package com.twitter.home\_mixer.model

import com.twitter.core\_workflows.user\_model.{thriftscala => um}

import com.twitter.dal.personal\_data.{thriftjava => pd}

import com.twitter.gizmoduck.{thriftscala => gt}

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

import com.twitter.ml.api.constant.SharedFeatures

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

import com.twitter.product\_mixer.core.feature.Feature

import com.twitter.product\_mixer.core.feature.FeatureWithDefaultOnFailure

import com.twitter.product\_mixer.core.feature.datarecord.BoolDataRecordCompatible

import com.twitter.product\_mixer.core.feature.datarecord.DataRecordFeature

import com.twitter.product\_mixer.core.feature.datarecord.DataRecordOptionalFeature

import com.twitter.product\_mixer.core.feature.datarecord.DoubleDataRecordCompatible

import com.twitter.product\_mixer.core.feature.datarecord.LongDiscreteDataRecordCompatible

import com.twitter.product\_mixer.core.model.marshalling.response.urt.metadata.TopicContextFunctionalityType

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

import com.twitter.search.common.features.{thriftscala => sc}

import com.twitter.search.earlybird.{thriftscala => eb}

import com.twitter.timelinemixer.clients.manhattan.DismissInfo

import com.twitter.timelinemixer.clients.persistence.TimelineResponseV3

import com.twitter.timelinemixer.injection.model.candidate.AudioSpaceMetaData

import com.twitter.timelines.conversation\_features.v1.thriftscala.ConversationFeatures

import com.twitter.timelines.impression.{thriftscala => imp}

import com.twitter.timelines.impressionbloomfilter.{thriftscala => blm}

import com.twitter.timelines.model.UserId

import com.twitter.timelines.prediction.features.common.TimelinesSharedFeatures

import com.twitter.timelines.prediction.features.engagement\_features.EngagementDataRecordFeatures

import com.twitter.timelines.prediction.features.recap.RecapFeatures

import com.twitter.timelines.prediction.features.request\_context.RequestContextFeatures

import com.twitter.timelines.service.{thriftscala => tst}

import com.twitter.timelineservice.model.FeedbackEntry

import com.twitter.timelineservice.suggests.logging.candidate\_tweet\_source\_id.{thriftscala => cts}

import com.twitter.timelineservice.suggests.{thriftscala => st}

import com.twitter.tsp.{thriftscala => tsp}

import com.twitter.tweetconvosvc.tweet\_ancestor.{thriftscala => ta}

import com.twitter.util.Time

object HomeFeatures {

// Candidate Features

object AncestorsFeature extends Feature[TweetCandidate, Seq[ta.TweetAncestor]]

object AudioSpaceMetaDataFeature extends Feature[TweetCandidate, Option[AudioSpaceMetaData]]

object TwitterListIdFeature extends Feature[TweetCandidate, Option[Long]]

/\*\*

\* For Retweets, this should refer to the retweeting user. Use [[SourceUserIdFeature]] if you want to know

\* who created the Tweet that was retweeted.

\*/

object AuthorIdFeature

extends DataRecordOptionalFeature[TweetCandidate, Long]

with LongDiscreteDataRecordCompatible {

override val featureName: String = SharedFeatures.AUTHOR\_ID.getFeatureName

override val personalDataTypes: Set[pd.PersonalDataType] = Set(pd.PersonalDataType.UserId)

}

object AuthorIsBlueVerifiedFeature extends Feature[TweetCandidate, Boolean]

object AuthorIsGoldVerifiedFeature extends Feature[TweetCandidate, Boolean]

object AuthorIsGrayVerifiedFeature extends Feature[TweetCandidate, Boolean]

object AuthorIsLegacyVerifiedFeature extends Feature[TweetCandidate, Boolean]

object AuthorIsCreatorFeature extends Feature[TweetCandidate, Boolean]

object AuthorIsProtectedFeature extends Feature[TweetCandidate, Boolean]

object AuthoredByContextualUserFeature extends Feature[TweetCandidate, Boolean]

object CachedCandidatePipelineIdentifierFeature extends Feature[TweetCandidate, Option[String]]

object CandidateSourceIdFeature

extends Feature[TweetCandidate, Option[cts.CandidateTweetSourceId]]

object ConversationFeature extends Feature[TweetCandidate, Option[ConversationFeatures]]

/\*\*

\* This field should be set to the focal Tweet's tweetId for all tweets which are expected to

\* be rendered in the same convo module. For non-convo module Tweets, this will be

\* set to None. Note this is different from how TweetyPie defines ConversationId which is defined

\* on all Tweets and points to the root tweet. This feature is used for grouping convo modules together.

