// For details of how to encode and decode thrift, check

// https://github.com/apache/thrift/blob/master/doc/specs/thrift-binary-protocol.md

// Definitions of the thrift binary format

typedef enum {

TTYPE\_STOP = 0,

TTYPE\_VOID = 1,

TTYPE\_BOOL = 2,

TTYPE\_BYTE = 3,

TTYPE\_DOUBLE = 4,

TTYPE\_I16 = 6,

TTYPE\_I32 = 8,

TTYPE\_I64 = 10,

TTYPE\_STRING = 11,

TTYPE\_STRUCT = 12,

TTYPE\_MAP = 13,

TTYPE\_SET = 14,

TTYPE\_LIST = 15,

TTYPE\_ENUM = 16,

} TTYPES;

// Fields of a batch prediction response

typedef enum {

BPR\_DUMMY ,

BPR\_PREDICTIONS,

} BPR\_FIELDS;

// Fields of a datarecord

typedef enum {

DR\_CROSS , // fake field for crosses

DR\_BINARY ,

DR\_CONTINUOUS ,

DR\_DISCRETE ,

DR\_STRING ,

DR\_SPARSE\_BINARY ,

DR\_SPARSE\_CONTINUOUS ,

DR\_BLOB ,

DR\_GENERAL\_TENSOR ,

DR\_SPARSE\_TENSOR ,

} DR\_FIELDS;

// Fields for General tensor

typedef enum {

GT\_DUMMY , // dummy field

GT\_RAW ,

GT\_STRING ,

GT\_INT32 ,

GT\_INT64 ,

GT\_FLOAT ,

GT\_DOUBLE ,

GT\_BOOL ,

} GT\_FIELDS;

typedef enum {

SP\_DUMMY , // dummy field

SP\_COO ,

} SP\_FIELDS;

// Enum values from tensor.thrift

typedef enum {

DATA\_TYPE\_FLOAT ,

DATA\_TYPE\_DOUBLE ,

DATA\_TYPE\_INT32 ,

DATA\_TYPE\_INT64 ,

DATA\_TYPE\_UINT8 ,

DATA\_TYPE\_STRING ,

DATA\_TYPE\_BYTE ,

DATA\_TYPE\_BOOL ,

} DATA\_TYPES;