Comparison between Neural Networks and traditional data structures primitives for indexing purposes

Raffaele, Giancarlo
Giosué, Lo Bosco
Domenico, Amato

[1] Dipartimento di Matematica e Informatica, Universitá degli studi di Palermo, ITALY.

Abstract. to be done

Keywords: neural network, binary search, hashing

1 Related Problems

1.1 Problem 1: Precedessor Search

Let X be set of *keys* drawn from a universe U, sorted according to the $\leq$ order relation defined on U. In what follows, we assume that each element of U can be represented with d bits. Moreover, we assume that if the binary format is such to represent an integer or a floating point, the arithmetic will be done accordingly. We also assume that the lexicographic ordering of the binary representation of each element in U preserves the order relation $\leq$. That is, a comparison between two elements of U can be done using either the binary representation as a string or a primitive comparison operation. For the special case of float IEEE 754 representation the lexicographic ordering preserve the order relation $\leq$ by flipping the sign bit of the representation. Moreover, we will consider each element of U either as a "number" in decimal notation, i.e., x, or as a vector in binary notation denoted as $\overrightarrow{x}$, according to its "internal format", i.e., integer or float. An analogous notation is followed for X. Our model is a unit-cost RAM with word size d. Arithmetic operations are consistent with element representation. That is, they will be float when at least one of the operands is floating point.

The *predecessor search problem* consist in finding the position of the largest key in X not greater than a given input key x.

Example 1. Let X be

$$X = \{2, 3, 4, 5, 12, 15, 18\}$$

Let us assume that 7 is the key to search. The predecessor search should return the position $i = 4$.

It is worth of mention that all of the techniques in the next two section apply verbatim to strings of finite length coded in binary. This case may turn out to be important.

Regression and Neural Nets

Consider a sorted set of keys X, with $|X| = n$. Let F be the *empirical cumulative distribution function* of elements of U with respect to (sample) X. That is, for each $z \in U$, $F(z) = \frac{|\{y \in X | y \leq z\}|}{n}$. Notice that knowledge of F provides a solution to our problem since, given an element z of U, only one evaluation of F provides $p(z) = \lceil F(z) \times n \rceil$ where $p(z)$ indicates the predecessor index. Moreover, F could be evaluated on $|U|$ elements. For this reason, the requested space used by any data structure is $\Omega(d2^d)$, i.e. the data structure have to be capable of distinguishing all the elements of $|U|$. However, F can also be first *estimated numerically*, based on knowledge of X, in order to obtain an implicit or explicit mathematical expression that we can use over and over again to solve our problem. Such a task can be phrased in terms of a regression problem. Indeed, our task is to find a function $\tilde{F}$ which gives a good approximation of F. This can be done iteratively, by properly refining the solution $\tilde{F}_i$ at each iteration i such that $E(\tilde{F}_i, F) \leq E(\tilde{F}_{i-1}, F)$, where E represents a chosen error function. Formally, we can set a tolerance δ and look for the $\tilde{F} = \tilde{F}_i$ such that $|E(\tilde{F}_i, F) - E(\tilde{F}_{i-1}, F)| \leq \delta$. This will allow to find the precision $\epsilon = max_j|\tilde{F}(x_j), F(x_j)|$ of the approximation that is used to restrict the interval of ~~search in "table" F to~~ solve the stated problem exactly. This is done by looking for $F(x)$ in the interval $[\tilde{F}(x) - \epsilon, \tilde{F}(x) + \epsilon]$. We now investigate how to find $\tilde{F}$ via regression using neural nets. We estimate the costs in time and space both for its determination and use.

Neural Nets Solution

We concentrate on *feed forward neural network* to determine $\tilde{F}$. For the purposes of this note, it is necessary to have a clear idea about space and time costs of the specific neural network model we want to adopt for the regression. In principle, the costs of a neural network can be distinguished into *preprocessing* and *query* costs. The factors that influence such costs are the following:

(1) ARCHITECTURE TOPOLOGY. It contributes as follows.
  – (1.a) **The atomic element of the neural network**. A scheme is provided in figure ??. Relevant are the basic operation on its $2d$ binary input lines and the activation via function f. The basic operation we consider here is the scalar product between the input vectors $\overrightarrow{w}$ and $\overrightarrow{x}$. The vectors $\overrightarrow{w}$ are named *weights*. Moreover, the adopted activation function f is the so called *reLU* i.e. $f(x) = max(0, x)$.
  – (1.b) **The number of hidden layers**. We assume that number to be K.
  – (1.c) **The number of atomic elements for each layer**. Including input and output as layers number *zero* and $K+1$, respectively, let those numbers be $n_0, n_1, n_2, .., n_K$.
(2) LEARNING ALGORITHM:
  – (2.a) **The error function** E, used to evaluate how close the approximation $\tilde{F}$ is to F. The most common choice, and the one adopted here, is the *mean square error* i.e. $E(\overrightarrow{A}, \overrightarrow{B}) = \frac{1}{m}\Sigma_{i=1}^{m}(a_i - b_i)^2$.
  – (2.b) **The incremental approximation strategy**. Starting from a "low quality" approximation $\tilde{F}_0$ of F, an iterative process produces better and better

approximations by successively reducing the error. Each step of this process is referred to as an *epoch*. The entire process is specified by the following parameters.
–– (2.a.1) **The learning rate** η. It influences the rate of convergence of the process. It can be set empirically or can be estimated via model validation techniques.
–– (2.a.2) **The Batch set** B, i..e. a subset of the "learning set" X of cardinality $m \leq n$. Note that the Batch set can be equal to X.
–– (2.a.3) **The tolerance** δ. It represents the level at which the residual error $|E(\bar{F}_i, F) - E(\bar{F}_{i-1}, F)|$ is small enough that the solution can be considered sufficiently accurate.
–– (2.a.4) **The Maximum number of epochs** N, i.e., when the process must stop. It may be an empirically chosen number, or more in general N may be thought of as depending on the two variables η and δ. Let $N(\eta, \delta)$ be the maximum number of epochs as a function of the two mentioned variables.

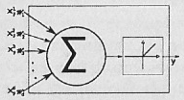

Fig. 1: The atomic element uses $(x^1, .., x^d)$ and $(w_1, .., w_d)$ as inputs. Each x^i is a binary digit while each w_i is a real value. The final output $y = max(0, \Sigma_{i=1}^{d} w_i x^i)$.

In particular, the preprocessing cost depends on both items described before, while query cost depends only on item 1.

Solution one: Zero Hidden Layer Topology
We have $K = 0$ hidden layers, as a consequence only one atomic element $n_0 = 1$. The adopted incremental approximation strategy is the so called *stochastic gradient descent algorithm*, which basically give a strategy in order to properly update the weights of each atomic elements of the network. In particular, it works in the following way:

- (1. Input): $\overrightarrow{X} = (\overrightarrow{x}_1, .., \overrightarrow{x}_n)$ matrix of size $d \times n$, batch size m, learning rate η, tolerance tol, maximum number of iterations N.
- (2. Step 0): set $\overrightarrow{W} = \overrightarrow{W_0}$ as a matrix of $1 \times d$ real random values. Set $i = 0$.
- (3. Step i): Let the batch matrix $\overrightarrow{B}_i$ of size $d \times m$ be obtained by choosing uniformly and at random m columns $j(i)_1, .., j(i)_m$ from $\overrightarrow{X}$. Let the target matrix of size $1 \times m$ be $\overrightarrow{T}_i = (F(\overrightarrow{x}_{j(i)_1}), .., F(\overrightarrow{x}_{j(i)_m}))$.
Update the weights in the following way:
$$\overrightarrow{W} = \overrightarrow{W} - \frac{\eta}{m} \overrightarrow{B_i} [\overrightarrow{W} \overrightarrow{B_i} - \overrightarrow{T}_i]^t \tag{1}$$
- Set $i = i + 1$
- if $(i < N)$ & $(max_i |F(\overrightarrow{x}_i) - \bar{F}(\overrightarrow{x}_i)| > \epsilon)$ goto 3, else exit.

The output of the *stochastic gradient descent algorithm* are the weights $\overrightarrow{W}$ computing after the last iteration of the approximation strategy. In terms of time and space costs, we have the following. The preprocessing cost of each epoch is $O(m*d)$ in time. For N epochs, the preprocessing cost is $O(N(\epsilon, \eta) * m * d)$ in time and $O(nd)$ in space. Note that we have considered N as dependent on ϵ and η taking into consideration that it can be computed by such quantities. The query time is $O(d)$ in time and $O(d)$ in space.

Solution two: Hidden Layers Topology with $K > 0$
In this case $n_k, k = 1, .., K$ are the atomic elements for each layer of the network. Let $\overrightarrow{W}^k$ be the matrix of the weights of size $n_{k-1} \times n_k$ where the generic element $\overrightarrow{W}^k_{i,j}$ is the weight from the atomic element j of the layer k to the atomic element j of the layer $k-1$.

inserire figura

The learning algorithm used for this kind of topology is the so called *back propagation algorithm* that uses at each epoch the *stochastic gradient descent algorithm* on each layer of the network, starting from the last one back to the first.

- (1. Input) : $\overrightarrow{X} = (\overrightarrow{x}_1, .., \overrightarrow{x}_n)$ matrix of size $d \times n$, batch size m, learning rate η, approximation ϵ, maximum number of iteration N.
- (2. Step 0): for each layer $k = 1, .., K$
set $\overrightarrow{W}^k = \overrightarrow{W_0^k}$ as a matrix of $n_k \times n_{k-1}$ real random values.
- (3. Step i): Let the batch matrix $\overrightarrow{B}_i$ of size $d \times m$ be obtained by choosing uniformly and at random m columns $j(i)_1, .., j(i)_m$ from $\overrightarrow{X}$. Let the target matrix of size $1 \times m$ be $\overrightarrow{T}_i = (F(\overrightarrow{x}_{j(i)_1}), .., F(\overrightarrow{x}_{j(i)_m}))$.
Let the matrix $\overrightarrow{Y}_i^k$ of size $n_k \times m$ be the *output matrix* at layer k of the network when using $\overrightarrow{B_i}$ as input. For the case $k = K$, $\overrightarrow{Y}_i^K = (\bar{F}(\overrightarrow{x}_{j(i)_1}), .., \bar{F}(\overrightarrow{x}_{j(i)_m}))$. For the case $k = 0$, $\overrightarrow{Y}_i^0 = B_i$.

compute

$$\delta_K = \overrightarrow{Y}_i^K - \overrightarrow{T_i}$$

– (4. Back propagation) for $k = K-1$ downto 1

$$\delta_k = \overrightarrow{W}^{k+1} \times \delta_{k+1}$$

– (5. Weight update) for $k = 1$ to K

$$\overrightarrow{W}^k = \overrightarrow{W}^k - \frac{\eta}{m}\overrightarrow{Y}_i^{k-1}[\delta_k]^t$$

– Set $i = i+1$
– if $(i < N)$ & $(max_i|F(\overrightarrow{x}_i) - \bar{F}(\overrightarrow{x}_i)| > \epsilon)$ goto 3, else exit.