\*/

object ConversationModuleFocalTweetIdFeature extends Feature[TweetCandidate, Option[Long]]

/\*\*

\* This field should always be set to the root Tweet in a conversation for all Tweets. For replies, this will

\* point back to the root Tweet. For non-replies, this will be the candidate's Tweet id. This is consistent with

\* the TweetyPie definition of ConversationModuleId.

\*/

object ConversationModuleIdFeature extends Feature[TweetCandidate, Option[Long]]

object DirectedAtUserIdFeature extends Feature[TweetCandidate, Option[Long]]

object EarlybirdFeature extends Feature[TweetCandidate, Option[sc.ThriftTweetFeatures]]

object EarlybirdScoreFeature extends Feature[TweetCandidate, Option[Double]]

object EarlybirdSearchResultFeature extends Feature[TweetCandidate, Option[eb.ThriftSearchResult]]

object EntityTokenFeature extends Feature[TweetCandidate, Option[String]]

object ExclusiveConversationAuthorIdFeature extends Feature[TweetCandidate, Option[Long]]

object FavoritedByCountFeature

extends DataRecordFeature[TweetCandidate, Double]

with DoubleDataRecordCompatible {

override val featureName: String =

EngagementDataRecordFeatures.InNetworkFavoritesCount.getFeatureName

override val personalDataTypes: Set[pd.PersonalDataType] =

Set(pd.PersonalDataType.CountOfPrivateLikes, pd.PersonalDataType.CountOfPublicLikes)

}

object FavoritedByUserIdsFeature extends Feature[TweetCandidate, Seq[Long]]

object FeedbackHistoryFeature extends Feature[TweetCandidate, Seq[FeedbackEntry]]

object RetweetedByCountFeature

extends DataRecordFeature[TweetCandidate, Double]

with DoubleDataRecordCompatible {

override val featureName: String =

EngagementDataRecordFeatures.InNetworkRetweetsCount.getFeatureName

override val personalDataTypes: Set[pd.PersonalDataType] =

Set(pd.PersonalDataType.CountOfPrivateRetweets, pd.PersonalDataType.CountOfPublicRetweets)

}

object RetweetedByEngagerIdsFeature extends Feature[TweetCandidate, Seq[Long]]

object RepliedByCountFeature

extends DataRecordFeature[TweetCandidate, Double]

with DoubleDataRecordCompatible {

override val featureName: String =

EngagementDataRecordFeatures.InNetworkRepliesCount.getFeatureName

override val personalDataTypes: Set[pd.PersonalDataType] =

Set(pd.PersonalDataType.CountOfPrivateReplies, pd.PersonalDataType.CountOfPublicReplies)

}

object RepliedByEngagerIdsFeature extends Feature[TweetCandidate, Seq[Long]]

object FollowedByUserIdsFeature extends Feature[TweetCandidate, Seq[Long]]

object TopicIdSocialContextFeature extends Feature[TweetCandidate, Option[Long]]

object TopicContextFunctionalityTypeFeature

extends Feature[TweetCandidate, Option[TopicContextFunctionalityType]]

object FromInNetworkSourceFeature extends Feature[TweetCandidate, Boolean]

object FullScoringSucceededFeature extends Feature[TweetCandidate, Boolean]

object HasDisplayedTextFeature extends Feature[TweetCandidate, Boolean]

object InReplyToTweetIdFeature extends Feature[TweetCandidate, Option[Long]]

object InReplyToUserIdFeature extends Feature[TweetCandidate, Option[Long]]

object IsAncestorCandidateFeature extends Feature[TweetCandidate, Boolean]

object IsExtendedReplyFeature

extends DataRecordFeature[TweetCandidate, Boolean]

