package com.twitter.home\_mixer.product.scored\_tweets

import com.twitter.conversions.DurationOps.\_

import com.twitter.home\_mixer.functional\_component.feature\_hydrator.FeedbackHistoryQueryFeatureHydrator

import com.twitter.home\_mixer.functional\_component.feature\_hydrator.ImpressionBloomFilterQueryFeatureHydrator

import com.twitter.home\_mixer.functional\_component.feature\_hydrator.RealGraphInNetworkScoresQueryFeatureHydrator

import com.twitter.home\_mixer.functional\_component.feature\_hydrator.RequestQueryFeatureHydrator

import com.twitter.home\_mixer.functional\_component.feature\_hydrator.TweetImpressionsQueryFeatureHydrator

import com.twitter.home\_mixer.functional\_component.filter.FeedbackFatigueFilter

import com.twitter.home\_mixer.functional\_component.filter.PreviouslySeenTweetsFilter

import com.twitter.home\_mixer.functional\_component.filter.PreviouslyServedTweetsFilter

import com.twitter.home\_mixer.functional\_component.filter.RejectTweetFromViewerFilter

import com.twitter.home\_mixer.functional\_component.filter.RetweetDeduplicationFilter

import com.twitter.home\_mixer.functional\_component.side\_effect.PublishClientSentImpressionsEventBusSideEffect

import com.twitter.home\_mixer.functional\_component.side\_effect.PublishClientSentImpressionsManhattanSideEffect

import com.twitter.home\_mixer.functional\_component.side\_effect.PublishImpressionBloomFilterSideEffect

import com.twitter.home\_mixer.functional\_component.side\_effect.UpdateLastNonPollingTimeSideEffect

import com.twitter.home\_mixer.model.HomeFeatures.ExclusiveConversationAuthorIdFeature

import com.twitter.home\_mixer.model.HomeFeatures.InNetworkFeature

import com.twitter.home\_mixer.model.HomeFeatures.InReplyToTweetIdFeature

import com.twitter.home\_mixer.model.HomeFeatures.IsSupportAccountReplyFeature

import com.twitter.home\_mixer.model.HomeFeatures.ScoreFeature

import com.twitter.home\_mixer.param.HomeGlobalParams.EnableImpressionBloomFilter

import com.twitter.home\_mixer.param.HomeMixerFlagName.TargetFetchLatency

import com.twitter.home\_mixer.param.HomeMixerFlagName.TargetScoringLatency

import com.twitter.home\_mixer.product.scored\_tweets.candidate\_pipeline.CachedScoredTweetsCandidatePipelineConfig

import com.twitter.home\_mixer.product.scored\_tweets.candidate\_pipeline.ScoredTweetsBackfillCandidatePipelineConfig

import com.twitter.home\_mixer.product.scored\_tweets.candidate\_pipeline.ScoredTweetsFrsCandidatePipelineConfig

import com.twitter.home\_mixer.product.scored\_tweets.candidate\_pipeline.ScoredTweetsInNetworkCandidatePipelineConfig

import com.twitter.home\_mixer.product.scored\_tweets.candidate\_pipeline.ScoredTweetsListsCandidatePipelineConfig

import com.twitter.home\_mixer.product.scored\_tweets.candidate\_pipeline.ScoredTweetsPopularVideosCandidatePipelineConfig

import com.twitter.home\_mixer.product.scored\_tweets.candidate\_pipeline.ScoredTweetsTweetMixerCandidatePipelineConfig

import com.twitter.home\_mixer.product.scored\_tweets.candidate\_pipeline.ScoredTweetsUtegCandidatePipelineConfig

import com.twitter.home\_mixer.product.scored\_tweets.feature\_hydrator.CachedScoredTweetsQueryFeatureHydrator

import com.twitter.home\_mixer.product.scored\_tweets.feature\_hydrator.ListIdsQueryFeatureHydrator

import com.twitter.home\_mixer.product.scored\_tweets.feature\_hydrator.RealGraphQueryFeatureHydrator

import com.twitter.home\_mixer.product.scored\_tweets.feature\_hydrator.RealTimeInteractionGraphUserVertexQueryFeatureHydrator

