package com.twitter.timelines.prediction.common.aggregates

import com.twitter.ml.api.matcher.FeatureMatcher

import com.twitter.timelines.data\_processing.ml\_util.aggregation\_framework.TypedAggregateGroup

import scala.collection.JavaConverters.\_

object FeatureSelectorConfig {

val BasePairsToStore = Seq(

("twitter\_wide\_user\_aggregate.pair", "\*"),

("twitter\_wide\_user\_author\_aggregate.pair", "\*"),

("user\_aggregate\_v5.continuous.pair", "\*"),

("user\_aggregate\_v7.pair", "\*"),

("user\_author\_aggregate\_v2.pair", "recap.earlybird.\*"),

("user\_author\_aggregate\_v2.pair", "recap.searchfeature.\*"),

("user\_author\_aggregate\_v2.pair", "recap.tweetfeature.embeds\*"),

("user\_author\_aggregate\_v2.pair", "recap.tweetfeature.link\_count\*"),

("user\_author\_aggregate\_v2.pair", "engagement\_features.in\_network.\*"),

("user\_author\_aggregate\_v2.pair", "recap.tweetfeature.is\_reply.\*"),

("user\_author\_aggregate\_v2.pair", "recap.tweetfeature.is\_retweet.\*"),

("user\_author\_aggregate\_v2.pair", "recap.tweetfeature.num\_mentions.\*"),

("user\_author\_aggregate\_v5.pair", "\*"),

("user\_author\_aggregate\_tweetsource\_v1.pair", "\*"),

("user\_engager\_aggregate.pair", "\*"),

("user\_mention\_aggregate.pair", "\*"),

("user\_request\_context\_aggregate.dow.pair", "\*"),

("user\_request\_context\_aggregate.hour.pair", "\*"),

("user\_aggregate\_v6.pair", "\*"),

("user\_original\_author\_aggregate\_v1.pair", "\*"),

("user\_original\_author\_aggregate\_v2.pair", "\*"),

("original\_author\_aggregate\_v1.pair", "\*"),

("original\_author\_aggregate\_v2.pair", "\*"),

("author\_topic\_aggregate.pair", "\*"),

("user\_list\_aggregate.pair", "\*"),

("user\_topic\_aggregate.pair", "\*"),

("user\_topic\_aggregate\_v2.pair", "\*"),

("user\_inferred\_topic\_aggregate.pair", "\*"),

("user\_inferred\_topic\_aggregate\_v2.pair", "\*"),

("user\_media\_annotation\_aggregate.pair", "\*"),

("user\_media\_annotation\_aggregate.pair", "\*"),

("user\_author\_good\_click\_aggregate.pair", "\*"),

("user\_engager\_good\_click\_aggregate.pair", "\*")

)

val PairsToStore = BasePairsToStore ++ Seq(

("user\_aggregate\_v2.pair", "\*"),

("user\_aggregate\_v5.boolean.pair", "\*"),

("user\_aggregate\_tweetsource\_v1.pair", "\*"),

)

val LabelsToStore = Seq(

"any\_label",

"recap.engagement.is\_favorited",

"recap.engagement.is\_retweeted",

"recap.engagement.is\_replied",

"recap.engagement.is\_open\_linked",

"recap.engagement.is\_profile\_clicked",

"recap.engagement.is\_clicked",

"recap.engagement.is\_photo\_expanded",

"recap.engagement.is\_video\_playback\_50",

"recap.engagement.is\_video\_quality\_viewed",

"recap.engagement.is\_replied\_reply\_impressed\_by\_author",

"recap.engagement.is\_replied\_reply\_favorited\_by\_author",

"recap.engagement.is\_replied\_reply\_replied\_by\_author",

"recap.engagement.is\_report\_tweet\_clicked",

"recap.engagement.is\_block\_clicked",

"recap.engagement.is\_mute\_clicked",

"recap.engagement.is\_dont\_like",

"recap.engagement.is\_good\_clicked\_convo\_desc\_favorited\_or\_replied",

"recap.engagement.is\_good\_clicked\_convo\_desc\_v2",

"itl.engagement.is\_favorited",

"itl.engagement.is\_retweeted",

"itl.engagement.is\_replied",

"itl.engagement.is\_open\_linked",

"itl.engagement.is\_profile\_clicked",

"itl.engagement.is\_clicked",

"itl.engagement.is\_photo\_expanded",

"itl.engagement.is\_video\_playback\_50"

)

val PairGlobsToStore = for {

(prefix, suffix) <- PairsToStore

label <- LabelsToStore

} yield FeatureMatcher.glob(prefix + "." + label + "." + suffix)

val BaseAggregateV2FeatureSelector = FeatureMatcher

.none()

.or(

FeatureMatcher.glob("meta.user\_id"),

FeatureMatcher.glob("meta.author\_id"),

FeatureMatcher.glob("entities.original\_author\_id"),

FeatureMatcher.glob("entities.topic\_id"),

FeatureMatcher

.glob("entities.inferred\_topic\_ids" + TypedAggregateGroup.SparseFeatureSuffix),

FeatureMatcher.glob("timelines.meta.list\_id"),

FeatureMatcher.glob("list.id"),

FeatureMatcher

.glob("engagement\_features.user\_ids.public" + TypedAggregateGroup.SparseFeatureSuffix),

FeatureMatcher

.glob("entities.users.mentioned\_screen\_names" + TypedAggregateGroup.SparseFeatureSuffix),

FeatureMatcher.glob("user\_aggregate\_v2.pair.recap.engagement.is\_dont\_like.\*"),

FeatureMatcher.glob("user\_author\_aggregate\_v2.pair.any\_label.recap.tweetfeature.has\_\*"),

FeatureMatcher.glob("request\_context.country\_code"),

FeatureMatcher.glob("request\_context.timestamp\_gmt\_dow"),

FeatureMatcher.glob("request\_context.timestamp\_gmt\_hour"),

FeatureMatcher.glob(

"semantic\_core.media\_understanding.high\_recall.non\_sensitive.entity\_ids" + TypedAggregateGroup.SparseFeatureSuffix)

)

val AggregatesV2ProdFeatureSelector = BaseAggregateV2FeatureSelector

.orList(PairGlobsToStore.asJava)

val ReducedPairGlobsToStore = (for {

(prefix, suffix) <- BasePairsToStore

label <- LabelsToStore

} yield FeatureMatcher.glob(prefix + "." + label + "." + suffix)) ++ Seq(

FeatureMatcher.glob("user\_aggregate\_v2.pair.any\_label.\*"),

FeatureMatcher.glob("user\_aggregate\_v2.pair.recap.engagement.is\_favorited.\*"),

FeatureMatcher.glob("user\_aggregate\_v2.pair.recap.engagement.is\_photo\_expanded.\*"),

FeatureMatcher.glob("user\_aggregate\_v2.pair.recap.engagement.is\_profile\_clicked.\*")

)

}