with BoolDataRecordCompatible {

override val featureName: String = RecapFeatures.IS\_EXTENDED\_REPLY.getFeatureName

override val personalDataTypes: Set[pd.PersonalDataType] = Set.empty

}

object IsRandomTweetFeature

extends DataRecordFeature[TweetCandidate, Boolean]

with BoolDataRecordCompatible {

override val featureName: String = TimelinesSharedFeatures.IS\_RANDOM\_TWEET.getFeatureName

override val personalDataTypes: Set[pd.PersonalDataType] = Set.empty

}

object IsReadFromCacheFeature extends Feature[TweetCandidate, Boolean]

object IsRetweetFeature extends Feature[TweetCandidate, Boolean]

object IsRetweetedReplyFeature extends Feature[TweetCandidate, Boolean]

object IsSupportAccountReplyFeature extends Feature[TweetCandidate, Boolean]

object LastScoredTimestampMsFeature extends Feature[TweetCandidate, Option[Long]]

object NonSelfFavoritedByUserIdsFeature extends Feature[TweetCandidate, Seq[Long]]

object NumImagesFeature extends Feature[TweetCandidate, Option[Int]]

object OriginalTweetCreationTimeFromSnowflakeFeature extends Feature[TweetCandidate, Option[Time]]

object PositionFeature extends Feature[TweetCandidate, Option[Int]]

// Internal id generated per prediction service request

object PredictionRequestIdFeature extends Feature[TweetCandidate, Option[Long]]

object QuotedTweetIdFeature extends Feature[TweetCandidate, Option[Long]]

object QuotedUserIdFeature extends Feature[TweetCandidate, Option[Long]]

object ScoreFeature extends Feature[TweetCandidate, Option[Double]]

object SemanticCoreIdFeature extends Feature[TweetCandidate, Option[Long]]

// Key for kafka logging

object ServedIdFeature extends Feature[TweetCandidate, Option[Long]]

object SimclustersTweetTopKClustersWithScoresFeature

extends Feature[TweetCandidate, Map[String, Double]]

object SocialContextFeature extends Feature[TweetCandidate, Option[tst.SocialContext]]

object SourceTweetIdFeature

extends DataRecordOptionalFeature[TweetCandidate, Long]

with LongDiscreteDataRecordCompatible {

override val featureName: String = TimelinesSharedFeatures.SOURCE\_TWEET\_ID.getFeatureName

override val personalDataTypes: Set[pd.PersonalDataType] = Set(pd.PersonalDataType.TweetId)

}

object SourceUserIdFeature extends Feature[TweetCandidate, Option[Long]]

object StreamToKafkaFeature extends Feature[TweetCandidate, Boolean]

object SuggestTypeFeature extends Feature[TweetCandidate, Option[st.SuggestType]]

object TSPMetricTagFeature extends Feature[TweetCandidate, Set[tsp.MetricTag]]

object TweetLanguageFeature extends Feature[TweetCandidate, Option[String]]

object TweetUrlsFeature extends Feature[TweetCandidate, Seq[String]]

object VideoDurationMsFeature extends Feature[TweetCandidate, Option[Int]]

object ViewerIdFeature

extends DataRecordFeature[TweetCandidate, Long]

with LongDiscreteDataRecordCompatible {

override def featureName: String = SharedFeatures.USER\_ID.getFeatureName

override def personalDataTypes: Set[pd.PersonalDataType] = Set(pd.PersonalDataType.UserId)

}

object WeightedModelScoreFeature extends Feature[TweetCandidate, Option[Double]]

object MentionUserIdFeature extends Feature[TweetCandidate, Seq[Long]]

object MentionScreenNameFeature extends Feature[TweetCandidate, Seq[String]]

object HasImageFeature extends Feature[TweetCandidate, Boolean]

object HasVideoFeature extends Feature[TweetCandidate, Boolean]

// Tweetypie VF Features

object IsHydratedFeature extends Feature[TweetCandidate, Boolean]

object IsNsfwFeature extends Feature[TweetCandidate, Boolean]

object QuotedTweetDroppedFeature extends Feature[TweetCandidate, Boolean]

