package com.twitter.timelineranker.uteg\_liked\_by\_tweets

import com.twitter.search.earlybird.thriftscala.ThriftSearchResult

import com.twitter.servo.util.FutureArrow

import com.twitter.timelineranker.core.CandidateEnvelope

import com.twitter.timelines.model.UserId

import com.twitter.util.Future

object RemoveCandidatesAuthoredByWeightedFollowingsTransform

extends FutureArrow[CandidateEnvelope, CandidateEnvelope] {

override def apply(envelope: CandidateEnvelope): Future[CandidateEnvelope] = {

val filteredSearchResults = envelope.query.utegLikedByTweetsOptions match {

case Some(opts) =>

envelope.searchResults.filterNot(isAuthorInWeightedFollowings(\_, opts.weightedFollowings))

case None => envelope.searchResults

}

Future.value(envelope.copy(searchResults = filteredSearchResults))

}

private def isAuthorInWeightedFollowings(

searchResult: ThriftSearchResult,

weightedFollowings: Map[UserId, Double]

): Boolean = {

searchResult.metadata match {

case Some(metadata) => weightedFollowings.contains(metadata.fromUserId)

case None => false

}

}

}