The preprocessing cost for N epochs of this model is $O(N(\epsilon,\eta) * m * \Sigma_{k=1}^{K+1} n_k * n_{k-1})$ in time and $O(n * d + \Sigma_{k=1}^{K+1} n_k * n_{k-1})$ in space. The quesry cost is $O(\Sigma_{k=1}^{K+1} n_k * n_{k-1})$ in time and $O(\Sigma_{k=0}^{K+1} n_k)$ in space, with $n_0 = d, n_{K+1} = 1$.

Solution by use of data structures

Static Case [RG: *Da rivedere*] Using a special data structure, the predecessor search of an arbitrary element (in the sorted order of S) can be solved in constant time and small space, e.g., [1]. We take here the Longest Common Prefix (LCP) and Partial Compacted Trie (PaCo) based monotone minimal perfect hash functions since, according to [1], they are the best performing in practice. Both of them use the idea of *bucketing* i.e. a way of distributing the computation of the hash function into a fixed number of buckets. The main difference between the two approaches is on the distribution process: PaCo uses a partial compacted binary trie while LCP uses longest common prefixes. The relevant theoretic costs are as follows. The LCP based occupies $O(n\ logd)$ bits in space and has constant time for the query. The PaCo based are $O(n\ loglogd)$ in space and $O(logd)$ in query time. The implementation comes from the sux4j library (http://sux.di.unimi.it/) which offers a java library for basic succinct data structures to be used for several purposes including minimal perfect hashing. The LCP and Paco based can be used by means of the two java classes *LcpMonotoneMinimalPerfectHashFunction* and *PaCoTrieDistributorMonotoneMinimalPerfectHashFunction*.

Dynamic Case [RG: *Bisogna prendere in considerazione anche le versioni dinamiche [3]*].

Experiments We have set an experimental protocol with the final goal of comparing the efficacy of the solution provided by the neural network with respect to other implementation based on special data structures (see 1.1 for some details of these methods). In general, we generate u datasets drawn from a given distribution, i.e., files $f_1, f_2, ..., f_u$. The size of those datasets ranges from a minimum of $1K$ and (for technical reasons) a maximum of $2G$. In each file, each

integer is represented as a 64-bits string. Given file f_i, we generate 10 query datasets, with sizes starting at 10% of the size of f_i with increments of 10, extracting them from f_i uniformly and at random. In what follows, let $q_{i,j}$ be one of those datasets.

For both the NN and the Data Structures, we compute (a) preprocessiung time; (b) space. For the NN, we also compute the precision ϵ for each dataset. As for the query set $q_{i,j}$, we compute the avarage seach time. It is to be noted that, in the case of the NN, once that it is used to estimate the position of the element being searched, an additional binary search is required to complete the operation. Moreover, in the case of NN, we computde the average absolute error $E_{i,j}$ relative to a query $q_{i,j}$. It is defined as

$$E_{i,j} = \sum_{x \in q_{i,j}} \frac{|h(x) - h_p(x)|}{N} \quad (2)$$

where $h(x)$ is the real index of the key x, $h_p(x)$ is the index of x predicted by the method and N is the size of the query set $q_{i,j}$.

The uniform case. We have generated $u = 13$ datasets. The results are reported in **bla-uno per gli NN ed uno per le data structures.**

Table 1
Training Results. For each files we annotated the number of element n and the size in Memory s, for each method used we meseaured the time T taken by model for training, the Space S occupied by model and data structures and the maximum error ϵ

File	*NN1*	*NN2*	*NN3*	*Paco*	*Lcp*
f_1 s= 1,024 Kb n= 1,28x10^2	T= 2,438s S= 7,72x10^{-1}Kb ϵ= 1,4x10 *er*= 16,41%	T = 6,286x10s S= 1,992x10^{2}Kb ϵ= 5 *er*= 3,13%	T = 9,725x10s S= 5,934x10^{2}Kb ϵ= 5 *er*= 3,13%	T = 2,5x10^{-1}s S= 1,22x10^{-1}Kb	T = 2,57x10^{-1}s S= 1,98x10^{-1}Kb
f_2 s= 2,048 Kb n= 2,56x10^2	T= 1,049s S= 7,72x10^{-1}Kb ϵ= 3,1x10 *er*= 16,02%	T= 2,5945x10 s S= 1,992x10^{2}Kb ϵ= 1,2x10 *er*= 2,25%	T= 5,587x10 s S= 5,934x10^{2}Kb ϵ= 1,1x10 *er*=2,15%	T= 4x10^{-2} s S= 2,15x10^{-1}Kb	T= 6,1x10^{-2}s S= 3,72x10^{-1}Kb

Table 1 – continued from previous page

File	*NN*1	*NN*2	*NN*3	*Paco*	*Lcp*
f_3 s= 4,096Kb n= 5,12x10^2	T= 6,75x10^{-1}s S= 7,72x10^{-1}Kb ϵ= 6,9x10 *er*= 12,06%	T= 2,7458x10s S= 1,992x10^{2}Kb ϵ= 2,5x10 *er*= 2,1%	T= 4,12x10s S= 5,934x10^{2}Kb ϵ= 2,4x10 *er*= 1,78%	T= 3,7x10^{-2}s S= 4,2310^{-1}Kb	T= 5,9x10^{-2}s S= 7,1610^{-1}Kb
f_4 s= 8,192Kb n= 1,024x10^3	T= 6,69x10^{-1}s S= 7,72x10^{-1}Kb ϵ= 1,37x10^2 *er*= 10,54%	T= 2,608x10s S= 1,992x10^{2}Kb ϵ= 5,4x10 *er*= 1,94%	T= 3,768x10s S= 5,934x10^{2}Kb ϵ= 5,2x10 *er*= 1,92%	T= 4,4x10^{-2}s S = 8,27x10^{-1}Kb	T= 6,6x10^{-2}s S= 1,419 Kb
f_5 s= 1,638x10Kb n= 2,048x10^3	T= 1,106s S= 7,72x10^{-1}Kb ϵ= 2,82x10^2 *er*= 10,16%	T= 2,948x10s S= 1,992x10^{2}Kb ϵ= 1,03x10^2 *er*= 1,34%	T= 3,65x10s S= 5,934x10^{2}Kb ϵ= 1,06x10^2 *er*= 1,51%	T= 4,9x10^{-2}s S= 1,645Kb	T= 6,7x10^{-2}s S= 2,849Kb
f_6 s= 3,277x10Kb n= 4,096x10^3*Kb*	T= 9,84x10^{-1}s S= 7,72x10^{-1}Kb ϵ= 5,29x10^2 *er*= 10,02%	T= 3,157x10s S= 1,992x10^{2}Kb ϵ= 2,08x10^2 *er*= 1,19%	T= 4,756x10s S= 5,934x10^{2}Kb ϵ= 2,08x10^2 *er*= 1,09%	T= 5,3x10^{-2}s S= 3,273Kb	T= 8x10^{-2}s S= 5,744Kb

Table 1 – continued from previous page

File	*NN*1	*NN*2	*NN*3	*Paco*	*Lcp*
f_7 s= 6,554x10Kb n= 8,192x10^3	T= 3,68 s S= 7,72x10^{-1}Kb ϵ= 1,184x10^3 *er*= 11,15%	T= 3,537x10s S= 1,992x10^{2}Kb ϵ= 4,19x10^2 *er*= 0,91%	T= 9,153x10s S= 5,934x10^{2}Kb ϵ= 3,96x10^2 *er*= 0,92%	T= 6,9x10^{-2}s S= 6,572Kb	T= 9,2x10^{-2}s S= 1,196x10Kb
f_8 s = 1,311x10^2 n= 1,6384x10^4	T= 5,019s S= 7,72x10^{-1}Kb ϵ= 2,14x10^3 *er*= 11,85%	T= 4,306x10s S= 1,992x10^{2}Kb ϵ= 8,28x10^2 *er*= 0,81%	T= 8,237x10s S= 5,934x10^{2}Kb ϵ= 8,03x10^2 *er*= 0,76%	T= 9,7x10^{-2}s S= 1,31x10Kb	T= 9,7x10^{-2}s S= 2,402x10Kb
f_9 s= 1,048x10^{3}Kb n= 1,31072x10^5	T = 2,7589x10s S= 7,72x10^{-1}Kb ϵ= 1,8125x10^4 *er*= 11,06%	T = 2,164x10^{2}s S= 1,992x10^{2}Kb ϵ= 6,564x10^3 *er*= 0,77%	T = 7,563x10^{2}s S= 5,934x10^{2}Kb ϵ= 6,075x10^3 *er*= 0,75%	T = 3,41x10^{-1}s S= 1,042x10^{2}Kb	T = 3,36x10^{-1}s S= 1,955x10^{2}Kb
f_{10} s= 8,388x10^{3}Kb n= 1,048576x10^{6}Kb	T = 2,185x10^{2}s S= 7,72x10^{-1}Kb ϵ= 1,47016x10^5 *er*= 9,46%	T = 3,002x10^{3}s S= 1,992x10^{2}Kb ϵ= 5,1888x10^4 *er*= 0,85%	T = 3,122x10^{3}s S= 5,934x10^{2}Kb ϵ= 4,9388x10^4 *er*= 0,75%	T = 1,119s S= 8,307x10^{2}Kb	T = 9,91x10^{-1}s S= 1,591x10^{3}Kb

Table 1 – continued from previous page

File	$NN1$	$NN2$	$NN3$	$Paco$	Lcp
f_{11} s= 1,342x10^{5}Kb n= 1,6777216x10^{7}Kb	T= 4,574x10^{3}s S= 7,72x10^{-1}Kb ϵ= 2,38806x10^6 er= 9,76%	T= 1,77x10^{4}s S = 1,992x10^{2}Kb ϵ= 8,33885x10^5 er= 0,85%	T= 1,73x10^{4}s S= 5,934x10^{2}Kb ϵ= 7,823x10^5 er= 0,70%	T= 8,021s S= 1,322x10^{3}Kb	T= 6,578s S= 2,603x10^{4}Kb
f_{12} s= 1,073x10^{6}Kb n= 1,34217728x10^{8}Kb	T= 4,399x10^{3}s S= 7,72x10^{-1}Kb ϵ= 1,9468669x10^7 er= 9,50%	T= 4,539x10^{4}s S= 1,992x10^{2}Kb ϵ= 6,690328x10^6 er= 0,71%	T= 5,781x10^{4}s S= 5,934x10^{2}Kb ϵ= 6,306102x10^6 er= 0,70%	T= 6,328x10s S= 1,05x10^{5}Kb	T= 4,889x10s S= 2,162x10^{4}Kb
f_{13} s= 2,147x10^{6}Kb n= 2,68435456x10^{8}Kb	T= 4,532x10^{3}s S= 7,72x10^{-1}Kb ϵ= 37391524x10^7 er= 9,17%	T= 1,377x10^{5}s S = 1,992x10^{2}Kb ϵ= 1,3356089x10^7 er= 0,74%	T= 7,302x10^{4}s S= 5,934x10^{2}Kb ϵ= 1,2691526x10^6 er= 0,61%	T= 1,333x10^{2}s S= 2,086x10^{5}Kb	T= 1,11x10^{2}s S= 4,336x10^{4}Kb

