package com.twitter.search.ingester.pipeline.strato\_fetchers;

import scala.Option;

import com.twitter.cuad.ner.plain.thriftjava.NamedEntities;

import com.twitter.cuad.ner.plain.thriftjava.NamedEntitiesRequestOptions;

import com.twitter.cuad.ner.thriftjava.ModelFamily;

import com.twitter.cuad.ner.thriftjava.NERCalibrateRequest;

import com.twitter.cuad.thriftjava.CalibrationLevel;

import com.twitter.cuad.thriftjava.NERCandidateSource;

import com.twitter.stitch.Stitch;

import com.twitter.strato.catalog.Fetch;

import com.twitter.strato.client.Client;

import com.twitter.strato.client.Fetcher;

import com.twitter.strato.data.Conv;

import com.twitter.strato.opcontext.ServeWithin;

import com.twitter.strato.thrift.TBaseConv;

import com.twitter.util.Duration;

import com.twitter.util.Future;

public class NamedEntityFetcher {

private static final String NAMED\_ENTITY\_STRATO\_COLUMN = "";

private static final ServeWithin SERVE\_WITHIN = new ServeWithin(

Duration.fromMilliseconds(100), Option.empty());

private static final NamedEntitiesRequestOptions REQUEST\_OPTIONS =

new NamedEntitiesRequestOptions(

new NERCalibrateRequest(CalibrationLevel.HIGH\_PRECISION, NERCandidateSource.NER\_CRF)

.setModel\_family(ModelFamily.CFB))

.setDisplay\_entity\_info(false);

private final Fetcher<Long, NamedEntitiesRequestOptions, NamedEntities> fetcher;

public NamedEntityFetcher(Client stratoClient) {

fetcher = stratoClient.fetcher(

NAMED\_ENTITY\_STRATO\_COLUMN,

true, // enables checking types against catalog

Conv.longConv(),

TBaseConv.forClass(NamedEntitiesRequestOptions.class),

TBaseConv.forClass(NamedEntities.class)).serveWithin(SERVE\_WITHIN);

}

public Future<Fetch.Result<NamedEntities>> fetch(long tweetId) {

return Stitch.run(fetcher.fetch(tweetId, REQUEST\_OPTIONS));

}

}