namespace java com.twitter.simclustersann.thriftjava

#@namespace scala com.twitter.simclustersann.thriftscala

include "finatra-thrift/finatra\_thrift\_exceptions.thrift"

include "com/twitter/simclusters\_v2/identifier.thrift"

include "com/twitter/simclusters\_v2/score.thrift"

struct Query {

1: required identifier.SimClustersEmbeddingId sourceEmbeddingId;

2: required SimClustersANNConfig config;

}

struct SimClustersANNTweetCandidate {

1: required i64 tweetId (personalDataType = 'TweetId');

2: required double score;

}

struct SimClustersANNConfig {

1: required i32 maxNumResults;

2: required double minScore;

3: required identifier.EmbeddingType candidateEmbeddingType;

4: required i32 maxTopTweetsPerCluster;

5: required i32 maxScanClusters;

6: required i32 maxTweetCandidateAgeHours;

7: required i32 minTweetCandidateAgeHours;

8: required ScoringAlgorithm annAlgorithm;

}

/\*\*

\* The algorithm type to identify the score algorithm.

\*\*/

enum ScoringAlgorithm {

DotProduct = 1,

CosineSimilarity = 2,

LogCosineSimilarity = 3,

CosineSimilarityNoSourceEmbeddingNormalization = 4, // Score = (Source dot Candidate) / candidate\_l2\_norm

}(hasPersonalData = 'false')

enum InvalidResponseParameter {

INVALID\_EMBEDDING\_TYPE = 1,

INVALID\_MODEL\_VERSION = 2,

}

exception InvalidResponseParameterException {

1: required InvalidResponseParameter errorCode,

2: optional string message // failure reason

}

service SimClustersANNService {

list<SimClustersANNTweetCandidate> getTweetCandidates(

1: required Query query;

) throws (

1: InvalidResponseParameterException e;

2: finatra\_thrift\_exceptions.ServerError serverError;

3: finatra\_thrift\_exceptions.ClientError clientError;

);

}