// Raw Tweet Text from Tweetypie

object TweetTextFeature extends Feature[TweetCandidate, Option[String]]

object AuthorEnabledPreviewsFeature extends Feature[TweetCandidate, Boolean]

object IsTweetPreviewFeature extends Feature[TweetCandidate, Boolean]

// SGS Features

/\*\*

\* By convention, this is set to true for retweets of non-followed authors

\* E.g. where somebody the viewer follows retweets a Tweet from somebody the viewer doesn't follow

\*/

object InNetworkFeature extends FeatureWithDefaultOnFailure[TweetCandidate, Boolean] {

override val defaultValue: Boolean = true

}

// Query Features

object AccountAgeFeature extends Feature[PipelineQuery, Option[Time]]

object ClientIdFeature

extends DataRecordOptionalFeature[PipelineQuery, Long]

with LongDiscreteDataRecordCompatible {

override def featureName: String = SharedFeatures.CLIENT\_ID.getFeatureName

override def personalDataTypes: Set[pd.PersonalDataType] = Set(pd.PersonalDataType.ClientType)

}

object CachedScoredTweetsFeature extends Feature[PipelineQuery, Seq[hmt.ScoredTweet]]

object DeviceLanguageFeature extends Feature[PipelineQuery, Option[String]]

object DismissInfoFeature

extends FeatureWithDefaultOnFailure[PipelineQuery, Map[st.SuggestType, Option[DismissInfo]]] {

override def defaultValue: Map[st.SuggestType, Option[DismissInfo]] = Map.empty

}

object FollowingLastNonPollingTimeFeature extends Feature[PipelineQuery, Option[Time]]

object GetInitialFeature

extends DataRecordFeature[PipelineQuery, Boolean]

with BoolDataRecordCompatible {

override def featureName: String = RequestContextFeatures.IS\_GET\_INITIAL.getFeatureName

override def personalDataTypes: Set[pd.PersonalDataType] = Set.empty

}

object GetMiddleFeature

extends DataRecordFeature[PipelineQuery, Boolean]

with BoolDataRecordCompatible {

override def featureName: String = RequestContextFeatures.IS\_GET\_MIDDLE.getFeatureName

override def personalDataTypes: Set[pd.PersonalDataType] = Set.empty

}

object GetNewerFeature

extends DataRecordFeature[PipelineQuery, Boolean]

with BoolDataRecordCompatible {

override def featureName: String = RequestContextFeatures.IS\_GET\_NEWER.getFeatureName

override def personalDataTypes: Set[pd.PersonalDataType] = Set.empty

}

object GetOlderFeature

extends DataRecordFeature[PipelineQuery, Boolean]

with BoolDataRecordCompatible {

override def featureName: String = RequestContextFeatures.IS\_GET\_OLDER.getFeatureName

override def personalDataTypes: Set[pd.PersonalDataType] = Set.empty

}

object GuestIdFeature

extends DataRecordOptionalFeature[PipelineQuery, Long]

with LongDiscreteDataRecordCompatible {

override def featureName: String = SharedFeatures.GUEST\_ID.getFeatureName

override def personalDataTypes: Set[pd.PersonalDataType] = Set(pd.PersonalDataType.GuestId)

}

object HasDarkRequestFeature extends Feature[PipelineQuery, Option[Boolean]]

object ImpressionBloomFilterFeature

extends FeatureWithDefaultOnFailure[PipelineQuery, blm.ImpressionBloomFilterSeq] {

override def defaultValue: blm.ImpressionBloomFilterSeq =

blm.ImpressionBloomFilterSeq(Seq.empty)

}

object IsForegroundRequestFeature extends Feature[PipelineQuery, Boolean]

object IsLaunchRequestFeature extends Feature[PipelineQuery, Boolean]

object LastNonPollingTimeFeature extends Feature[PipelineQuery, Option[Time]]

object NonPollingTimesFeature extends Feature[PipelineQuery, Seq[Long]]

object PersistenceEntriesFeature extends Feature[PipelineQuery, Seq[TimelineResponseV3]]

object PollingFeature extends Feature[PipelineQuery, Boolean]

object PullToRefreshFeature extends Feature[PipelineQuery, Boolean]

// Scores from Real Graph representing the relationship between the viewer and another user

object RealGraphInNetworkScoresFeature extends Feature[PipelineQuery, Map[UserId, Double]]

object RequestJoinIdFeature extends Feature[TweetCandidate, Option[Long]]