Table 2
Prediction Results for query from 10% to 30%. TP_m is the mean time taken by the NN model to obtain an index with a certain error ϵ and T_m is the mean time to find an integer in the array, in the case of NN we used a binary search method in a vector with maximum length $2\epsilon + 1$

File	Method	10%	20%	30%
f_1	NN_1	$T_m = 3{,}33\text{x}10^{-1}$ s $TP_m = 1{,}096\text{x}10^{-2}$ s	$T_m = 7{,}5\text{x}10^{-4}$ s $TP_m = 2{,}852\text{x}10^{-5}$ s	$T_m = 8{,}728\text{x}10^{-4}$ s $TP_m = 1{,}076\text{x}10^{-5}$ s
	NN_2	$T_m = 1{,}681\text{x}10^{-2}$ s $TP_m = 1.417\text{x}10^{-3}$ s	$T_m = 6{,}477\text{x}10^{-4}$ s $TP_m = 2{,}441\text{x}10^{-5}$ s	$T_m = 7{,}36\text{x}10^{-4}$ s $TP_m = 1{,}276\text{x}10^{-5}$ s

Table 2 – continued from previous page

File	Method	10%	20%	30%
	NN_3	$T_m = 1{,}734\text{x}10^{-2}$ s $TP_m = 2{,}077\text{x}10^{-3}$ s	$T_m = 7{,}279\text{x}10^{-4}$ s $TP_m = 3{,}083\text{x}10^{-5}$ s	$T_m = 1{,}135\text{x}10^{-3}$ s $TP_m = 1{,}516\text{x}10^{-5}$ s
	$Paco$	$T_m = 5{,}401\text{x}10^{-5}$	$T_m = 9{,}422\text{x}10^{-6}$	$T_m = 9{,}819\text{x}10^{-6}$
	Lcp	$T_m = 3{,}957\text{x}10^{-5}$s	$T_m = 6{,}842\text{x}10^{-6}$s	$T_m = 7{,}774\text{x}10^{-6}$s
f_2	NN_1	$T_m = 2{,}82\text{x}10^{-2}$ s $TP_m = 1{,}055\text{x}10^{-3}$ s	$T_m = 7{,}816\text{x}10^{-4}$ s $TP_m = 1{,}561\text{x}10^{-5}$ s	$T_m = 1{,}158\text{x}10^{-3}$ s $TP_m = 8{,}382\text{x}10^{-5}$ s
	NN_2	$T_m = 2{,}845\text{x}10^{-2}$ s $TP_m = 1{,}195\text{x}10^{-3}$ s	$T_m = 1{,}05\text{x}10^{-3}$ s $TP_m = 1{,}099\text{x}10^{-5}$ s	$T_m = 1{,}256\text{x}10^{-3}$ s $TP_m = 2{,}396\text{x}10^{-5}$ s
	NN_3	$T_m = 3{,}032\text{x}10^{-2}$ s $TP_m = 1{,}26\text{x}10^{-3}$ s	$T_m = 9{,}794\text{x}10^{-4}$ s $TP_m = 9{,}495\text{x}10^{-5}$ s	$T_m = 1{,}519\text{x}10^{-3}$ s $TP_m = 1{,}358\text{x}10^{-5}$ s
	$Paco$	$T_m = 5{,}255\text{x}10^{-6}$ s	$T_m = 3{,}332\text{x}10^{-6}$ s	$T_m = 2{,}692\text{x}10^{-6}$ s
	Lcp	$T_m = 4{,}5\text{x}10^{-6}$s	$T_m = 2{,}255\text{x}10^{-6}$s	$T_m = 1{,}811\text{x}10^{-5}$s
f_3	NN_1	$T_m = 4{,}025\text{x}10^{-2}$ s $TP_m = 8{,}097\text{x}10^{-4}$ s	$T_m = 1{,}312\text{x}10^{-3}$ s $TP_m = 1{,}014\text{x}10^{-5}$ s	$T_m = 1{,}852\text{x}10^{-3}$ s $TP_m = 8{,}141\text{x}10^{-6}$ s
	NN_2	$T_m = 4{,}268\text{x}10^{-2}$ s $TP_m = 8{,}956\text{x}10^{-4}$ s	$T_m = 1{,}686\text{x}10^{-3}$ s $TP_m = 9{,}803\text{x}10^{-6}$ s	$T_m = 1{,}912\text{x}10^{-3}$ s $TP_m = 8{,}404\text{x}10^{-6}$ s
	NN_3	$T_m = 4{,}625\text{x}10^{-2}$ s $TP_m = 9{,}064\text{x}10^{-4}$ s	$T_m = 1{,}639\text{x}10^{-3}$ s $TP_m = 9{,}238\text{x}10^{-6}$ s	$T_m = 1{,}924\text{x}10^{-3}$ s $TP_m = 7{,}816\text{x}10^{-6}$ s
	$Paco$	$T_m = 2{,}257\text{x}10^{-6}$ s	$T_m = 1{,}309\text{x}10^{-6}$ s	$T_m = 9{,}815\text{x}10^{-7}$ s
	Lcp	$T_m = 1{,}766\text{x}10^{-6}$s	$T_m = 7{,}132\text{x}10^{-7}$s	$T_m = 1{,}571\text{x}10^{-6}$s
f_4	NN_1	$T_m = 5{,}637\text{x}10^{-2}$ s $TP_m = 6{,}019\text{x}10^{-4}$ s	$T_m = 2{,}12\text{x}10^{-3}$ s $TP_m = 7{,}916\text{x}10^{-6}$ s	$T_m = 2{,}87\text{x}10^{-3}$ s $TP_m = 4{,}650\text{x}10^{-6}$ s
	NN_2	$T_m = 5{,}723\text{x}10^{-2}$ s $TP_m = 6{,}191\text{x}10^{-4}$ s	$T_m = 2{,}621\text{x}10^{-3}$ s $TP_m = 7{,}808\text{x}10^{-6}$ s	$T_m = 3{,}052\text{x}10^{-3}$ s $TP_m = 5{,}772\text{x}10^{-6}$ s
	NN_3	$T_m = 6{,}087\text{x}10^{-2}$ s $TP_m = 6{,}548\text{x}10^{-4}$ s	$T_m = 2{,}409\text{x}10^{-3}$ s $TP_m = 7{,}703\text{x}10^{-6}$ s	$T_m = 2{,}789\text{x}10^{-3}$ s $TP_m = 5{,}822\text{x}10^{-6}$ s

Table 2 – continued from previous page

File	Method	10%	20%	30%
	Paco	$T_m = 1{,}019\text{x}10^{-6}$ s	$T_m = 6{,}179\text{x}10^{-7}$ s	$T_m = 6{,}063\text{x}10^{-7}$ s
	Lcp	$T_m = 6{,}423\text{x}10^{-7}$s	$T_m = 3{,}789\text{x}10^{-7}$s	$T_m = 4{,}734\text{x}10^{-7}$s
f_5	NN_1	$T_m = 7{,}549\text{x}10^{-2}$ s $TP_m = 4{,}118\text{x}10^{-4}$ s	$T_m = 3{,}205\text{x}10^{-3}$ s $TP_m = 6{,}251\text{x}10^{-6}$ s	$T_m = 4{,}363\text{x}10^{-3}$ s $TP_m = 6{,}247\text{x}10^{-6}$ s
	NN_2	$T_m = 7{,}685\text{x}10^{-2}$ s $TP_m = 4{,}521\text{x}10^{-4}$ s	$T_m = 3{,}42\text{x}10^{-3}$ s $TP_m = 5{,}344\text{x}10^{-6}$ s	$T_m = 4{,}638\text{x}10^{-3}$ s $TP_m = 5{,}602\text{x}10^{-6}$ s
	NN_3	$T_m = 7{,}99\text{x}10^{-2}$ s $TP_m = 4{,}201\text{x}10^{-4}$ s	$T_m = 3{,}497\text{x}10^{-3}$ s $TP_m = 7{,}188\text{x}10^{-6}$ s	$T_m = 5{,}296\text{x}10^{-3}$ s $TP_m = 5{,}488\text{x}10^{-6}$ s
	Paco	$T_m = 3{,}871\text{x}10^{-6}$ s	$T_m = 3{,}012\text{x}10^{-6}$ s	$T_m = 1{,}142\text{x}10^{-6}$ s
	Lcp	$T_m = 1{,}366\text{x}10^{-6}$s	$T_m = 1{,}281\text{x}10^{-6}$s	$T_m = 6{,}645\text{x}10^{-7}$s
f_6	NN_1	$T_m = 9{,}632\text{x}10^{-2}$ s $TP_m = 2{,}693\text{x}10^{-4}$ s	$T_m = 6{,}048\text{x}10^{-3}$ s $TP_m = 5{,}782\text{x}10^{-6}$ s	$T_m = 8{,}742\text{x}10^{-3}$ s $TP_m = 5{,}43\text{x}10^{-6}$ s
	NN_2	$T_m = 9{,}783\text{x}10^{-2}$ s $TP_m = 2{,}542\text{x}10^{-4}$ s	$T_m = 6{,}26\text{x}10^{-3}$ s $TP_m = 5{,}097\text{x}10^{-6}$ s	$T_m = 9{,}096\text{x}10^{-3}$ s $TP_m = 5{,}506\text{x}10^{-6}$ s
	NN_3	$T_m = 1{,}044\text{x}10^{-1}$ s $TP_m = 2{,}69\text{x}10^{-4}$ s	$T_m = 7{,}122\text{x}10^{-3}$ s $TP_m = 5{,}505\text{x}10^{-6}$ s	$T_m = 9{,}599\text{x}10^{-3}$ s $TP_m = 6{,}911\text{x}10^{-6}$ s
	Paco	$T_m = 7{,}431\text{x}10^{-7}$ s	$T_m = 6{,}218\text{x}10^{-7}$ s	$T_m = 9{,}325\text{x}10^{-6}$ s
	Lcp	$T_m = 4{,}893\text{x}10^{-7}$ s	$T_m = 3{,}94\text{x}10^{-7}$s	$T_m = 3{,}901\text{x}10^{-7}$s
f_7	NN_1	$T_m = 1{,}217\text{x}10^{-1}$ s $TP_m = 1{,}611\text{x}10^{-4}$ s	$T_m = 1{,}018\text{x}10^{-2}$ s $TP_m = 4{,}853\text{x}10^{-6}$ s	$T_m = 1{,}534\text{x}10^{-2}$ s $TP_m = 5{,}343\text{x}10^{-6}$ s
	NN_2	$T_m = 1{,}231\text{x}10^{-1}$ s $TP_m = 1{,}618\text{x}10^{-4}$ s	$T_m = 1{,}065\text{x}10^{-2}$ s $TP_m = 6{,}139\text{x}10^{-6}$ s	$T_m = 1{,}6\text{x}10^{-2}$ s $TP_m = 5{,}23\text{x}10^{-6}$ s
	NN_3	$T_m = 1{,}287\text{x}10^{-1}$ s $TP_m = 1{,}7\text{x}10^{-4}$ s	$T_m = 1{,}267\text{x}10^{-2}$ s $TP_m = 6{,}526\text{x}10^{-6}$ s	$T_m = 1{,}731\text{x}10^{-2}$ s $TP_m = 6{,}09\text{x}10^{-6}$ s
	Paco	$T_m = 3{,}719\text{x}10^{-7}$ s	$T_m = 3{,}671\text{x}10^{-7}$ s	$T_m = 3{,}575\text{x}10^{-7}$ s
	Lcp	$T_m = 3{,}241\text{x}10^{-7}$s	$T_m = 3{,}079\text{x}10^{-7}$s	$T_m = 3{,}056\text{x}10^{-7}$s

