

default = 1.1,

min = 0.0,

max = 1.1

)

object NumFollowerThresholdForHealthAndQualityFilters

extends FSBoundedParam[Double](

name = "health\_signal\_store\_num\_follower\_threshold\_for\_health\_and\_quality\_filters",

default = 10000000000.0,

min = 0.0,

max = 10000000000.0

)

object NumFollowerThresholdForHealthAndQualityFiltersPreranking

extends FSBoundedParam[Double](

name =

"health\_signal\_store\_num\_follower\_threshold\_for\_health\_and\_quality\_filters\_preranking",

default = 10000000.0,

min = 0.0,

max = 10000000000.0

)

/\*\*

\* Param to set Spammy Tweet score threshold for IN candidates

\*/

object SpammyTweetInThreshold

extends FSBoundedParam[Double](

name = "health\_signal\_store\_spammy\_tweet\_in\_threshold",

default = 1.1,

min = 0.0,

max = 1.1

)

/\*\*

\* Param to control bucketing for the Spammy Tweet score

\*/

object SpammyTweetBucketingThreshold

extends FSBoundedParam[Double](

name = "health\_signal\_store\_spammy\_tweet\_bucketing\_threshold",

default = 1.0,

min = 0.0,

max = 1.0

)

/\*\*

\* Param to specify the maximum number of Explore Video Tweets to request

\*/

object MaxExploreVideoTweets

extends FSBoundedParam[Int](

name = "explore\_video\_tweets\_max\_candidates",

default = 100,

min = 0,

max = 500

)

/\*\*

\* Param to enable social context feature set

\*/

object EnableBoundedFeatureSetForSocialContext

extends FSParam[Boolean](

name = "feature\_hydration\_user\_social\_context\_bounded\_feature\_set\_enable",

default = true)

/\*\*

\* Param to enable stp user social context feature set

\*/

object EnableStpBoundedFeatureSetForUserSocialContext

extends FSParam[Boolean](

name = "feature\_hydration\_stp\_social\_context\_bounded\_feature\_set\_enable",

default = true)

/\*\*

\* Param to enable core user history social context feature set

\*/

object EnableCoreUserHistoryBoundedFeatureSetForSocialContext

extends FSParam[Boolean](

name = "feature\_hydration\_core\_user\_history\_social\_context\_bounded\_feature\_set\_enable",

default = true)

/\*\*

\* Param to enable skipping post-ranking filters

\*/

object SkipPostRankingFilters

extends FSParam[Boolean](

name = "frigate\_push\_modeling\_skip\_post\_ranking\_filters",

default = false)

object MagicFanoutSimClusterDotProductNonHeavyUserThreshold

extends FSBoundedParam[Double](

name = "frigate\_push\_magicfanout\_simcluster\_non\_heavy\_user\_dot\_product\_threshold",

default = 0.0,

min = 0.0,

max = 100.0

)

object MagicFanoutSimClusterDotProductHeavyUserThreshold

extends FSBoundedParam[Double](

name = "frigate\_push\_magicfanout\_simcluster\_heavy\_user\_dot\_product\_threshold",

default = 10.0,

min = 0.0,

max = 100.0

)

object EnableReducedFatigueRulesForSeeLessOften

extends FSParam[Boolean](

name = "seelessoften\_enable\_reduced\_fatigue",

default = false

)

}