// Internal id generated per request, mainly to deduplicate re-served cached tweets in logging

object ServedRequestIdFeature extends Feature[PipelineQuery, Option[Long]]

object ServedTweetIdsFeature extends Feature[PipelineQuery, Seq[Long]]

object ServedTweetPreviewIdsFeature extends Feature[PipelineQuery, Seq[Long]]

object TimelineServiceTweetsFeature extends Feature[PipelineQuery, Seq[Long]]

object TimestampFeature

extends DataRecordFeature[PipelineQuery, Long]

with LongDiscreteDataRecordCompatible {

override def featureName: String = SharedFeatures.TIMESTAMP.getFeatureName

override def personalDataTypes: Set[pd.PersonalDataType] = Set.empty

}

object TimestampGMTDowFeature

extends DataRecordFeature[PipelineQuery, Long]

with LongDiscreteDataRecordCompatible {

override def featureName: String = RequestContextFeatures.TIMESTAMP\_GMT\_DOW.getFeatureName

override def personalDataTypes: Set[pd.PersonalDataType] = Set.empty

}

object TimestampGMTHourFeature

extends DataRecordFeature[PipelineQuery, Long]

with LongDiscreteDataRecordCompatible {

override def featureName: String = RequestContextFeatures.TIMESTAMP\_GMT\_HOUR.getFeatureName

override def personalDataTypes: Set[pd.PersonalDataType] = Set.empty

}

object TweetImpressionsFeature extends Feature[PipelineQuery, Seq[imp.TweetImpressionsEntry]]

object UserFollowedTopicsCountFeature extends Feature[PipelineQuery, Option[Int]]

object UserFollowingCountFeature extends Feature[PipelineQuery, Option[Int]]

object UserScreenNameFeature extends Feature[PipelineQuery, Option[String]]

object UserStateFeature extends Feature[PipelineQuery, Option[um.UserState]]

object UserTypeFeature extends Feature[PipelineQuery, Option[gt.UserType]]

object WhoToFollowExcludedUserIdsFeature

extends FeatureWithDefaultOnFailure[PipelineQuery, Seq[Long]] {

override def defaultValue = Seq.empty

}

// Result Features

object ServedSizeFeature extends Feature[PipelineQuery, Option[Int]]

object HasRandomTweetFeature extends Feature[PipelineQuery, Boolean]

object IsRandomTweetAboveFeature extends Feature[TweetCandidate, Boolean]

object ServedInConversationModuleFeature extends Feature[TweetCandidate, Boolean]

object ConversationModule2DisplayedTweetsFeature extends Feature[TweetCandidate, Boolean]

object ConversationModuleHasGapFeature extends Feature[TweetCandidate, Boolean]

object SGSValidLikedByUserIdsFeature extends Feature[TweetCandidate, Seq[Long]]

object SGSValidFollowedByUserIdsFeature extends Feature[TweetCandidate, Seq[Long]]

object PerspectiveFilteredLikedByUserIdsFeature extends Feature[TweetCandidate, Seq[Long]]

object ScreenNamesFeature extends Feature[TweetCandidate, Map[Long, String]]

object RealNamesFeature extends Feature[TweetCandidate, Map[Long, String]]

/\*\*

\* Features around the focal Tweet for Tweets which should be rendered in convo modules.

\* These are needed in order to render social context above the root tweet in a convo modules.

\* For example if we have a convo module A-B-C (A Tweets, B replies to A, C replies to B), the descendant features are

\* for the Tweet C. These features are None except for the root Tweet for Tweets which should render into

\* convo modules.

\*/

object FocalTweetAuthorIdFeature extends Feature[TweetCandidate, Option[Long]]

object FocalTweetInNetworkFeature extends Feature[TweetCandidate, Option[Boolean]]

object FocalTweetRealNamesFeature extends Feature[TweetCandidate, Option[Map[Long, String]]]

object FocalTweetScreenNamesFeature extends Feature[TweetCandidate, Option[Map[Long, String]]]

object MediaUnderstandingAnnotationIdsFeature extends Feature[TweetCandidate, Seq[Long]]

}