Table 2 – continued from previous page

File	Method	10%	20%	30%
f_8	NN_1	$T_m = 1{,}51\text{x}10^{-1}$ s $TP_m = 1{,}024\text{x}10^{-4}$ s	$T_m = 1{,}96\text{x}10^{-2}$ s $TP_m = 5{,}501\text{x}10^{-6}$ s	$T_m = 2{,}709\text{x}10^{-2}$ s $TP_m = 5{,}748\text{x}10^{-6}$ s
	NN_2	$T_m = 1{,}554\text{x}10^{-1}$ s $TP_m = 1{,}047\text{x}10^{-4}$ s	$T_m = 2{,}049\text{x}10^{-2}$ s $TP_m = 6{,}298\text{x}10^{-6}$ s	$T_m = 3{,}138\text{x}10^{-2}$ s $TP_m = 6{,}344\text{x}10^{-6}$ s
	NN_3	$T_m = 1{,}613\text{x}10^{-1}$ s $TP_m = 1{,}057\text{x}10^{-4}$ s	$T_m = 2{,}222\text{x}10^{-2}$ s $TP_m = 6{,}168\text{x}10^{-6}$ s	$T_m = 3{,}348\text{x}10^{-2}$ s $TP_m = 6{,}251\text{x}10^{-6}$ s
	Paco	$T_m = 3{,}974\text{x}10^{-7}$ s	$T_m = 3{,}845\text{x}10^{-7}$ s	$T_m = 3{,}438\text{x}10^{-7}$ s
	Lcp	$T_m = 1{,}717\text{x}10^{-7}$ s	$T_m = 1{,}478\text{x}10^{-7}$ s	$T_m = 2{,}089\text{x}10^{-7}$ s
f_9	NN_1	$T_m = 2{,}322\text{x}10^{-1}$ s $TP_m = 1{,}94\text{x}10^{-5}$ s	$T_m = 1{,}28\text{x}10^{-1}$ s $TP_m = 5{,}298\text{x}10^{-6}$ s	$T_m = 1{,}908\text{x}10^{-1}$ s $TP_m = 5{,}181\text{x}10^{-6}$ s
	NN_2	$T_m = 2{,}431\text{x}10^{-1}$ s $TP_m = 1{,}949\text{x}10^{-5}$ s	$T_m = 1{,}455\text{x}10^{-1}$ s $TP_m = 5{,}89\text{x}10^{-6}$ s	$T_m = 2{,}039\text{x}10^{-1}$ s $TP_m = 5{,}634\text{x}10^{-6}$ s
	NN_3	$T_m = 2{,}54\text{x}10^{-1}$ s $TP_m = 2{,}085\text{x}10^{-5}$ s	$T_m = 1{,}573\text{x}10^{-1}$ s $TP_m = 6{,}93\text{x}10^{-6}$ s	$T_m = 2{,}238\text{x}10^{-1}$ s $TP_m = 6{,}049\text{x}10^{-6}$ s
	Paco	$T_m = 1{,}512\text{x}10^{-6}$ s	$T_m = 9{,}543\text{x}10^{-7}$ s	$T_m = 1{,}086\text{x}10^{-6}$ s
	Lcp	$T_m = 1{,}559\text{x}10^{-7}$ s	$T_m = 1{,}732\text{x}10^{-7}$ s	$T_m = 1{,}45\text{x}10^{-7}$ s
f_{10}	NN_1	$T_m = 7{,}058\text{x}10^{-1}$ s $TP_m = 7{,}199\text{x}10^{-6}$ s	$T_m = 9{,}88\text{x}10^{-1}$ s $TP_m = 4{,}951\text{x}10^{-6}$ s	$T_m = 1{,}536$ s $TP_m = 5{,}125\text{x}10^{-6}$ s
	NN_2	$T_m = 7{,}435\text{x}10^{-1}$ s $TP_m = 7{,}555\text{x}10^{-6}$ s	$T_m = 1{,}064$ s $TP_m = 5{,}687\text{x}10^{-6}$ s	$T_m = 1{,}634$ s $TP_m = 5{,}427\text{x}10^{-6}$ s
	NN_3	$T_m = 7{,}996\text{x}10^{-1}$ s $TP_m = 8{,}034\text{x}10^{-6}$ s	$T_m = 1{,}14$ s $TP_m = 5{,}742\text{x}10^{-6}$ s	$T_m = 1{,}688$ s $TP_m = 5{,}85\text{x}10^{-6}$ s
	Paco	$T_m = 6{,}313\text{x}10^{-7}$ s	$T_m = 5{,}45\text{x}10^{-7}$ s	$T_m = 5{,}404\text{x}10^{-7}$ s
	Lcp	$T_m = 2{,}038\text{x}10^{-7}$ s	$T_m = 1{,}583\text{x}10^{-7}$ s	$T_m = 1{,}584\text{x}10^{-7}$ s
f_{11}	NN_1	$T_m =$ NA $TP_m = 5{,}334\text{x}10^{-6}$ s	$T_m =$ NA $TP_m = 5{,}002\text{x}10^{-6}$ s	$T_m =$ NA $TP_m = 5{,}041\text{x}10^{-6}$ s

Table 2 – continued from previous page

File	Method	10%	20%	30%
	NN_2	$T_m = NA$ $TP_m = 5{,}583\text{x}10^{-6}$ s	$T_m = NA$ $TP_m = 5{,}382\text{x}10^{-6}$ s	$T_m = NA$ $TP_m = 5{,}29\text{x}10^{-6}$ s
	NN_3	$T_m = NA$ $TP_m = 5{,}806\text{x}10^{-6}$ s	$T_m = NA$ $TP_m = 5{,}719\text{x}10^{-6}$ s	$T_m = NA$ $TP_m = 5{,}663\text{x}10^{-6}$ s
	$Paco$	$T_m = 7{,}399\text{x}10^{-7}$ s	$T_m = 7{,}356\text{x}10^{-7}$ s	$T_m = 7{,}375\text{x}10^{-7}$ s
	Lcp	$T_m = 2{,}383\text{x}10^{-7}$ s	$T_m = 2{,}402\text{x}10^{-7}$ s	$T_m = 2{,}382\text{x}10^{-7}$ s
f_{12}	NN_1	$T_m = NA$ $TP_m = 4{,}859\text{x}10^{-6}$ s	$T_m = NA$ $TP_m = 4{,}874\text{x}10^{-6}$ s	$T_m = NA$ $TP_m = 4{,}9\text{x}10^{-6}$ s
	NN_2	$T_m = NA$ $TP_m = 5{,}245\text{x}10^{-6}$ s	$T_m = NA$ $TP_m = 5{,}353\text{x}10^{-6}$ s	$T_m = NA$ $TP_m = 5{,}239\text{x}10^{-6}$ s
	NN_3	$T_m = NA$ $TP_m = 5{,}775\text{x}10^{-6}$ s	$T_m = NA$ $TP_m = 5{,}98\text{x}10^{-6}$ s	$T_m = NA$ $TP_m = 5{,}592\text{x}10^{-6}$ s
	$Paco$	$T_m = 8{,}451\text{x}10^{-7}$ s	$T_m = 8{,}474\text{x}10^{-7}$ s	$T_m = 8{,}431\text{x}10^{-7}$ s
	Lcp	$T_m = 2{,}402\text{x}10^{-7}$ s	$T_m = 2{,}371\text{x}10^{-7}$ s	$T_m = 2{,}377\text{x}10^{-7}$ s
f_{13}	NN_1	$T_m = NA$ $TP_m = 5{,}084\text{x}10^{-6}$ s	$T_m = NA$ $TP_m = 5{,}051\text{x}10^{-6}$ s	$T_m = NA$ $TP_m = 5{,}056\text{x}10^{-6}$ s
	NN_2	$T_m = NA$ $TP_m = 5{,}368\text{x}10^{-6}$ s	$T_m = NA$ $TP_m = 5{,}403\text{x}10^{-6}$ s	$T_m = NA$ $TP_m = 5{,}4\text{x}10^{-6}$ s
	NN_3	$T_m = NA$ $TP_m = 5{,}722\text{x}10^{-6}$ s	$T_m = NA$ $TP_m = 5{,}715\text{x}10^{-6}$ s	$T_m = NA$ $TP_m = 5{,}711\text{x}10^{-6}$ s
	$Paco$	$T_m = 8{,}972\text{x}10^{-7}$ s	$T_m = 9{,}035\text{x}10^{-7}$ s	$T_m = 9{,}064\text{x}10^{-7}$ s
	Lcp	$T_m = 2{,}579\text{x}10^{-7}$ s	$T_m = 2{,}594\text{x}10^{-7}$ s	$T_m = 2{,}576\text{x}10^{-7}$ s

Table 3
Prediction Results for query from 40% to 60%. Tp_m is the mean time taken by the NN model to obtain an index with a certain error ϵ and T_m is the mean time to find an integer in the array

