Poster: Privacy-Preserving String Search for Genome Sequences using Fully Homomorphic Encryption

Yu Ishimaki[1] Kana Shimizu[1,2] Koji Nuida[2,3] Hayato Yamana[1]

[1]Waseda University, Tokyo, Japan [2]National Institute of Advanced Industrial Science and Technology(AIST), Tokyo, Japan
[3]Japan Science and Technology Agency (JST) PRESTO Researcher

Introduction (Secure Genome Search Protocol)


= 

Genome Sequence Individual ID

Privacy protection for genome sequences is indispensable

Does Bob have a typical pattern for the disease?

1 2 3 4 5 6 ...
A T C G T G...

Bob

User(Client)

G T G 4

30%

search!!

Server

	1	2	3	4	5	6	
(normal)	T	G	A	G	T	A	...
(normal)	T	A	C	G	A	G	...
(normal)	A	C	A	T	T	C	...
(disease)	G	C	T	G	T	G	...

- User only knows the statistics
- Server does not learn the user's query

Previous Work using Additively Homomorphic Encryption (PBWT-sec[1] [4])

- Efficient searchable data structure(PBWT)[2]

Query= G T G

	1	2	3	4	5	6	...
x_1	T	G	A	G	T	A	...
x_2	T	A	C	G	A	G	...
x_3	A	C	A	T	T	C	...
x_4	G	C	T	G	T	G	...

→

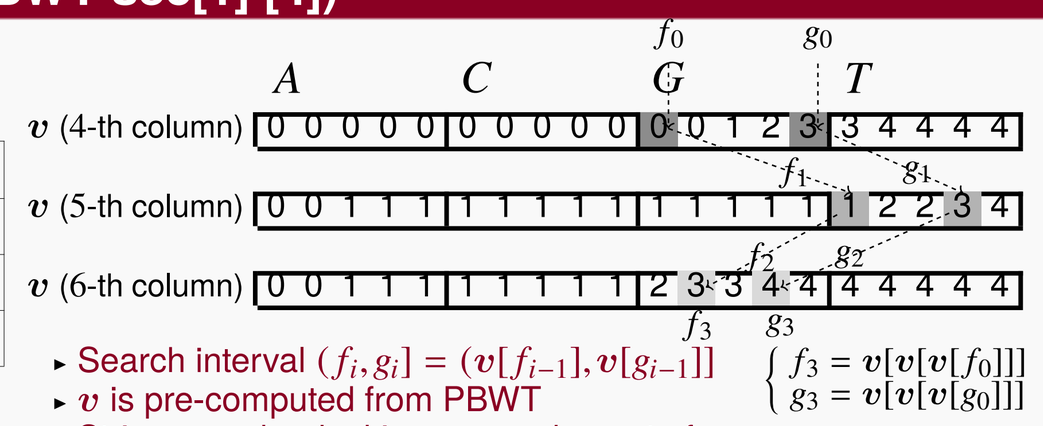
1	2	3	4	5	6	...
x_3	x_2	x_3	x_1	x_2	x_1	...
x_4	x_3	x_1	x_2	x_1	x_3	...
x_1	x_4	x_2	x_4	x_4	x_2	...
x_2	x_1	x_4	x_3	x_3	x_4	...

→

	2	3	4	5	6	...
1	C	C	T	T	G	...
2	C	A	G	A	A	...
3	G	T	G	T	G	...
4	A	A	G	T	C	...

f_1 f_2 f_3 ... g_1 g_2 g_3

original database sorted order PBWT



- Search interval $(f_i, g_i] = (v[f_{i-1}], v[g_{i-1}]]$
- v is pre-computed from PBWT
- String search = looking-up an element of v

$$\begin{cases} f_3 = v[v[v[f_0]]] \\ g_3 = v[v[v[g_0]]] \end{cases}$$

- Look-up $v[t]$ by Oblivious Transfer (OT) to protect "t"

$\boldsymbol{v} = [v[1], \ldots, v[t], \ldots, v[n]]$

Query: $\boldsymbol{q} = [q[1] = 0, \ldots, q[t] = 1, \ldots, q[n] = 0]$

$$c = \bigoplus_{i=1}^{n} (\mathrm{Enc}(q[i]) \otimes v[i]) = \boxed{\mathrm{v[t]}}$$

- Look-up $v[\ldots v[t] \ldots]$ by Recursive OT to protect "$\boldsymbol{v}$" [1]

$$c = \bigoplus_{i=1}^{n} (\mathrm{Enc}(q[i]) \otimes (v[i] + r) \mod n)$$

Set the next query

$$\boldsymbol{q} = [q[1] = 0, \ldots, q[(v[t] + r) \mod n] = 1, \ldots, q[n] = 0]$$

r-left permutation

$$\vec{\mathrm{Enc}}(\boldsymbol{q}) = [\mathrm{Enc}(q[1]) = 0, \ldots, \mathrm{Enc}(q[v[t]] = 1), \ldots, \mathrm{Enc}(q[n]) = 0]$$

Again compute c

PBWT-sec only counts # of prefix match (advanced statistics and wildcard search are not available) because it uses additively HE → FHE is desired!

