**Experiment**

The configuration we have used to simulate the proposed privacy-preserving candidate hiring architecture is an Intel i7-12700 processor clocked at a max frequency of 4900MHz and having 12 cores per socket.

The experiments have been done with the consideration of experts performing their tasks parallelly and then average execution time has been calculated.

The experiment results have been represented in two different types. First, we compare all four frameworks with an encryption key size of 128 bits, and later, another comparison between frameworks 2, 3, and 4 has been done by varying the encryption key sizes for homomorphic encryption. Framework 1 has not been considered in the second type because it does not use any homomorphic encryption, so changing the encryption key size will not affect its average execution time; considering it in the second comparison will not be so fruitful; the result of it will be the same for all the test cases hence it has been dropped in second type comparison.

The experiments have been performed by varying the encryption key size in [128, 256, 512] bits. For 128, all four models have been compared, i.e., type1, and for 256 and 512-bit encryption, frameworks 2,3 and 4 have been compared, i.e., type2.

Both types of experiments are performed considering the following two settings after picking an encryption key size from above above-described list of different key sizes. We try to study the average execution time for both types of experiments under the following two settings

1. By keeping the number of candidates constant and the number of experts variable.
2. Next, we varied the number of candidates, and the number of experts was kept constant.

**Experiment results for setting 1**

The number of experts has varied in [2,4,6,8,10], and the number of candidates has been kept at 20. The results comparing all four models for the encryption of size 128-bit have been shown in Figure 1,. 128-bit encryption is used in frameworks 2, 3, and 4, for framework 1 has no effect due to this encryption key size as it doesn’t contain homomorphic encryption. Later for the second type, a comparison between average execution time and number of experts for the second, third, and fourth frameworks with encryption key sizes of 256 and 512 bits has been shown in Figure 2 and Figure 3, respectively.

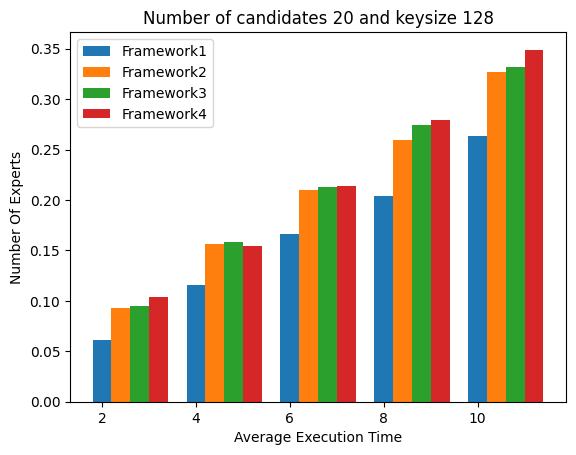


Figure 1: Comparing the execution time of all models with an encryption key size of 128-bit.

The results of Figure 1 show that the execution times increase as the complexity of the framework increases. Our framework 1 is the very basic model and involves only a digital signature as the step where encryption is used, so the computational cost is also very low in comparison to other frameworks; it took about 0.26 seconds to declare the winner, but when we consider other frameworks they have higher complexity than framework 1, and it also effects on the average execution time for the same reason. If we see framework2, it takes about 0.32 seconds for ten experts under the same settings. When more complex steps containing SMPC and multiple steps of homomorphic encryption are introduced, such as in frameworks 3 and 4, the average execution times grow faster. For the third framework, the average execution time for ten experts is

0.332 seconds, and for the fourth framework, it took 0.349 seconds because of the introduction of the extra step of winner declaration by the employer in comparison to framework 3. SMPC increases the communication overhead as well as computation overhead, impacting overall execution time. Framework 3 and Framework 4 require the SMPC to be done before declaring the winner, hence a greater communication overhead, which is the reason for the extra execution time in these two models in comparison to the execution time of the first two frameworks.

The key size also greatly impacts the average execution time; the complexity of the function associated with homomorphic encryption increases upon increasing the encryption key size. In Figure 2, with the encryption key size of 256-bit, framework 2 took 0.3490 seconds, and Framework 3 had 0.450 seconds on ten experts, which is comparatively much more than what it was for the key size of 128-bit. Similarly, for framework four, it can be seen that execution time becomes 0.460 seconds for ten experts as it has one more step than framework three, as discussed above. When we change the key size to 512 bits, the time complexity goes tremendously high, which is 0.783 seconds and 1.104 seconds for ten experts in Framework 2 and Framework 3, respectively, and for Framework 4, it becomes 1.1413 seconds. The same is shown in Figure 3. This growth in average execution times on increasing the encryption key size is due to the use of homomorphic encryption, as described in the equation. Multiplication is one of the functions of HE and is performed on the ciphertexts to add the original input, but when we are using the 256-bit or 512-bit encryption key size, the ciphertext generated through the HE is equal to key size and performing multiplication to compute addition of the input on such ciphertext creates a great computational overhead increasing the average execution time of framework three and framework 4.