File	Method	40%	50%	60%
f_1	NN_1	$T_m = 7{,}057\text{x}10^{-4}$ s $TP_m = 9{,}943\text{x}10^{-6}$ s	$T_m = 8{,}217\text{x}10^{-4}$ s $TP_m = 5{,}022\text{x}10^{-6}$ s	$T_m = 1{,}214\text{x}10^{-2}$ s $TP_m = 8{,}564\text{x}10^{-6}$ s
	NN_2	$T_m = 8{,}735\text{x}10^{-4}$ s $TP_m = 9{,}64\text{x}10^{-6}$ s	$T_m = 8{,}966\text{x}10^{-4}$ s $TP_m = 4{,}992\text{x}10^{-6}$ s	$T_m = 1{,}275\text{x}10^{-3}$ s $TP_m = 9{,}389\text{x}10^{-6}$ s
	NN_3	$T_m = 8{,}722\text{x}10^{-4}$ s $TP_m = 8{,}728\text{x}10^{-6}$ s	$T_m = 8{,}232\text{x}10^{-4}$ s $TP_m = 6{,}221\text{x}10^{-6}$ s	$T_m = 1{,}572\text{x}10^{-3}$ s $TP_m = 9{,}364\text{x}10^{-6}$ s
	$Paco$	$T_m = 1{,}039\text{x}10^{-5}$	$T_m = 8{,}973\text{x}10^{-6}$	$T_m = 8{,}663\text{x}10^{-6}$
	Lcp	$T_m = 7{,}231\text{x}10^{-6}$s	$T_m = 6{,}883\text{x}10^{-6}$s	$T_m = 7{,}152\text{x}10^{-6}$s
f_2	NN_1	$T_m = 1{,}439\text{x}10^{-3}$ s $TP_m = 8{,}623\text{x}10^{-6}$ s	$T_m = 1{,}271\text{x}10^{-3}$ s $TP_m = 5{,}223\text{x}10^{-6}$ s	$T_m = 1{,}957\text{x}10^{-3}$ s $TP_m = 5{,}527\text{x}10^{-6}$ s
	NN_2	$T_m = 1{,}336\text{x}10^{-3}$ s $TP_m = 7{,}611\text{x}10^{-6}$ s	$T_m = 1{,}398\text{x}10^{-3}$ s $TP_m = 5{,}947\text{x}10^{-6}$ s	$T_m = 1{,}833\text{x}10^{-3}$ s $TP_m = 7{,}738\text{x}10^{-6}$ s
	NN_3	$T_m = 1{,}339\text{x}10^{-3}$ s $TP_m = 8{,}192\text{x}10^{-6}$ s	$T_m = 1{,}407\text{x}10^{-3}$ s $TP_m = 5{,}426\text{x}10^{-6}$ s	$T_m = 1{,}863\text{x}10^{-3}$ s $TP_m = 7{,}479\text{x}10^{-6}$ s
	$Paco$	$T_m = 2{,}374\text{x}10^{-6}$ s	$T_m = 2{,}454\text{x}10^{-6}$ s	$T_m = 1{,}99\text{x}10^{-6}$ s
	Lcp	$T_m = 1{,}885\text{x}10^{-6}$s	$T_m = 2{,}041\text{x}10^{-6}$s	$T_m = 1{,}862\text{x}10^{-6}$s
f_3	NN_1	$T_m = 2{,}474\text{x}10^{-3}$ s $TP_m = 5{,}144\text{x}10^{-6}$ s	$T_m = 2{,}416\text{x}10^{-3}$ s $TP_m = 6{,}095\text{x}10^{-6}$ s	$T_m = 2{,}423\text{x}10^{-3}$ s $TP_m = 4{,}443\text{x}10^{-6}$ s
	NN_2	$T_m = 2{,}271\text{x}10^{-3}$ s $TP_m = 5{,}8\text{x}10^{-6}$ s	$T_m = 2{,}152\text{x}10^{-3}$ s $TP_m = 6{,}01\text{x}10^{-6}$ s	$T_m = 2{,}46\text{x}10^{-3}$ s $TP_m = 5{,}196\text{x}10^{-6}$ s
	NN_3	$T_m = 2{,}4\text{x}10^{-3}$ s $TP_m = 7{,}287\text{x}10^{-6}$ s	$T_m = 2{,}394\text{x}10^{-3}$ s $TP_m = 5{,}401\text{x}10^{-6}$ s	$T_m = 2{,}64\text{x}10^{-3}$ s $TP_m = 5{,}809\text{x}10^{-6}$ s
	$Paco$	$T_m = 8{,}316\text{x}10^{-7}$ s	$T_m = 6{,}964\text{x}10^{-7}$ s	$T_m = 6{,}883\text{x}10^{-7}$ s
	Lcp	$T_m = 7{,}604\text{x}10^{-7}$s	$T_m = 6{,}905\text{x}10^{-7}$s	$T_m = 8{,}308\text{x}10^{-7}$s

Table 3 – continued from previous page

File	Method	40%	50%	60%
f_4	NN_1	$T_m = 3,497x10^{-3}$ s $TP_m = 5,692x10^{-6}$ s	$T_m = 3,399x10^{-3}$ s $TP_m = 5,859x10^{-6}$ s	$T_m = 5,043x10^{-3}$ s $TP_m = 5,823x10^{-6}$ s
	NN_2	$T_m = 3,537x10^{-3}$ s $TP_m = 5,596x10^{-6}$ s	$T_m = 4,288x10^{-3}$ s $TP_m = 7,025x10^{-6}$ s	$T_m = 4,758x10^{-3}$ s $TP_m = 5,281x10^{-6}$ s
	NN_3	$T_m = 3,714x10^{-3}$ s $TP_m = 7,125x10^{-6}$ s	$T_m = 4,275x10^{-3}$ s $TP_m = 8,099x10^{-6}$ s	$T_m = 4,842x10^{-3}$ s $TP_m = 5,533x10^{-6}$ s
	$Paco$	$T_m = 6,459x10^{-7}$ s	$T_m = 7,983x10^{-7}$ s	$T_m = 6,11x10^{-7}$ s
	Lcp	$T_m = 3,441x10^{-7}$s	$T_m = 6,22x10^{-7}$s	$T_m = 3,379x10^{-7}$s
f_5	NN_1	$T_m = 5,353x10^{-3}$ s $TP_m = 4,77x10^{-6}$ s	$T_m = 6,015x10^{-3}$ s $TP_m = 4,558x10^{-6}$ s	$T_m = 7,672x10^{-3}$ s $TP_m = 4,555x10^{-6}$ s
	NN_2	$T_m = 5,78x10^{-3}$ s $TP_m = 6,062x10^{-6}$ s	$T_m = 7,092x10^{-3}$ s $TP_m = 5,615x10^{-6}$ s	$T_m = 8,093x10^{-3}$ s $TP_m = 6,733x10^{-6}$ s
	NN_3	$T_m = 6,219x10^{-3}$ s $TP_m = 5,335x10^{-6}$ s	$T_m = 7,198x10^{-3}$ s $TP_m = 5,913x10^{-6}$ s	$T_m = 8,751x10^{-3}$ s $TP_m = 6,081x10^{-6}$ s
	$Paco$	$T_m = 1,077x10^{-6}$ s	$T_m = 1,143x10^{-6}$ s	$T_m = 1.046x10^{-6}$ s
	Lcp	$T_m = 2,871x10^{-7}$s	$T_m = 3,251x10^{-7}$s	$T_m = 2,893x10^{-7}$s
f_6	NN_1	$T_m = 1,177x10^{-2}$ s $TP_m = 5,038x10^{-6}$ s	$T_m = 1,116x10^{-2}$ s $TP_m = 4,797x10^{-6}$ s	$T_m = 1,553x10^{-2}$ s $TP_m = 4,751x10^{-6}$ s
	NN_2	$T_m = 1,136x10^{-2}$ s $TP_m = 5,265x10^{-6}$ s	$T_m = 1,428x10^{-2}$ s $TP_m = 5,933x10^{-6}$ s	$T_m = 1,506x10^{-2}$ s $TP_m = 6,424x10^{-6}$ s
	NN_3	$T_m = 1,337x10^{-2}$ s $TP_m = 6,315x10^{-6}$ s	$T_m = 1,357x10^{-2}$ s $TP_m = 6,137x10^{-6}$ s	$T_m = 1,76x10^{-2}$ s $TP_m = 7,06x10^{-6}$ s
	$Paco$	$T_m = 5,512x10^{-7}$ s	$T_m = 5,968x10^{-7}$ s	$T_m = 6,569x10^{-7}$ s
	Lcp	$T_m = 3,925x10^{-7}$s	$T_m = 4,128x10^{-7}$s	$T_m = 3,935x10^{-7}$s
f_7	NN_1	$T_m = 1,954x10^{-2}$ s $TP_m = 5,439x10^{-6}$ s	$T_m = 2,429x10^{-2}$ s $TP_m = 5,004x10^{-6}$ s	$T_m = 2,541x10^{-2}$ s $TP_m = 5,422x10^{-6}$ s

Table 3 – continued from previous page

File	Method	40%	50%	60%
	NN_2	$T_m = 2,162x10^{-2}$ s $TP_m = 5,816x10^{-6}$ s	$T_m = 2,419x10^{-2}$ s $TP_m = 5,916x10^{-6}$ s	$T_m = 2,864x10^{-2}$ s $TP_m = 5,42x10^{-6}$ s
	NN_3	$T_m = 2,216x10^{-2}$ s $TP_m = 6,651x10^{-6}$ s	$T_m = 2,838x10^{-2}$ s $TP_m = 6,449x10^{-6}$ s	$T_m = 3,166x10^{-2}$ s $TP_m = 5,771x10^{-6}$ s
	$Paco$	$T_m = 3,733x10^{-7}$ s	$T_m = 3,585x10^{-7}$ s	$T_m = 4,128x10^{-7}$ s
	Lcp	$T_m = 3,048x10^{-7}$s	$T_m = 3,116x10^{-7}$s	$T_m = 2,646x10^{-7}$s
f_8	NN_1	$T_m = 3,582x10^{-2}$ s $TP_m = 5,315x10^{-6}$ s	$T_m = 4,394x10^{-2}$ s $TP_m = 5,202x10^{-6}$ s	$T_m = 5,037x10^{-2}$ s $TP_m = 5,41x10^{-6}$ s
	NN_2	$T_m = 3,952x10^{-2}$ s $TP_m = 6,027x10^{-6}$ s	$T_m = 4,747x10^{-2}$ s $TP_m = 5,612x10^{-6}$ s	$T_m = 5,537x10^{-2}$ s $TP_m = 5,725x10^{-6}$ s
	NN_3	$T_m = 4,216x10^{-2}$ s $TP_m = 6,216x10^{-6}$ s	$T_m = 4,859x10^{-2}$ s $TP_m = 6,702x10^{-6}$ s	$T_m = 5,975x10^{-2}$ s $TP_m = 6,27x10^{-6}$ s
	$Paco$	$T_m = 3,644x10^{-7}$ s	$T_m = 4,173x10^{-7}$ s	$T_m = 5,651x10^{-7}$ s
	Lcp	$T_m = 2,142x10^{-7}$ s	$T_m = 2,142x10^{-7}$ s	$T_m = 2,138x10^{-7}$ s
f_9	NN_1	$T_m = 2,581x10^{-1}$ s $TP_m = 5,102x10^{-6}$ s	$T_m = 3,12x10^{-1}$ s $TP_m = 5,156x10^{-6}$ s	$T_m = 3,892x10^{-1}$ s $TP_m = 4,919x10^{-6}$ s
	NN_2	$T_m = 2,691x10^{-1}$ s $TP_m = 5,554x10^{-6}$ s	$T_m = 3,455x10^{-1}$ s $TP_m = 5,661x10^{-6}$ s	$T_m = 4,274x10^{-1}$ s $TP_m = 5,746x10^{-6}$ s
	NN_3	$T_m = 2,967x10^{-1}$ s $TP_m = 6,105x10^{-6}$ s	$T_m = 3,536x10^{-1}$ s $TP_m = 6,006x10^{-6}$ s	$T_m = 4,626x10^{-1}$ s $TP_m = 5,826x10^{-6}$ s
	$Paco$	$T_m = 7,382x10^{-7}$ s	$T_m = 6,25x10^{-7}$ s	$T_m = 4,304x10^{-7}$ s
	Lcp	$T_m = 2,236x10^{-7}$ s	$T_m = 2,347x10^{-7}$ s	$T_m = 1,811x10^{-7}$ s
f_{10}	NN_1	$T_m = 2,049$ s $TP_m = 5,261x10^{-6}$ s	$T_m = 2,525$ s $TP_m = 5,157x10^{-6}$ s	$T_m = 2,981$ s $TP_m = 5,016x10^{-6}$ s
	NN_2	$T_m = 2,142$ s $TP_m = 5,447x10^{-6}$ s	$T_m = 2,73$ s $TP_m = 5,331x10^{-6}$ s	$T_m = 3,186$ s $TP_m = 5,514x10^{-6}$ s

