Computation over Encrypted Data via Homomorphic Encryption

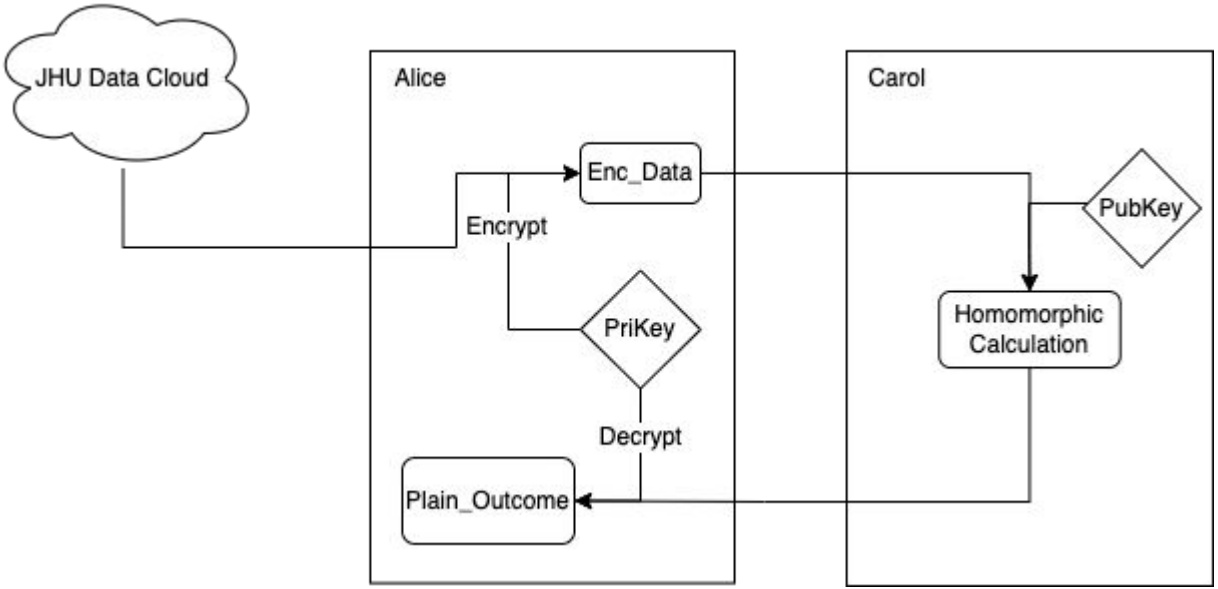
Ruicheng Yang (ry2172), Yuheng Zhong (yz6422)

05/09/2022

Part 1

Solution Design

Solution Design



Flow Chart

Context Encryption

Data Retrieve

Encryption

Computation over HE Data

Decryption

Result

Solution Design

TenSEAL

https://github.com/OpenMined/TenSEAL

A Python library doing homomorphic encryption over vectors or tensors, based on Microsoft SEAL

Supports FHE with CKKS and BFV schemes

Supports arithmetic except division, and several matrix operations

Solution Design

Components

Dataset

COVID-19 Data Repository by the Center for Systems Science and Engineering (CSSE) at Johns Hopkins University

Libraries

TenSEAL - Fully homomorphic encryption
numpy - Arrays and matrices
pandas - Data manipulation and analysis
pickle - File processing
cProfile - Performance analysis

Environment

MacBook Pro 15, Intel i7

Part 2

Coding Details

Coding Details

About data set

https://github.com/CSSEGISandData/COVID-19

An open source COVID-19 data set maintained by JHU (time-series, updated daily)

