package com.twitter.search.common.relevance.classifiers;

import com.google.common.base.Preconditions;

import com.twitter.search.common.relevance.entities.TwitterMessage;

/\*\*

\* Interface to perform quality evaluation for a single @TwitterMessage

\* object or a group of them.

\*

\*/

public abstract class TweetEvaluator {

/\*\*

\* Passed in TwitterMessage is examined and any extractable

\* features are stored in TweetFeatures field of TwitterMessage.

\*

\* @param tweet TwitterMessage to perform classification on.

\*/

public abstract void evaluate(final TwitterMessage tweet);

/\*\*

\* Classify a group of TwitterMessages and store the features in their corresponding

\* TweetFeatures fields.

\*

\* This default implementation just iterates through the map and classifies each

\* individual tweet. Batching for better performance, if applicable, can be implemented by

\* concrete subclasses.

\*

\* @param tweets TwitterMessages to perform classification on.

\*/

public void evaluate(final Iterable<TwitterMessage> tweets) {

Preconditions.checkNotNull(tweets);

for (TwitterMessage tweet: tweets) {

evaluate(tweet);

}

}

}