import com.twitter.home\_mixer.product.scored\_tweets.feature\_hydrator.RequestTimeQueryFeatureHydrator

import com.twitter.home\_mixer.product.scored\_tweets.feature\_hydrator.TwhinUserEngagementQueryFeatureHydrator

import com.twitter.home\_mixer.product.scored\_tweets.feature\_hydrator.TwhinUserFollowQueryFeatureHydrator

import com.twitter.home\_mixer.product.scored\_tweets.feature\_hydrator.UserLanguagesFeatureHydrator

import com.twitter.home\_mixer.product.scored\_tweets.feature\_hydrator.UserStateQueryFeatureHydrator

import com.twitter.home\_mixer.product.scored\_tweets.feature\_hydrator.offline\_aggregates.PartAAggregateQueryFeatureHydrator

import com.twitter.home\_mixer.product.scored\_tweets.feature\_hydrator.offline\_aggregates.PartBAggregateQueryFeatureHydrator

import com.twitter.home\_mixer.product.scored\_tweets.feature\_hydrator.real\_time\_aggregates.UserEngagementRealTimeAggregatesFeatureHydrator

import com.twitter.home\_mixer.product.scored\_tweets.filter.DuplicateConversationTweetsFilter

import com.twitter.home\_mixer.product.scored\_tweets.filter.OutOfNetworkCompetitorFilter

import com.twitter.home\_mixer.product.scored\_tweets.filter.OutOfNetworkCompetitorURLFilter

import com.twitter.home\_mixer.product.scored\_tweets.filter.ScoredTweetsSocialContextFilter

import com.twitter.home\_mixer.product.scored\_tweets.marshaller.ScoredTweetsResponseDomainMarshaller

import com.twitter.home\_mixer.product.scored\_tweets.marshaller.ScoredTweetsResponseTransportMarshaller

import com.twitter.home\_mixer.product.scored\_tweets.model.ScoredTweetsQuery

import com.twitter.home\_mixer.product.scored\_tweets.model.ScoredTweetsResponse

import com.twitter.home\_mixer.product.scored\_tweets.param.ScoredTweetsParam.MaxInNetworkResultsParam

import com.twitter.home\_mixer.product.scored\_tweets.param.ScoredTweetsParam.MaxOutOfNetworkResultsParam

import com.twitter.home\_mixer.product.scored\_tweets.scoring\_pipeline.ScoredTweetsHeuristicScoringPipelineConfig

import com.twitter.home\_mixer.product.scored\_tweets.scoring\_pipeline.ScoredTweetsModelScoringPipelineConfig

import com.twitter.home\_mixer.product.scored\_tweets.selector.KeepBestOutOfNetworkCandidatePerAuthorPerSuggestType

import com.twitter.home\_mixer.product.scored\_tweets.side\_effect.CachedScoredTweetsSideEffect

import com.twitter.home\_mixer.product.scored\_tweets.side\_effect.ScribeScoredCandidatesSideEffect

import com.twitter.home\_mixer.product.scored\_tweets.side\_effect.ScribeServedCommonFeaturesAndCandidateFeaturesSideEffect

import com.twitter.home\_mixer.{thriftscala => t}

import com.twitter.inject.annotations.Flag

import com.twitter.product\_mixer.component\_library.feature\_hydrator.query.async.AsyncQueryFeatureHydrator

import com.twitter.product\_mixer.component\_library.feature\_hydrator.query.impressed\_tweets.ImpressedTweetsQueryFeatureHydrator

import com.twitter.product\_mixer.component\_library.feature\_hydrator.query.param\_gated.ParamGatedQueryFeatureHydrator

import com.twitter.product\_mixer.component\_library.feature\_hydrator.query.social\_graph.SGSFollowedUsersQueryFeatureHydrator

import com.twitter.product\_mixer.component\_library.filter.PredicateFeatureFilter

import com.twitter.product\_mixer.component\_library.model.candidate.TweetCandidate

import com.twitter.product\_mixer.component\_library.selector.DropDuplicateCandidates

import com.twitter.product\_mixer.component\_library.selector.DropFilteredMaxCandidates