We use US confirmed and deaths as our data

Coding Details

About data set

	FIPS	1/22/20	1/23/20	1/24/20	1/25/20	1/26/20	1/27/20	1/28/20	1/29/20	1/30/20	...	4/28/22	4/29/22	4/30/22	5/1/22	5/2/22	5/3/22	5/4/22
0	1001	0	0	0	0	0	0	0	0	0	...	15826	15827	15827	15827	15833	15835	15839
1	1003	0	0	0	0	0	0	0	0	0	...	55633	55643	55643	55643	55664	55685	55695
2	1005	0	0	0	0	0	0	0	0	0	...	5665	5668	5668	5668	5670	5671	5671
3	1007	0	0	0	0	0	0	0	0	0	...	6439	6442	6442	6442	6442	6442	6443
4	1009	0	0	0	0	0	0	0	0	0	...	14977	14980	14980	14980	14979	14981	14983
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
3337	56039	0	0	0	0	0	0	0	0	0	...	10009	10009	10009	10009	10009	10047	10047
3338	56041	0	0	0	0	0	0	0	0	0	...	5627	5627	5627	5627	5627	5635	5635
3339	90056	0	0	0	0	0	0	0	0	0	...	0	0	0	0	0	0	0
3340	56043	0	0	0	0	0	0	0	0	0	...	2358	2358	2358	2358	2358	2357	2357
3341	56045	0	0	0	0	0	0	0	0	0	...	1588	1588	1588	1588	1588	1589	1589

E.g. Confirmed dataframe

Coding Details

Public functions

```
getFIPSInd(FIPS_lookup_table, county=' ', state=' ')
```

Get an FIPS indicator vector for further query

```
getFIPSInd(FIPS_lookup, County='Bibb', State='Alabama')
```

```
[0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
```

E.g. Bibb in Alabama State

Coding Details

Public functions

```
getDates(begin_date='1/22/20', end_date=today (ET))
```

Get a date list, starting from 1/22/20 to today by default

```
last_week_date = datetime.datetime.strftime(datetime.datetime.now(timezone('US/Eastern'))
                                            - datetime.timedelta(days=7), '%-m/%-d/%-y')
getDates(begin_date=last_week_date)
```

```
['5/1/22', '5/2/22', '5/3/22', '5/4/22', '5/5/22', '5/6/22', '5/7/22']
```

E.g. A list of dates, from last week

Coding Details

Alice

`CKKSSetup()` & `BFVSetup()`

Set up an FHE context, save as 2 files after serialization

```
secret_ctx = context.serialize(save_secret_key=True)
with open(secret_ctx_path, 'wb') as f: pickle.dump(secret_ctx, f)
context.make_context_public() # drop the secret key from the context
public_ctx = context.serialize()
with open(public_ctx_path, 'wb') as f: pickle.dump(public_ctx, f)
```

Part of this function

Coding Details

Alice

`dataProcess(scheme)`

Load data set, finish FHE (using secret context), store encrypted data (serialized)

```
enc_confirmed
```

```
{'1/1/21': <tenseal.tensors.bfvvector.BFVVector at 0x7f73094e0fd0>,
 '1/1/22': <tenseal.tensors.bfvvector.BFVVector at 0x7f7309647ad0>,
 '1/10/21': <tenseal.tensors.bfvvector.BFVVector at 0x7f73094e0190>,
 '1/10/22': <tenseal.tensors.bfvvector.BFVVector at 0x7f7309647610>,
```

E.g. Encrypted confirmed cases `(scheme='BFV')`

Coding Details

Alice

```
decrypt(scheme)
```

Decrypt encrypted result (using secret context)

Coding Details

Carol

`fetchFHEData(scheme)`

Get encrypted data files, deserialize (using public context)

```
enc_confirmed
```

```
{'1/1/21': <tenseal.tensors.bfvvector.BFVVector at 0x7f73094e0fd0>,
 '1/1/22': <tenseal.tensors.bfvvector.BFVVector at 0x7f7309647ad0>,
 '1/10/21': <tenseal.tensors.bfvvector.BFVVector at 0x7f73094e0190>,
 '1/10/22': <tenseal.tensors.bfvvector.BFVVector at 0x7f7309647610>,
```

E.g. Encrypted confirmed cases `(scheme='BFV')`

Coding Details

Carol

`EvalFunc(enc_confirmed, enc_deaths, FIPS_lookup_table)`

Do computation(s) over encrypted data, store encrypted result (serialized)

```
FIPSindicator_caliandnys = np.add(getFIPSInd(FIPS_lookup, State='California'), getFIPSInd(FIPS_lookup, State='New York'))
print('Confirmed cases sum from last week (' + last_week_date + '-' + latest_record_date + ') in California and New York State')
func = (enc_confirmed[latest_record_date] - enc_confirmed[last_week_date]).dot(FIPSindicator_caliandnys)

with open(enc_result_path, 'wb') as f:
    pickle.dump(func.serialize(), f)
```