Table 3 – continued from previous page

File	Method	40%	50%	60%
	NN_3	$T_m = 2,274$ s $TP_m = 5,892\text{x}10^{-6}$ s	$T_m = 2,841$ s $TP_m = 5,939\text{x}10^{-6}$ s	$T_m = 3,389$ s $TP_m = 6,067\text{x}10^{-6}$ s
	$Paco$	$T_m = 5,288\text{x}10^{-7}$ s	$T_m = 5,311\text{x}10^{-7}$ s	$T_m = 5,326\text{x}10^{-7}$ s
	Lcp	$T_m = 1,558\text{x}10^{-7}$ s	$T_m = 1,554\text{x}10^{-7}$ s	$T_m = 1,448\text{x}10^{-7}$ s
f_{11}	NN_1	$T_m = NA$ $TP_m = 4,928\text{x}10^{-6}$ s	$T_m = NA$ $TP_m = 4,881\text{x}10^{-6}$ s	$T_m = NA$ $TP_m = 4,889\text{x}10^{-6}$ s
	NN_2	$T_m = NA$ $TP_m = 5,277\text{x}10^{-6}$ s	$T_m = NA$ $TP_m = 5,27\text{x}10^{-6}$ s	$T_m = NA$ $TP_m = 5,272\text{x}10^{-6}$ s
	NN_3	$T_m = NA$ $TP_m = 5,756\text{x}10^{-6}$ s	$T_m = NA$ $TP_m = 5,596\text{x}10^{-6}$ s	$T_m = NA$ $TP_m = 5,621\text{x}10^{-6}$ s
	$Paco$	$T_m = 7,233\text{x}10^{-7}$ s	$T_m = 7,206\text{x}10^{-7}$ s	$T_m = 7,221\text{x}10^{-7}$ s
	Lcp	$T_m = 2,242\text{x}10^{-7}$ s	$T_m = 2,258\text{x}10^{-7}$ s	$T_m = 2,259\text{x}10^{-7}$ s
f_{12}	NN_1	$T_m = NA$ $TP_m = 4,847\text{x}10^{-6}$ s	$T_m = NA$ $TP_m = 4,899\text{x}10^{-6}$ s	$T_m = NA$ $TP_m = 4,871\text{x}10^{-6}$ s
	NN_2	$T_m = NA$ $TP_m = 5,219\text{x}10^{-6}$ s	$T_m = NA$ $TP_m = 5,199\text{x}10^{-6}$ s	$T_m = NA$ $TP_m = 5,239\text{x}10^{-6}$ s
	NN_3	$T_m =$ $TP_m = 5,56\text{x}10^{-6}$ s	$T_m =$ $TP_m = 5,586\text{x}10^{-6}$ s	$T_m =$ $TP_m = 5,588\text{x}10^{-6}$ s
	$Paco$	$T_m = 8,484\text{x}10^{-7}$ s	$T_m = 8,528\text{x}10^{-7}$ s	$T_m = 8,522\text{x}10^{-7}$ s
	Lcp	$T_m = 2,386\text{x}10^{-7}$ s	$T_m = 2,396\text{x}10^{-7}$ s	$T_m = 2,394\text{x}10^{-7}$ s
f_{13}	NN_1	$T_m = NA$ $TP_m = 5,057\text{x}10^{-6}$ s	$T_m = NA$ $TP_m = 5,051\text{x}10^{-6}$ s	$T_m = NA$ $TP_m = 5,05\text{x}10^{-6}$ s
	NN_2	$T_m = NA$ $TP_m = 5,401\text{x}10^{-6}$ s	$T_m = NA$ $TP_m = 5,402\text{x}10^{-6}$ s	$T_m = NA$ $TP_m = 5,395\text{x}10^{-6}$ s
	NN_3	$T_m = NA$ $TP_m = 5,715\text{x}10^{-6}$ s	$T_m = NA$ $TP_m = 5,718\text{x}10^{-6}$ s	$T_m = NA$ $TP_m = 5,708\text{x}10^{-6}$ s

Table 3 – continued from previous page

File	Method	40%	50%	60%
	$Paco$	$T_m = 9,069\text{x}10^{-7}$ s	$T_m = 9,091\text{x}10^{-7}$ s	$T_m = 9,082\text{x}10^{-7}$ s
	Lcp	$T_m = 2,566\text{x}10^{-7}$ s	$T_m = 2,567\text{x}10^{-7}$ s	$T_m = 2,567\text{x}10^{-7}$ s

Table 4
Prediction Results for query from 70% to 100%. TP_m is the mean time taken by the NN model to obtain an index with a certain error ϵ and T_m is the mean time to find an integer in the array, in the case of NN we used a binary search method in a vector with maximum length $2\epsilon + 1$

File	Method	70%	80%	90%	100%
f_1	NN_1	$T_m = 1,467\text{x}10^{-3}$ s $TP_m = 7,415\text{x}10^{-6}$ s	$T_m = 1,198\text{x}10^{-3}$ s $TP_m = 5,86\text{x}10^{-6}$ s	$T_m = 1,125\text{x}10^{-3}$ s $TP_m = 5,558\text{x}10^{-6}$ s	$T_m = 1,295\text{x}10^{-3}$ s $TP_m = 5,376\text{x}10^{-6}$ s
	NN_2	$T_m = 1,343\text{x}10^{-3}$ s $TP_m = 5,969\text{x}10^{-6}$ s	$T_m = 1,633\text{x}10^{-3}$ s $TP_m = 6,909\text{x}10^{-6}$ s	$T_m = 1,258\text{x}10^{-3}$ s $TP_m = 5,227\text{x}10^{-6}$ s	$T_m = 1,244\text{x}10^{-3}$ s $TP_m = 6,056\text{x}10^{-6}$ s
	NN_3	$T_m = 1,299\text{x}10^{-3}$ s $TP_m = 7,659\text{x}10^{-6}$ s	$T_m = 1,308\text{x}10^{-3}$ s $TP_m = 6,76\text{x}10^{-6}$ s	$T_m = 1,447\text{x}10^{-3}$ s $TP_m = 5,832\text{x}10^{-6}$ s	$T_m = 1,491\text{x}10^{-3}$ s $TP_m = 6,603\text{x}10^{-6}$ s
	$Paco$	$T_m = 6,928\text{x}10^{-6}$ s	$T_m = 7,291\text{x}10^{-6}$ s	$T_m = 6,675\text{x}10^{-6}$ s	$T_m = 5,493\text{x}10^{-6}$ s
	Lcp	$T_m = 6,133\text{x}10^{-6}$s	$T_m = 5,165\text{x}10^{-6}$s	$T_m = 3,949\text{x}10^{-6}$s	$T_m = 3,327\text{x}10^{-6}$s
f_2	NN_1	$T_m = 1,722\text{x}10^{-3}$ s $TP_m = 6,447\text{x}10^{-6}$ s	$T_m = 1,942\text{x}10^{-3}$ s $TP_m = 5,873\text{x}10^{-6}$ s	$T_m = 2,054\text{x}10^{-3}$ s $TP_m = 5,398\text{x}10^{-6}$ s	$T_m = 2,236\text{x}10^{-3}$ s $TP_m = 4,047\text{x}10^{-6}$ s
	NN_2	$T_m = 1,946\text{x}10^{-3}$ s $TP_m = 7,672\text{x}10^{-6}$ s	$T_m = 2,192\text{x}10^{-3}$ s $TP_m = 6,388\text{x}10^{-6}$ s	$T_m = 1,959\text{x}10^{-3}$ s $TP_m = 5,746\text{x}10^{-6}$ s	$T_m = 2,121\text{x}10^{-3}$ s $TP_m = 5,322\text{x}10^{-6}$ s
	NN_3	$T_m = 2,062\text{x}10^{-3}$ s $TP_m = 7,416\text{x}10^{-6}$ s	$T_m = 2,246\text{x}10^{-3}$ s $TP_m = 7,322\text{x}10^{-6}$ s	$T_m = 2,393\text{x}10^{-3}$ s $TP_m = 6,808\text{x}10^{-6}$ s	$T_m = 2,043\text{x}10^{-3}$ s $TP_m = 6,228\text{x}10^{-6}$ s
	$Paco$	$T_m = 2,495\text{x}10^{-6}$ s	$T_m = 2,074\text{x}10^{-6}$ s	$T_m = 1,171\text{x}10^{-6}$ s	$T_m = 1,486\text{x}10^{-6}$ s
	Lcp	$T_m = 1,64\text{x}10^{-6}$s	$T_m = 1,235\text{x}10^{-6}$s	$T_m = 9,077\text{x}10^{-7}$s	$T_m = 8,751\text{x}10^{-7}$s
f_3	NN_1	$T_m = 2,897\text{x}10^{-3}$ s $TP_m = 5,001\text{x}10^{-6}$ s	$T_m = 3,833\text{x}10^{-3}$ s $TP_m = 6,338\text{x}10^{-6}$ s	$T_m = 4,152\text{x}10^{-3}$ s $TP_m = 5,613\text{x}10^{-6}$ s	$T_m = 3,4\text{x}10^{-3}$ s $TP_m = 4,475\text{x}10^{-6}$ s