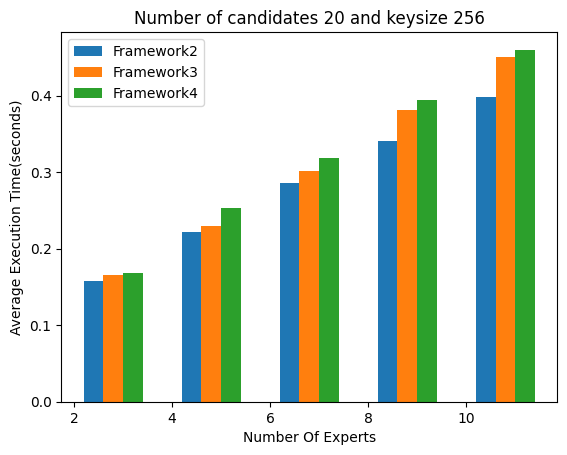


Figure 2: Comparison of the execution time of models 2,3 and 4 with an encryption key size of 256 bits

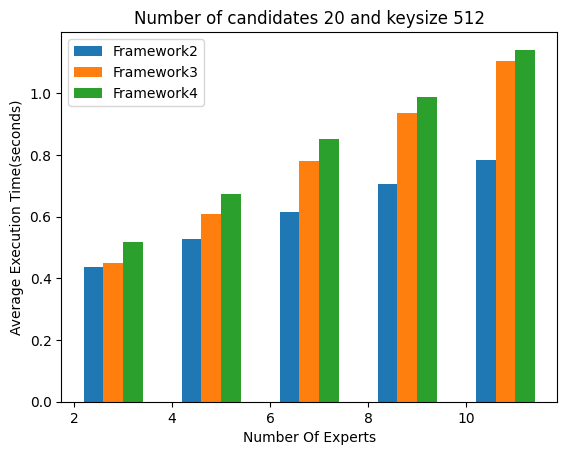


Figure 3: Comparison of the execution time of models 2, 3, and 4 with an encryption key size of 512 bits

**Experiment results for setting 2**

For the second setting, when we keep the number of experts constant at five and vary the number of candidates in [10, 20, 30, 40, 50], a similar time difference as of setting one between all four frameworks is found. Figure 4 shows the execution time difference between all four models with encryption key sizes of 128 bits, and Figure 5 and Figure 6 shows the execution times of frameworks 2, 3, and 4 for the encryption key sizes of 256 bits and 512 bits, respectively.

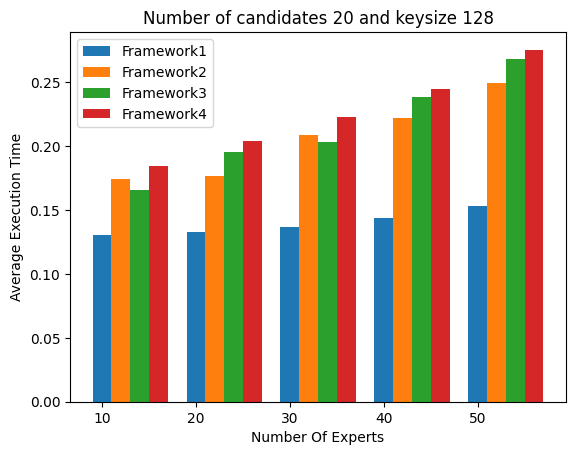


Fig 4: Comparing the execution time of all models with an encryption key size of 128-bit.

The comparison shows that our framework 1 takes an average execution time of 0.1529 for 50 candidates. Framework 2, Framework 3, and Framework 4 are taking 0.2492 seconds, 0.2686 seconds, and 0.2755, respectively, for 50 candidates.

Similarly, when we start varying the key sizes from 128 bits to 256 and 512 bits, a huge change is found in frameworks 2, 3, and 4 because of the same reasons as are already discussed in setting 1. Average Execution times of the same are shown in Figure 5 and Figure 6. See the average execution time for 256 bits for frameworks 2,3, and 4 is 0.39 seconds, 0.464 seconds, and 0.491 seconds, respectively, which increases rapidly on taking the encryption key size of 512 bits due to the same reasons discussed in setting one. Framework 2, 3, and 4 took 1.08, 1.33, 1.46 seconds respectively.

One key difference that can be observed between both settings is that average execution time increases rapidly with the increasing number of candidates in comparison to the increasing number of experts.

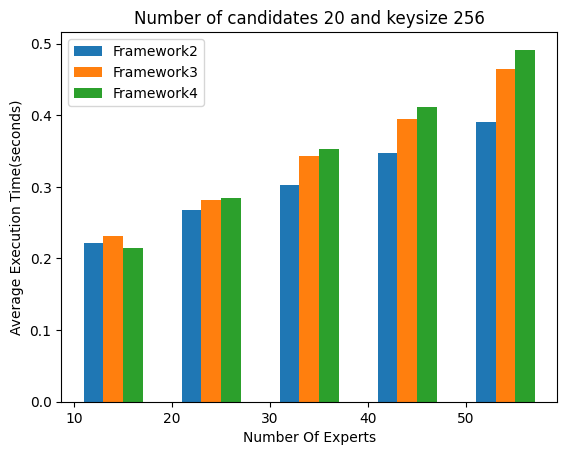


Figure 5: Comparison of the execution time of models 2, 3, and 4 with an encryption key size of 256 bits

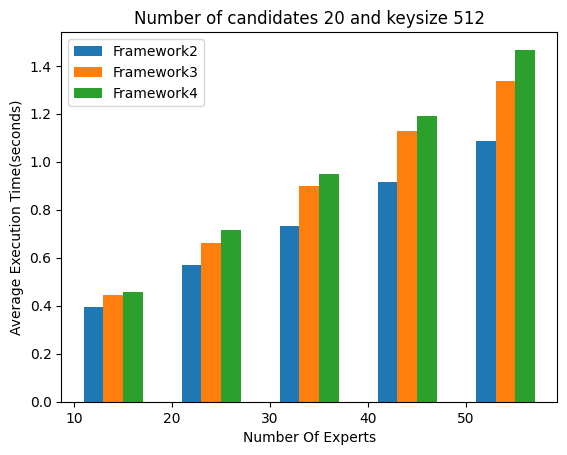


Figure 6: Comparison of the execution time of models 2, 3, and 4 with an encryption key size of 512 bits