Our Proposed Method adopting Fully Homomorphic Encryption

- Polynomial-CRT packing[3]

encrypt one vector into one ciphertext

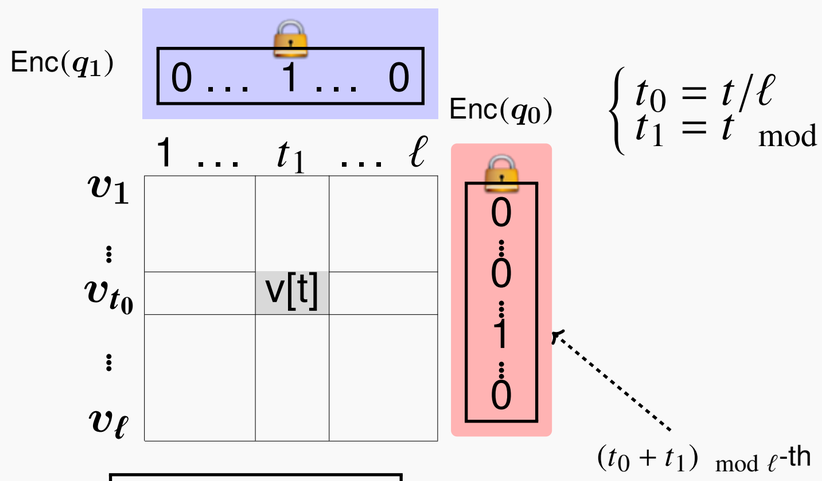
$\boxed{2\,0\,4\,0\,0} \oplus \boxed{0\,0\,0\,0\,1} = \boxed{2\,0\,4\,0\,1}$

$\boxed{2\,0\,4\,0\,0} \otimes \boxed{1\,0\,2\,0\,1} = \boxed{2\,0\,8\,0\,0}$

$\mathrm{Perm}(\boxed{2\,0\,4\,5\,3}, 2) = \boxed{4\,5\,3\,2\,0}$

- Utilize element-wise computation

- Look-up $v[t]$ by 2D representation



server's vector

$$\begin{cases} t_0 = t/\ell \\ t_1 = t \mod \ell \end{cases}$$

$$c = \bigoplus_{i=1}^{\ell} (\ \mathrm{Enc}(\boldsymbol{q_1}) \otimes \mathrm{Perm}(\mathrm{Enc}(\boldsymbol{q_0}), i) \otimes \boldsymbol{v_i}\ )$$

Yield
- $\boxed{0 \ldots 1 \ldots 0}$ iff $i = t_0$
- $\boxed{0 \ldots 0 \ldots 0}$ (all zero vector) otherwise

$c = \boxed{0 \ldots \mathrm{v[t]} \ldots 0}$ Get v[t] !

- Temporal masking r is omitted here

Experimental Result

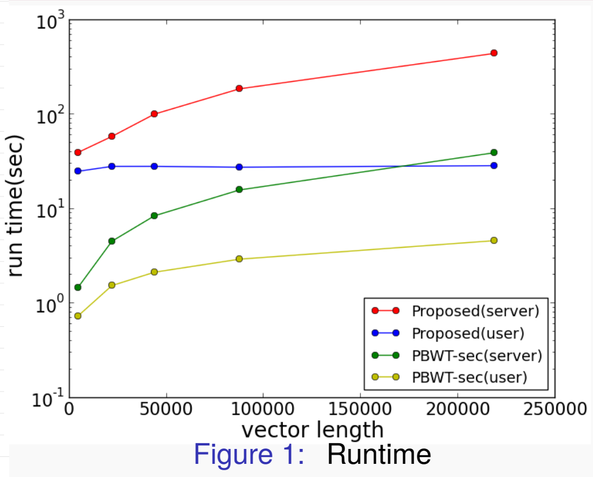


Figure 1: Runtime

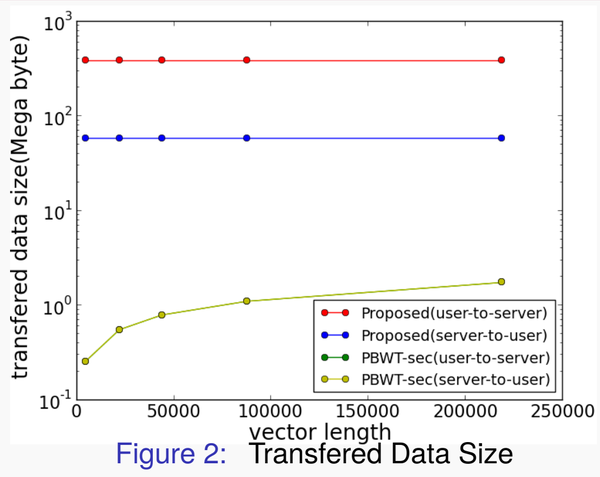


Figure 2: Transfered Data Size

Conclusion and Future Work

Conclusion

- Implemented secure genome search protocol with FHE, only 10 times slower than PBWT-sec.

Future Work

- Extend the functionality
  - conducting wild card search
  - calculating more advanced statistics

[1] K. Shimizu, K. Nuida and S. Rätsch: *Efficient Privacy-Preserving String Search and an Application in Genomics*, Bioinformatics, doi:10.1093/bioinformatics/btw050, 2016.

[2] R. Durbin: *Efficient haplotype matching and storage using the Positional Burrows-Wheeler Transform (PBWT)*, Bioinformatics, Vol. 30, No. 9, pp. 1266-1272, 2014.

[3] N. P. Smart and F. Vercauteren: *Fully homomorphic SIMD operations*, Designs, Codes and Cryptography, Vol. 71, No. 1, pp. 57-81, 2014.

[4] PBWT-sec. https://github.com/iskana/PBWT-sec/, accessed on 2016-4-1.