import com.twitter.product\_mixer.component\_library.selector.DropMaxCandidates

import com.twitter.product\_mixer.component\_library.selector.IdAndClassDuplicationKey

import com.twitter.product\_mixer.component\_library.selector.InsertAppendResults

import com.twitter.product\_mixer.component\_library.selector.PickFirstCandidateMerger

import com.twitter.product\_mixer.component\_library.selector.UpdateSortCandidates

import com.twitter.product\_mixer.component\_library.selector.sorter.FeatureValueSorter

import com.twitter.product\_mixer.core.functional\_component.common.AllExceptPipelines

import com.twitter.product\_mixer.core.functional\_component.common.AllPipelines

import com.twitter.product\_mixer.core.functional\_component.configapi.StaticParam

import com.twitter.product\_mixer.core.functional\_component.feature\_hydrator.QueryFeatureHydrator

import com.twitter.product\_mixer.core.functional\_component.filter.Filter

import com.twitter.product\_mixer.core.functional\_component.marshaller.TransportMarshaller

import com.twitter.product\_mixer.core.functional\_component.premarshaller.DomainMarshaller

import com.twitter.product\_mixer.core.functional\_component.selector.Selector

import com.twitter.product\_mixer.core.functional\_component.side\_effect.PipelineResultSideEffect

import com.twitter.product\_mixer.core.model.common.identifier.CandidatePipelineIdentifier

import com.twitter.product\_mixer.core.model.common.identifier.ComponentIdentifier

import com.twitter.product\_mixer.core.model.common.identifier.FilterIdentifier

import com.twitter.product\_mixer.core.model.common.identifier.RecommendationPipelineIdentifier

import com.twitter.product\_mixer.core.model.common.identifier.ScoringPipelineIdentifier

import com.twitter.product\_mixer.core.model.common.presentation.ItemCandidateWithDetails

import com.twitter.product\_mixer.core.pipeline.FailOpenPolicy

import com.twitter.product\_mixer.core.pipeline.candidate.CandidatePipelineConfig

import com.twitter.product\_mixer.core.pipeline.recommendation.RecommendationPipelineConfig

import com.twitter.product\_mixer.core.pipeline.scoring.ScoringPipelineConfig

import com.twitter.product\_mixer.core.quality\_factor.BoundsWithDefault

import com.twitter.product\_mixer.core.quality\_factor.LinearLatencyQualityFactorConfig

import com.twitter.product\_mixer.core.quality\_factor.QualityFactorConfig

import com.twitter.util.Duration

import javax.inject.Inject

import javax.inject.Singleton

@Singleton

class ScoredTweetsRecommendationPipelineConfig @Inject() (

scoredTweetsInNetworkCandidatePipelineConfig: ScoredTweetsInNetworkCandidatePipelineConfig,

scoredTweetsUtegCandidatePipelineConfig: ScoredTweetsUtegCandidatePipelineConfig,

scoredTweetsTweetMixerCandidatePipelineConfig: ScoredTweetsTweetMixerCandidatePipelineConfig,

scoredTweetsFrsCandidatePipelineConfig: ScoredTweetsFrsCandidatePipelineConfig,

scoredTweetsListsCandidatePipelineConfig: ScoredTweetsListsCandidatePipelineConfig,

scoredTweetsPopularVideosCandidatePipelineConfig: ScoredTweetsPopularVideosCandidatePipelineConfig,

scoredTweetsBackfillCandidatePipelineConfig: ScoredTweetsBackfillCandidatePipelineConfig,

cachedScoredTweetsCandidatePipelineConfig: CachedScoredTweetsCandidatePipelineConfig,

requestQueryFeatureHydrator: RequestQueryFeatureHydrator[ScoredTweetsQuery],

requestTimeQueryFeatureHydrator: RequestTimeQueryFeatureHydrator,

realTimeInteractionGraphUserVertexQueryFeatureHydrator: RealTimeInteractionGraphUserVertexQueryFeatureHydrator,

userStateQueryFeatureHydrator: UserStateQueryFeatureHydrator,

userEngagementRealTimeAggregatesFeatureHydrator: UserEngagementRealTimeAggregatesFeatureHydrator,