E.g. A function for confirmed case sum from last week in California and NYS

Coding Details

Performance Showdown

plaintextQuery

```
Confirmed cases sum from last week (5/1/22-5/7/22) in California and New York State
109137
         2016 function calls (1986 primitive calls) in 0.004 seconds

   Ordered by: internal time

   ncalls  tottime  percall  cumtime  percall filename:lineno(function)
        1    0.000    0.000    0.004    0.004 <ipython-input-27-03c9cf1ead02>:3(plainQuery)
        2    0.000    0.000    0.000    0.000 {pandas._libs.ops.scalar_compare}
      375    0.000    0.000    0.000    0.000 {built-in method builtins.isinstance}
        2    0.000    0.000    0.000    0.000 managers.py:224(_rebuild_blknos_and_blklocs)
        2    0.000    0.000    0.000    0.000 base.py:2018(is_unique)
        5    0.000    0.000    0.000    0.000 socket.py:480(send)
        9    0.000    0.000    0.001    0.000 series.py:315(__init__)
        1    0.000    0.000    0.004    0.004 {built-in method builtins.exec}
        2    0.000    0.000    0.000    0.000 {pandas._libs.lib.infer_dtype}
        9    0.000    0.000    0.000    0.000 generic.py:5435(__finalize__)
        6    0.000    0.000    0.001    0.000 frame.py:3418(__getitem__)
```

Coding Details

Performance Showdown

CKKSQuery

```
Confirmed cases sum from last week (5/1/22-5/7/22) in California and New York State
CKKS result: 109137
         1703 function calls (1680 primitive calls) in 0.436 seconds

   Ordered by: internal time

   ncalls  tottime  percall  cumtime  percall filename:lineno(function)
        1    0.274    0.274    0.274    0.274 {built-in method _tenseal_cpp.deserialize}
        1    0.142    0.142    0.142    0.142 ckksvector.py:133(dot)
        1    0.004    0.004    0.278    0.278 enc_context.py:166(load)
        1    0.003    0.003    0.003    0.003 abstract_tensor.py:81(_decrypt)
        3    0.002    0.001    0.002    0.001 {built-in method io.open}
        1    0.002    0.002    0.002    0.002 abstract_tensor.py:39(load)
        1    0.002    0.002    0.002    0.002 abstract_tensor.py:72(serialize)
        1    0.001    0.001    0.435    0.435 <ipython-input-36-33f452956e2f>:9(CKKSQuery)
        2    0.001    0.000    0.001    0.000 {built-in method _pickle.load}
        1    0.001    0.001    0.149    0.149 <ipython-input-2-5a4990d12220>:148(CKKSEvalFunc)
```

Coding Details

Performance Showdown

BFVQuery

```
Confirmed cases sum from last week (5/1/22-5/7/22) in California and New York State
BFV result: 109137
         1703 function calls (1680 primitive calls) in 0.779 seconds

   Ordered by: internal time

   ncalls  tottime  percall  cumtime  percall filename:lineno(function)
        1    0.440    0.440    0.440    0.440 {built-in method _tenseal_cpp.deserialize}
        1    0.319    0.319    0.320    0.320 bfvvector.py:118(dot)
        1    0.003    0.003    0.003    0.003 abstract_tensor.py:72(serialize)
        1    0.003    0.003    0.003    0.003 abstract_tensor.py:81(_decrypt)
        1    0.002    0.002    0.002    0.002 abstract_tensor.py:39(load)
       10    0.002    0.000    0.002    0.000 socket.py:480(send)
        1    0.001    0.001    0.001    0.001 bfvvector.py:95(sub)
        2    0.001    0.001    0.001    0.001 {built-in method _pickle.load}
        1    0.001    0.001    0.001    0.001 managers.py:224(_rebuild_blknos_and_blklocs)
        1    0.001    0.001    0.440    0.440 enc_context.py:166(load)
```

Part 3

Demo

Appendix

GitHub Link

https://github.com/lyuheng13/HomomorphicComp