Table 4 – continued from previous page

File	Method	70%	80%	90%	100%
	NN_2	$T_m = 3{,}118\text{x}10^{-3}$ s $TP_m = 6{,}298\text{x}10^{-6}$ s	$T_m = 3{,}816\text{x}10^{-3}$ s $TP_m = 5{,}746\text{x}10^{-6}$ s	$T_m = 3{,}65\text{x}10^{-3}$ s $TP_m = 6{,}257\text{x}10^{-6}$ s	$T_m = 3{,}57\text{x}10^{-3}$ s $TP_m = 5{,}35\text{x}10^{-6}$ s
	NN_3	$T_m = 3{,}6\text{x}10^{-3}$ s $TP_m = 6{,}718\text{x}10^{-6}$ s	$T_m = 4{,}087\text{x}10^{-3}$ s $TP_m = 6{,}264\text{x}10^{-6}$ s	$T_m = 4{,}147\text{x}10^{-3}$ s $TP_m = 5{,}698\text{x}10^{-6}$ s	$T_m = 3{,}767\text{x}10^{-3}$ s $TP_m = 5{,}158\text{x}10^{-6}$ s
	$Paco$	$T_m = 9{,}453\text{x}10^{-7}$ s	$T_m = 7{,}004\text{x}10^{-7}$ s	$T_m = 6{,}908\text{x}10^{-7}$ s	$T_m = 1{,}423\text{x}10^{-6}$ s
	Lcp	$T_m = 8{,}583\text{x}10^{-7}$s	$T_m = 7{,}278\text{x}10^{-7}$s	$T_m = 8{,}419\text{x}10^{-7}$s	$T_m = 8{,}227\text{x}10^{-7}$s
f_4	NN_1	$T_m = 5{,}184\text{x}10^{-3}$ s $TP_m = 4{,}754\text{x}10^{-6}$ s	$T_m = 5{,}436\text{x}10^{-3}$ s $TP_m = 4{,}749\text{x}10^{-6}$ s	$T_m = 5{,}846\text{x}10^{-3}$ s $TP_m = 4{,}936\text{x}10^{-6}$ s	$T_m = 6{,}197\text{x}10^{-3}$ s $TP_m = 4{,}823\text{x}10^{-6}$ s
	NN_2	$T_m = 5{,}145\text{x}10^{-3}$ s $TP_m = 6{,}079\text{x}10^{-6}$ s	$T_m = 5{,}628\text{x}10^{-3}$ s $TP_m = 5{,}117\text{x}10^{-6}$ s	$T_m = 6{,}215\text{x}10^{-3}$ s $TP_m = 4{,}837\text{x}10^{-6}$ s	$T_m = 6{,}972\text{x}10^{-3}$ s $TP_m = 5{,}024\text{x}10^{-6}$ s
	NN_3	$T_m = 5{,}72\text{x}10^{-3}$ s $TP_m = 6{,}275\text{x}10^{-6}$ s	$T_m = 6{,}036\text{x}10^{-3}$ s $TP_m = 7{,}618\text{x}10^{-6}$ s	$T_m = 6{,}743\text{x}10^{-3}$ s $TP_m = 6{,}368\text{x}10^{-6}$ s	$T_m = 7{,}179\text{x}10^{-3}$ s $TP_m = 6{,}446\text{x}10^{-6}$ s
	$Paco$	$T_m = 6{,}25\text{x}10^{-7}$ s	$T_m = 6{,}517\text{x}10^{-7}$ s	$T_m = 6{,}242\text{x}10^{-7}$ s	$T_m = 1{,}169\text{x}10^{-6}$ s
	Lcp	$T_m = 3{,}378\text{x}10^{-7}$s	$T_m = 3{,}414\text{x}10^{-7}$s	$T_m = 5{,}746\text{x}10^{-7}$s	$T_m = 4{,}23\text{x}10^{-7}$s
f_5	NN_1	$T_m = 9{,}376\text{x}10^{-3}$ s $TP_m = 4{,}865\text{x}10^{-6}$ s	$T_m = 1{,}045\text{x}10^{-2}$ s $TP_m = 4{,}767\text{x}10^{-6}$ s	$T_m = 1{,}089\text{x}10^{-2}$ s $TP_m = 5{,}542\text{x}10^{-6}$ s	$T_m = 1{,}188\text{x}10^{-2}$ s $TP_m = 5{,}306\text{x}10^{-6}$ s
	NN_2	$T_m = 9{,}912\text{x}10^{-3}$ s $TP_m = 5{,}099\text{x}10^{-6}$ s	$T_m = 9{,}699\text{x}10^{-3}$ s $TP_m = 5{,}553\text{x}10^{-6}$ s	$T_m = 1{,}232\text{x}10^{-2}$ s $TP_m = 5{,}502\text{x}10^{-6}$ s	$T_m = 1{,}228\text{x}10^{-2}$ s $TP_m = 5{,}455\text{x}10^{-6}$ s
	NN_3	$T_m = 9{,}921\text{x}10^{-3}$ s $TP_m = 6{,}539\text{x}10^{-6}$ s	$T_m = 1{,}09\text{x}10^{-2}$ s $TP_m = 5{,}786\text{x}10^{-6}$ s	$T_m = 1{,}333\text{x}10^{-2}$ s $TP_m = 5{,}779\text{x}10^{-6}$ s	$T_m = 1{,}333\text{x}10^{-2}$ s $TP_m = 5{,}258\text{x}10^{-6}$ s
	$Paco$	$T_m = 1{,}128\text{x}10^{-6}$ s	$T_m = 1{,}031\text{x}10^{-6}$ s	$T_m = 9{,}974\text{x}10^{-7}$ s	$T_m = 1{,}256\text{x}10^{-6}$ s
	Lcp	$T_m = 2{,}878\text{x}10^{-7}$s	$T_m = 2{,}899\text{x}10^{-7}$s	$T_m = 2{,}928\text{x}10^{-7}$s	$T_m = 2{,}881\text{x}10^{-7}$s
f_6	NN_1	$T_m = 1{,}619\text{x}10^{-2}$ s $TP_m = 5{,}388\text{x}10^{-6}$ s	$T_m = 1{,}888\text{x}10^{-2}$ s $TP_m = 4{,}865\text{x}10^{-6}$ s	$T_m = 2{,}121\text{x}10^{-2}$ s $TP_m = 5{,}043\text{x}10^{-6}$ s	$T_m = 2{,}143\text{x}10^{-2}$ s $TP_m = 4{,}892\text{x}10^{-6}$ s
	NN_2	$T_m = 1{,}731\text{x}10^{-2}$ s $TP_m = 5{,}169\text{x}10^{-6}$ s	$T_m = 2{,}036\text{x}10^{-2}$ s $TP_m = 5{,}704\text{x}10^{-6}$ s	$T_m = 2{,}083\text{x}10^{-2}$ s $TP_m = 5{,}762\text{x}10^{-6}$ s	$T_m = 2{,}313\text{x}10^{-2}$ s $TP_m = 5{,}892\text{x}10^{-6}$ s

Table 4 – continued from previous page

File	Method	70%	80%	90%	100%
	NN_3	$T_m = 1{,}942\text{x}10^{-2}$ s $TP_m = 6{,}903\text{x}10^{-6}$ s	$T_m = 2{,}414\text{x}10^{-2}$ s $TP_m = 6{,}932\text{x}10^{-6}$ s	$T_m = 2{,}382\text{x}10^{-2}$ s $TP_m = 6{,}79\text{x}10^{-6}$ s	$T_m = 2{,}553\text{x}10^{-2}$ s $TP_m = 6{,}38\text{x}10^{-6}$ s
	$Paco$	$T_m = 5{,}599\text{x}10^{-7}$ s	$T_m = 5{,}569\text{x}10^{-7}$ s	$T_m = 5{,}684\text{x}10^{-7}$ s	$T_m = 5{,}753\text{x}10^{-7}$ s
	Lcp	$T_m = 4{,}002\text{x}10^{-7}$s	$T_m = 4{,}011\text{x}10^{-7}$s	$T_m = 3{,}947\text{x}10^{-7}$s	$T_m = 3{,}457\text{x}10^{-7}$s
f_7	NN_1	$T_m = 3{,}014\text{x}10^{-2}$ s $TP_m = 5{,}336\text{x}10^{-6}$ s	$T_m = 3{,}315\text{x}10^{-2}$ s $TP_m = 5{,}299\text{x}10^{-6}$ s	$T_m = 4{,}011\text{x}10^{-2}$ s $TP_m = 5{,}175\text{x}10^{-6}$ s	$T_m = 4{,}263\text{x}10^{-2}$ s $TP_m = 5{,}252\text{x}10^{-6}$ s
	NN_2	$T_m = 3{,}328\text{x}10^{-2}$ s $TP_m = 5{,}651\text{x}10^{-6}$ s	$T_m = 3{,}594\text{x}10^{-2}$ s $TP_m = 5{,}925\text{x}10^{-6}$ s	$T_m = 4{,}358\text{x}10^{-2}$ s $TP_m = 5{,}718\text{x}10^{-6}$ s	$T_m = 4{,}487\text{x}10^{-2}$ s $TP_m = 5{,}496\text{x}10^{-6}$ s
	NN_3	$T_m = 3{,}592\text{x}10^{-2}$ s $TP_m = 6{,}479\text{x}10^{-6}$ s	$T_m = 3{,}875\text{x}10^{-2}$ s $TP_m = 6{,}339\text{x}10^{-6}$ s	$T_m = 4{,}549\text{x}10^{-2}$ s $TP_m = 6{,}266\text{x}10^{-6}$ s	$T_m = 5{,}011\text{x}10^{-2}$ s $TP_m = 6{,}239\text{x}10^{-6}$ s
	$Paco$	$T_m = 6{,}042\text{x}10^{-7}$ s	$T_m = 6{,}088\text{x}10^{-7}$ s	$T_m = 6{,}054\text{x}10^{-7}$ s	$T_m = 4{,}443\text{x}10^{-7}$ s
	Lcp	$T_m = 3{,}17\text{x}10^{-7}$s	$T_m = 4{,}191\text{x}10^{-7}$s	$T_m = 4{,}098\text{x}10^{-7}$s	$T_m = 4{,}227\text{x}10^{-7}$s
f_8	NN_1	$T_m = 5{,}652\text{x}10^{-2}$ s $TP_m = 5{,}386\text{x}10^{-6}$ s	$T_m = 6{,}431\text{x}10^{-2}$ s $TP_m = 5{,}091\text{x}10^{-6}$ s	$T_m = 7{,}256\text{x}10^{-2}$ s $TP_m = 5{,}351\text{x}10^{-6}$ s	$T_m = 7{,}958\text{x}10^{-2}$ s $TP_m = 5{,}142\text{x}10^{-6}$ s
	NN_2	$T_m = 6{,}236\text{x}10^{-2}$ s $TP_m = 5{,}878\text{x}10^{-6}$ s	$T_m = 7{,}018\text{x}10^{-2}$ s $TP_m = 5{,}811\text{x}10^{-6}$ s	$T_m = 7{,}782\text{x}10^{-2}$ s $TP_m = 5{,}789\text{x}10^{-6}$ s	$T_m = 8{,}494\text{x}10^{-2}$ s $TP_m = 5{,}923\text{x}10^{-6}$ s
	NN_3	$T_m = 6{,}733\text{x}10^{-2}$ s $TP_m = 8{,}002\text{x}10^{-6}$ s	$T_m = 8{,}036\text{x}10^{-2}$ s $TP_m = 6{,}604\text{x}10^{-6}$ s	$T_m = 8{,}956\text{x}10^{-2}$ s $TP_m = 5{,}841\text{x}10^{-6}$ s	$T_m = 8{,}621\text{x}10^{-2}$ s $TP_m = 5{,}987\text{x}10^{-6}$ s
	$Paco$	$T_m = 6{,}212\text{x}10^{-7}$ s	$T_m = 6{,}093\text{x}10^{-7}$ s	$T_m = 4{,}376\text{x}10^{-7}$ s	$T_m = 4{,}941\text{x}10^{-7}$ s
	Lcp	$T_m = 2{,}177\text{x}10^{-7}$ s	$T_m = 2{,}113\text{x}10^{-7}$ s	$T_m = 2{,}101\text{x}10^{-7}$ s	$T_m = 2{,}4\text{x}10^{-7}$ s
f_9	NN_1	$T_m = 4{,}489\text{x}10^{-1}$ s $TP_m = 4{,}916\text{x}10^{-6}$ s	$T_m = 5{,}042\text{x}10^{-1}$ s $TP_m = 5{,}013\text{x}10^{-6}$ s	$T_m = 5{,}662\text{x}10^{-1}$ s $TP_m = 4{,}892\text{x}10^{-6}$ s	$T_m = 6{,}477\text{x}10^{-1}$ s $TP_m = 4{,}77\text{x}10^{-6}$ s
	NN_2	$T_m = 4{,}998\text{x}10^{-1}$ s $TP_m = 5{,}43\text{x}10^{-6}$ s	$T_m = 5{,}521\text{x}10^{-1}$ s $TP_m = 5{,}424\text{x}10^{-6}$ s	$T_m = 6{,}144\text{x}10^{-1}$ s $TP_m = 5{,}319\text{x}10^{-6}$ s	$T_m = 6{,}788\text{x}10^{-1}$ s $TP_m = 5{,}28\text{x}10^{-6}$ s
	NN_3	$T_m = 5{,}021\text{x}10^{-1}$ s $TP_m = 5{,}674\text{x}10^{-6}$ s	$T_m = 5{,}765\text{x}10^{-1}$ s $TP_m = 6{,}188\text{x}10^{-6}$ s	$T_m = 6{,}591\text{x}10^{-1}$ s $TP_m = 5{,}696\text{x}10^{-6}$ s	$T_m = 7{,}152\text{x}10^{-1}$ s $TP_m = 5{,}792\text{x}10^{-6}$ s