twhinUserEngagementQueryFeatureHydrator: TwhinUserEngagementQueryFeatureHydrator,

twhinUserFollowQueryFeatureHydrator: TwhinUserFollowQueryFeatureHydrator,

cachedScoredTweetsQueryFeatureHydrator: CachedScoredTweetsQueryFeatureHydrator,

sgsFollowedUsersQueryFeatureHydrator: SGSFollowedUsersQueryFeatureHydrator,

scoredTweetsModelScoringPipelineConfig: ScoredTweetsModelScoringPipelineConfig,

impressionBloomFilterQueryFeatureHydrator: ImpressionBloomFilterQueryFeatureHydrator[

ScoredTweetsQuery

],

manhattanTweetImpressionsQueryFeatureHydrator: TweetImpressionsQueryFeatureHydrator[

ScoredTweetsQuery

],

memcacheTweetImpressionsQueryFeatureHydrator: ImpressedTweetsQueryFeatureHydrator,

listIdsQueryFeatureHydrator: ListIdsQueryFeatureHydrator,

feedbackHistoryQueryFeatureHydrator: FeedbackHistoryQueryFeatureHydrator,

publishClientSentImpressionsEventBusSideEffect: PublishClientSentImpressionsEventBusSideEffect,

publishClientSentImpressionsManhattanSideEffect: PublishClientSentImpressionsManhattanSideEffect,

publishImpressionBloomFilterSideEffect: PublishImpressionBloomFilterSideEffect,

realGraphInNetworkScoresQueryFeatureHydrator: RealGraphInNetworkScoresQueryFeatureHydrator,

realGraphQueryFeatureHydrator: RealGraphQueryFeatureHydrator,

userLanguagesFeatureHydrator: UserLanguagesFeatureHydrator,

partAAggregateQueryFeatureHydrator: PartAAggregateQueryFeatureHydrator,

partBAggregateQueryFeatureHydrator: PartBAggregateQueryFeatureHydrator,

cachedScoredTweetsSideEffect: CachedScoredTweetsSideEffect,

scribeScoredCandidatesSideEffect: ScribeScoredCandidatesSideEffect,

scribeServedCommonFeaturesAndCandidateFeaturesSideEffect: ScribeServedCommonFeaturesAndCandidateFeaturesSideEffect,

updateLastNonPollingTimeSideEffect: UpdateLastNonPollingTimeSideEffect[

ScoredTweetsQuery,

ScoredTweetsResponse

],

@Flag(TargetFetchLatency) targetFetchLatency: Duration,

@Flag(TargetScoringLatency) targetScoringLatency: Duration)

extends RecommendationPipelineConfig[

ScoredTweetsQuery,

TweetCandidate,

ScoredTweetsResponse,

t.ScoredTweetsResponse

] {

override val identifier: RecommendationPipelineIdentifier =

RecommendationPipelineIdentifier("ScoredTweets")

private val SubscriptionReplyFilterId = "SubscriptionReply"

private val MaxBackfillTweets = 50

private val scoringStep = RecommendationPipelineConfig.scoringPipelinesStep

override val fetchQueryFeatures: Seq[QueryFeatureHydrator[ScoredTweetsQuery]] = Seq(

requestQueryFeatureHydrator,

realGraphInNetworkScoresQueryFeatureHydrator,

cachedScoredTweetsQueryFeatureHydrator,

sgsFollowedUsersQueryFeatureHydrator,

ParamGatedQueryFeatureHydrator(

EnableImpressionBloomFilter,

impressionBloomFilterQueryFeatureHydrator

),

manhattanTweetImpressionsQueryFeatureHydrator,

memcacheTweetImpressionsQueryFeatureHydrator,

listIdsQueryFeatureHydrator,

userStateQueryFeatureHydrator,

AsyncQueryFeatureHydrator(scoringStep, feedbackHistoryQueryFeatureHydrator),

AsyncQueryFeatureHydrator(scoringStep, realGraphQueryFeatureHydrator),

AsyncQueryFeatureHydrator(scoringStep, requestTimeQueryFeatureHydrator),

AsyncQueryFeatureHydrator(scoringStep, userLanguagesFeatureHydrator),

AsyncQueryFeatureHydrator(scoringStep, userEngagementRealTimeAggregatesFeatureHydrator),

