differences

Ed Mazurek

9/5/2018

ABI function comparison

I took your excel file, pulled it into R, got the contract addresses from it, and wrote my own etherscan scraper. I used the etherscan API to pull the ABIs for each contract. Etherscan API wouldn't let me pull REP's ABI since it's unverified. I pulled all the others. After doing this, I did some data jiggling so that my file (data frame, rather) looked like yours.

After doing this, I compared our two lists, and found differences.

Main differences:

- Your file doesn't include underscores in function names. My method preserves the underscores in the function names.
- Your file has erroneous function names for MKR. It appends 9f8f to some names.

Here are the abi functions that I have that differ from yours:

```
my.abi.fn.data %>%
  anti_join(jay.abi.fn.data) %>%
  select(-Address)
```

```
## Joining, by = c("Symbol", "Address", "Type", "Constant", "Name", "Signature", "Variab
le Names")
```

Symbol	Type	Constant	Name	Signature	Variable Names
USDT	function	1	_totalSupply	_totalSupply()	NA
USDT	function	1	MAX_UINT	MAX_UINT()	NA
MKR	function	0	burn	burn(address,uint256)	guy,wad,
MKR	function	0	mint	mint(uint256)	wad,
MKR	function	0	approve	approve(address)	guy,
PPT	function	1	BURN_ADDRESS	BURN_ADDRESS()	NA
LOOM	function	1	INITIAL_SUPPLY	INITIAL_SUPPLY()	NA
WAX	function	1	INITIAL_SUPPLY	INITIAL_SUPPLY()	NA
NOAH	function	1	INITIAL_SUPPLY	INITIAL_SUPPLY()	NA
LRC	function	1	BLOCKS_PER_PHASE	BLOCKS_PER_PHASE()	NA
LRC	function	1	MAX_UNSOLD_RATIO	MAX_UNSOLD_RATIO()	NA

Symbol	Type	Constant	Name	Signature	Variable Names
LRC	function	1	HARD_CAP	HARD_CAP()	NA
LRC	function	1	BASE_RATE	BASE_RATE()	NA
LRC	function	1	NUM_OF_PHASE	NUM_OF_PHASE()	NA
CMT	function	1	INITIAL_SUPPLY	INITIAL_SUPPLY()	NA
ENG	function	1	INITIAL_SUPPLY	INITIAL_SUPPLY()	NA
DROP	function	1	_totalSupply	_totalSupply()	NA

Here are the abi functions that you have that differ from mine:

```
jay.abi.fn.data %>%
  anti_join(my.abi.fn.data) %>%
  filter(Symbol != 'REP') %>%
  select(-Address) %>%
  as.data.frame()
```

```
## Joining, by = c("Symbol", "Address", "Type", "Constant", "Name", "Signature", "Variab
le Names")
```

Symbol	Type	Constant	Name	Signature	Variable Names
USDT	function	1	MAXUINT	MAXUINT()	NA
MKR	function	0	burn9f8f	burn(address,uint256)	guy,wad,
MKR	function	0	mint9f8f	mint(uint256)	wad,
MKR	function	0	approve9f8f	approve(address)	guy,
PPT	function	1	BURNADDRESS	BURNADDRESS()	NA
LOOM	function	1	INITIALSUPPLY	INITIALSUPPLY()	NA
WAX	function	1	INITIALSUPPLY	INITIALSUPPLY()	NA
NOAH	function	1	INITIALSUPPLY	INITIALSUPPLY()	NA
LRC	function	1	BLOCKSPERPHASE	BLOCKSPERPHASE()	NA
LRC	function	1	MAXUNSOLDRATIO	MAXUNSOLDRATIO()	NA
LRC	function	1	HARDCAP	HARDCAP()	NA
LRC	function	1	BASERATE	BASERATE()	NA
LRC	function	1	NUMOFPHASE	NUMOFPHASE()	NA
CMT	function	1	INITIALSUPPLY	INITIALSUPPLY()	NA
ENG	function	1	INITIALSUPPLY	INITIALSUPPLY()	NA

ABI Event comparsion

For events, I noticed no differences between the data I pulled from Etherscan and the data you provided.

```
my.abi.event.data %>%
  anti_join(jay.abi.event.data) %>%
  select(-Address)
```

```
## Joining, by = c("Symbol", "Address", "Type", "Constant", "Name", "Signature", "Variab
le Names")
```

Symbol Type Constant Name Signature Variable Names ——- —— ——— —— ———- —————

```
jay.abi.event.data %>%
  anti_join(my.abi.event.data) %>%
  filter(Symbol != 'REP') %>%
  select(-Address) %>%
  as.data.frame()
```

```
## Joining, by = c("Symbol", "Address", "Type", "Constant", "Name", "Signature", "Variab
le Names")
```

Symbol Type Constant Name Signature Variable Names ——- —— ——— —— ———- —————

Btw, here’s the code for my scraper:

```
require(readxl)
require(dplyr)
require(readr)
require(jsonlite)

data <- read_excel("Top ICOs Function and Event Encodings.xlsx", skip=1)
data2 <- read_excel("Top ICOs Function and Event Encodings.xlsx", sheet=2, skip=1)

contract.addr <- data %>% distinct(Address, .keep_all=T) %>% select(Symbol, Address) %>%
 split(seq(nrow(.)))

# lapply(X = contract.addr, FUN = function(x) {
#   curl::curl_download(url = paste0('https://api.etherscan.io/api?module=contract&actio
n=getabi&address=', x$Address, '&apikey=YourApiKeyToken'),
#                       destfile = paste0('abi_', x$Symbol, '.json'))
#   Sys.sleep(1)
# })

make.df <- function(address, symbol) {

  this.json <- read_json(path = paste0('abi_', symbol,'.json'))
  print(symbol)
  if(this.json$status == "0") {
  return()
  }
  else{
  this.json <- this.json$result %>% fromJSON(simplifyVector=F, simplifyDataFrame = F, si
mplifyMatrix = T, flatten=F)

  sapply(X = this.json, FUN = function(x) {
    input.types = sapply(X = x$inputs, FUN = function(y) {y$type})
    input.names = mapply(FUN = function(y, z) {
      return(ifelse(y$name == '', paste0('param_',z), y$name))
      }, x$inputs, seq_along(x$inputs)-1)
    list(Type = x$type %>% as.character(),
         Name = x$name %>% as.character(),
         Constant = x$constant,
         Signature = paste0(x$name, "(", paste0(input.types, collapse = ','), ")") %>% a
s.character(),
         `Variable Names` = paste0(input.names, collapse = ',') %>% as.character())
    }) %>% t() %>% as.data.frame() %>%
    mutate(Address = address,
           Symbol = symbol,
           `Variable Names` = ifelse(`Variable Names` == '', NA, paste0(`Variable Names
`, ',')),
           Constant = ifelse(Constant == "TRUE", 1, 0)) %>%
    return()
  }
}

result <- sapply(X = contract.addr, FUN = function(x) {make.df(x$Address, x$Symbol)}) %
>%
  bind_rows() %>%
  mutate(Name = as.character(Name),
         Signature = as.character(Signature),
         Type = as.character(Type),
         Constant = as.double(Constant))

my.abi.fn.data <- result %>% filter(Type == "function") %>% select(Symbol, Address, Typ
e, Constant, Name, Signature, `Variable Names`)
jay.abi.fn.data <- data %>% select(-Encoding, -X__1)

my.abi.event.data <- result %>% filter(Type == "event") %>% select(Symbol, Address, Typ
e, Constant, Name, Signature, `Variable Names`)
jay.abi.event.data <- data2 %>% select(-Encoding, -X__1)
```