AsyncQueryFeatureHydrator(scoringStep, realTimeInteractionGraphUserVertexQueryFeatureHydrator),

AsyncQueryFeatureHydrator(scoringStep, twhinUserFollowQueryFeatureHydrator),

AsyncQueryFeatureHydrator(scoringStep, twhinUserEngagementQueryFeatureHydrator),

AsyncQueryFeatureHydrator(scoringStep, partAAggregateQueryFeatureHydrator),

AsyncQueryFeatureHydrator(scoringStep, partBAggregateQueryFeatureHydrator),

)

override val candidatePipelines: Seq[

CandidatePipelineConfig[ScoredTweetsQuery, \_, \_, TweetCandidate]

] = Seq(

cachedScoredTweetsCandidatePipelineConfig,

scoredTweetsInNetworkCandidatePipelineConfig,

scoredTweetsUtegCandidatePipelineConfig,

scoredTweetsTweetMixerCandidatePipelineConfig,

scoredTweetsFrsCandidatePipelineConfig,

scoredTweetsListsCandidatePipelineConfig,

scoredTweetsPopularVideosCandidatePipelineConfig,

scoredTweetsBackfillCandidatePipelineConfig

)

override val postCandidatePipelinesSelectors: Seq[Selector[ScoredTweetsQuery]] = Seq(

DropDuplicateCandidates(

pipelineScope = AllPipelines,

duplicationKey = IdAndClassDuplicationKey,

mergeStrategy = PickFirstCandidateMerger

),

InsertAppendResults(AllPipelines)

)

override val globalFilters: Seq[Filter[ScoredTweetsQuery, TweetCandidate]] = Seq(

// sort these to have the "cheaper" filters run first

RejectTweetFromViewerFilter,

RetweetDeduplicationFilter,

PreviouslySeenTweetsFilter,

PreviouslyServedTweetsFilter,

PredicateFeatureFilter.fromPredicate(

FilterIdentifier(SubscriptionReplyFilterId),

shouldKeepCandidate = { features =>

features.getOrElse(InReplyToTweetIdFeature, None).isEmpty ||

features.getOrElse(ExclusiveConversationAuthorIdFeature, None).isEmpty

}

),

FeedbackFatigueFilter

)

override val candidatePipelineFailOpenPolicies: Map[CandidatePipelineIdentifier, FailOpenPolicy] =

Map(

cachedScoredTweetsCandidatePipelineConfig.identifier -> FailOpenPolicy.Always,

scoredTweetsInNetworkCandidatePipelineConfig.identifier -> FailOpenPolicy.Always,

scoredTweetsUtegCandidatePipelineConfig.identifier -> FailOpenPolicy.Always,

scoredTweetsTweetMixerCandidatePipelineConfig.identifier -> FailOpenPolicy.Always,

scoredTweetsFrsCandidatePipelineConfig.identifier -> FailOpenPolicy.Always,

scoredTweetsListsCandidatePipelineConfig.identifier -> FailOpenPolicy.Always,

scoredTweetsPopularVideosCandidatePipelineConfig.identifier -> FailOpenPolicy.Always,

scoredTweetsBackfillCandidatePipelineConfig.identifier -> FailOpenPolicy.Always

)

override val scoringPipelineFailOpenPolicies: Map[ScoringPipelineIdentifier, FailOpenPolicy] =

Map(

ScoredTweetsHeuristicScoringPipelineConfig.identifier -> FailOpenPolicy.Always

)

private val candidatePipelineQualityFactorConfig = LinearLatencyQualityFactorConfig(

qualityFactorBounds = BoundsWithDefault(minInclusive = 0.1, maxInclusive = 1.0, default = 0.95),

initialDelay = 60.seconds,

targetLatency = targetFetchLatency,

targetLatencyPercentile = 95.0,

delta = 0.00125

)

private val scoringPipelineQualityFactorConfig =

candidatePipelineQualityFactorConfig.copy(targetLatency = targetScoringLatency)

override val qualityFactorConfigs: Map[ComponentIdentifier, QualityFactorConfig] = Map(

// candidate pipelines

scoredTweetsInNetworkCandidatePipelineConfig.identifier -> candidatePipelineQualityFactorConfig,

scoredTweetsUtegCandidatePipelineConfig.identifier -> candidatePipelineQualityFactorConfig,

scoredTweetsTweetMixerCandidatePipelineConfig.identifier -> candidatePipelineQualityFactorConfig,

scoredTweetsFrsCandidatePipelineConfig.identifier -> candidatePipelineQualityFactorConfig,

scoredTweetsListsCandidatePipelineConfig.identifier -> candidatePipelineQualityFactorConfig,

scoredTweetsPopularVideosCandidatePipelineConfig.identifier -> candidatePipelineQualityFactorConfig,

scoredTweetsBackfillCandidatePipelineConfig.identifier -> candidatePipelineQualityFactorConfig,

// scoring pipelines

scoredTweetsModelScoringPipelineConfig.identifier -> scoringPipelineQualityFactorConfig,

)

override val scoringPipelines: Seq[ScoringPipelineConfig[ScoredTweetsQuery, TweetCandidate]] =

Seq(

// scoring pipeline - run on non-cached candidates only since cached ones are already scored

scoredTweetsModelScoringPipelineConfig,

// re-scoring pipeline - run on all candidates since these are request specific

ScoredTweetsHeuristicScoringPipelineConfig

)

override val postScoringFilters = Seq(

ScoredTweetsSocialContextFilter,

OutOfNetworkCompetitorFilter,

OutOfNetworkCompetitorURLFilter,

DuplicateConversationTweetsFilter,

PredicateFeatureFilter.fromPredicate(

FilterIdentifier("IsSupportAccountReply"),

shouldKeepCandidate = { features =>

!features.getOrElse(IsSupportAccountReplyFeature, false)

})

)

override val resultSelectors: Seq[Selector[ScoredTweetsQuery]] = Seq(

KeepBestOutOfNetworkCandidatePerAuthorPerSuggestType(AllPipelines),

UpdateSortCandidates(AllPipelines, FeatureValueSorter.descending(ScoreFeature)),

DropFilteredMaxCandidates(

pipelineScope =

AllExceptPipelines(Set(scoredTweetsBackfillCandidatePipelineConfig.identifier)),

filter = {

case ItemCandidateWithDetails(\_, \_, features) =>

features.getOrElse(InNetworkFeature, false)

case \_ => false

},

maxSelectionsParam = MaxInNetworkResultsParam

),

DropFilteredMaxCandidates(

pipelineScope = AllPipelines,

filter = {

case ItemCandidateWithDetails(\_, \_, features) =>

!features.getOrElse(InNetworkFeature, false)

case \_ => false

},

maxSelectionsParam = MaxOutOfNetworkResultsParam

),

DropMaxCandidates(

candidatePipeline = scoredTweetsBackfillCandidatePipelineConfig.identifier,

maxSelectionsParam = StaticParam(MaxBackfillTweets)

),

InsertAppendResults(AllPipelines)

)

override val resultSideEffects: Seq[

PipelineResultSideEffect[ScoredTweetsQuery, ScoredTweetsResponse]

] = Seq(

cachedScoredTweetsSideEffect,

publishClientSentImpressionsEventBusSideEffect,

publishClientSentImpressionsManhattanSideEffect,

publishImpressionBloomFilterSideEffect,

scribeScoredCandidatesSideEffect,

scribeServedCommonFeaturesAndCandidateFeaturesSideEffect,

updateLastNonPollingTimeSideEffect

)

override val domainMarshaller: DomainMarshaller[

ScoredTweetsQuery,

ScoredTweetsResponse

] = ScoredTweetsResponseDomainMarshaller

override val transportMarshaller: TransportMarshaller[

ScoredTweetsResponse,

t.ScoredTweetsResponse

] = ScoredTweetsResponseTransportMarshaller

}