Table 4 – continued from previous page

File	Method	70%	80%	90%	100%
	Paco	$T_m = 4{,}328\text{x}10^{-7}$ s	$T_m = 4{,}4\text{x}10^{-7}$ s	$T_m = 4{,}361\text{x}10^{-7}$ s	$T_m = 4{,}653\text{x}10^{-7}$ s
	Lcp	$T_m = 1{,}614\text{x}10^{-7}$ s	$T_m = 2{,}024\text{x}10^{-7}$ s	$T_m = 2{,}156\text{x}10^{-7}$ s	$T_m = 1{,}529\text{x}10^{-7}$ s
f_{10}	NN_1	$T_m = 3{,}481$ s $TP_m = 4{,}894\text{x}10^{-6}$ s	$T_m = 3{,}998$ s $TP_m = 5{,}037\text{x}10^{-6}$ s	$T_m = 4{,}525$ s $TP_m = 4{,}911\text{x}10^{-6}$ s	$T_m = 5{,}051$ s $TP_m = 4{,}914\text{x}10^{-6}$ s
	NN_2	$T_m = 3{,}739$ s $TP_m = 5{,}267\text{x}10^{-6}$ s	$T_m = 4{,}263$ s $TP_m = 5{,}359\text{x}10^{-6}$ s	$T_m = 4{,}957$ s $TP_m = 5{,}479\text{x}10^{-6}$ s	$T_m = 5{,}446$ s $TP_m = 5{,}282\text{x}10^{-6}$ s
	NN_3	$T_m = 4{,}011$ s $TP_m = 5{,}763\text{x}10^{-6}$ s	$T_m = 4{,}584$ s $TP_m = 5{,}723\text{x}10^{-6}$ s	$T_m = 5{,}229$ s $TP_m = 5{,}828\text{x}10^{-6}$ s	$T_m = 5{,}712$ s $TP_m = 5{,}888\text{x}10^{-6}$ s
	Paco	$T_m = 5{,}286\text{x}10^{-7}$ s	$T_m = 5{,}303\text{x}10^{-7}$ s	$T_m = 5{,}399\text{x}10^{-7}$ s	$T_m = 5{,}324\text{x}10^{-7}$ s
	Lcp	$T_m = 1{,}57\text{x}10^{-7}$ s	$T_m = 1{,}453\text{x}10^{-7}$ s	$T_m = 1{,}479\text{x}10^{-7}$ s	$T_m = 1{,}466\text{x}10^{-7}$ s
f_{11}	NN_1	T_m = NA $TP_m = 5{,}041\text{x}10^{-6}$ s	T_m = NA $TP_m = 4{,}976\text{x}10^{-6}$ s	T_m = NA $TP_m = 4{,}789\text{x}10^{-6}$ s	T_m = NA $TP_m = 4{,}895\text{x}10^{-6}$ s
	NN_2	T_m = NA $TP_m = 5{,}237\text{x}10^{-6}$ s	T_m = NA $TP_m = 5{,}289\text{x}10^{-6}$ s	T_m = NA $TP_m = 5{,}334\text{x}10^{-6}$ s	T_m = NA $TP_m = 5{,}237\text{x}10^{-6}$ s
	NN_3	T_m = NA $TP_m = 5{,}596\text{x}10^{-6}$ s	T_m = NA $TP_m = 5{,}504\text{x}10^{-6}$ s	T_m = NA $TP_m = 5{,}666\text{x}10^{-6}$ s	T_m = NA $TP_m = 5{,}612\text{x}10^{-6}$ s
	Paco	$T_m = 7{,}191\text{x}10^{-7}$ s	$T_m = 7{,}234\text{x}10^{-7}$ s	$T_m = 7{,}237\text{x}10^{-7}$ s	$T_m = 7{,}26\text{x}10^{-7}$ s
	Lcp	$T_m = 2{,}299\text{x}10^{-7}$ s	$T_m = 2{,}266\text{x}10^{-7}$ s	$T_m = 2{,}26\text{x}10^{-7}$ s	$T_m = 2{,}289\text{x}10^{-7}$ s
f_{12}	NN_1	T_m = NA $TP_m = 4{,}867\text{x}10^{-6}$ s	T_m = NA $TP_m = 4{,}866\text{x}10^{-6}$ s	T_m = NA $TP_m = 4{,}863\text{x}10^{-6}$ s	T_m = NA $TP_m = 4{,}874\text{x}10^{-6}$ s
	NN_2	T_m = NA $TP_m = 5{,}228\text{x}10^{-6}$ s	T_m = NA $TP_m = 5{,}217\text{x}10^{-6}$ s	T_m = NA $TP_m = 5{,}211\text{x}10^{-6}$ s	T_m = NA $TP_m = 5{,}218\text{x}10^{-6}$ s
	NN_3	T_m = NA $TP_m = 5{,}591\text{x}10^{-6}$ s	T_m = NA $TP_m = 5{,}594\text{x}10^{-6}$ s	T_m = NA $TP_m = 5{,}585\text{x}10^{-6}$ s	T_m = NA $TP_m = 5{,}536\text{x}10^{-6}$ s
	Paco	$T_m = 8{,}527\text{x}10^{-7}$ s	$T_m = 8{,}543\text{x}10^{-7}$ s	$T_m = 8{,}552\text{x}10^{-7}$ s	$T_m = 8{,}571\text{x}10^{-7}$ s
	Lcp	$T_m = 2{,}385\text{x}10^{-7}$ s	$T_m = 2{,}394\text{x}10^{-7}$ s	$T_m = 2{,}39\text{x}10^{-7}$ s	$T_m = 2{,}391\text{x}10^{-7}$ s

Table 4 – continued from previous page

File	Method	70%	80%	90%	100%
f_{13}	NN_1	T_m = NA $TP_m = 5{,}058\text{x}10^{-6}$ s	T_m = NA $TP_m = 5{,}051\text{x}10^{-6}$ s	T_m = NA $TP_m = 5{,}051\text{x}10^{-6}$ s	T_m = NA $TP_m = 5{,}053\text{x}10^{-6}$ s
	NN_2	T_m = NA $TP_m = 5{,}397\text{x}10^{-6}$ s	T_m = NA $TP_m = 5{,}395\text{x}10^{-6}$ s	T_m = NA $TP_m = 5{,}396\text{x}10^{-6}$ s	T_m = NA $TP_m = 5{,}399\text{x}10^{-6}$ s
	NN_3	T_m = NA $TP_m = 5{,}718\text{x}10^{-6}$ s	T_m = NA $TP_m = 5{,}707\text{x}10^{-6}$ s	T_m = NA $TP_m = 5{,}714\text{x}10^{-6}$ s	T_m = NA $TP_m = 5{,}716\text{x}10^{-6}$ s
	Paco	$T_m = 9{,}103\text{x}10^{-7}$ s	$T_m = 9{,}116\text{x}10^{-7}$ s	$T_m = 9{,}124\text{x}10^{-7}$ s	$T_m = 9{,}132\text{x}10^{-7}$ s
	Lcp	$T_m = 2{,}565\text{x}10^{-7}$ s	$T_m = 2{,}579\text{x}10^{-7}$ s	$T_m = 2{,}577\text{x}10^{-7}$ s	$T_m = 2{,}581\text{x}10^{-7}$ s

Table 5
Time to find the search interval ϵ for file10 using a CUDA C++ implementation with global memory. Total time is the sum of the single operation time without the GPU memory loads. Mean time is calculated as the ratio between the total time and the number of elements to be predicted.

	NN1			*NN2*		
	10%	50%	100%	10%	50%	100%
#Elements	104857	524288	1048576	104857	524288	1048576
Mult	$5{,}290\text{x}10^{-4}$s	$2{,}93\text{x}10^{-3}$s	$5{,}93\text{x}10^{-3}$s	$2{,}7045\text{x}10^{-1}$s	1,24866s	2,48187s
Sum	$1{,}733\text{x}10^{-4}$s	$8{,}63\text{x}10^{-4}$s	$1{,}72\text{x}10^{-3}$s	$4{,}9706\text{x}10^{-2}$s	$2{,}4319\text{x}10^{-1}$s	$4{,}8604\text{x}10^{-1}$s
Tanh	$2{,}739\text{x}10^{-4}$s	$1{,}44\text{x}10^{-3}$s	$3{,}90\text{x}10^{-3}$s	$69458\text{x}10^{-2}$s	$3{,}3668\text{x}10^{-1}$s	$6{,}7386\text{x}10^{-1}$s
Relu	$1{,}731\text{x}10^{-4}$s	$8{,}64\text{x}10^{-4}$s	$1{,}72\text{x}10^{-3}$s	$4{,}343\text{x}10^{-1}$s	$2{,}1767\text{x}10^{-1}$s	$4{,}346\text{x}10^{-1}$s
Time Total	$1{,}15\text{x}10^{-3}$s	$6{,}09\text{x}10^{-3}$s	$1{,}33\text{x}10^{-2}$s	$8{,}24\text{x}10^{-1}$s	1,78s	4,08s
Mean Time	$1{,}0963\text{x}10^{-8}$s	$1{,}162\text{x}10^{-8}$s	$1{,}267\text{x}10^{-8}$s	$7{,}858\text{x}10^{-6}$s	$3{,}393\text{x}10^{-6}$s	$3{,}888\text{x}10^{-6}$s

1.2 Problem 2: Prefix Search

bla

Table 6
Time for CPU-GPU copy operations to find the search interval for file10 using a CUDA C++ implementation with global memory. Cuda Copy is the sum of the time of every single input and output operation calling CUDA API cudaMemCopy. Mean Cuda Copy is calculated as the ratio between Cuda Copy and Elements

	*NN*1			*NN*2		
	10%	50%	100%	10%	50%	100%
#Elements	104857	524288	1048576	104857	524288	1048576
Cuda Copy	$2{,}6402\text{x}10^{-2}$s	$2{,}704\text{x}10^{-2}$s	$2{,}7685\text{x}10^{-2}$s	$4{,}7116\text{x}10^{-2}$s	$2{,}3482\text{x}10^{-1}$s	$4{,}6861\text{x}10^{-1}$s
Mean Cuda Copy	$2{,}518\text{x}10^{-7}$s	$5{,}197\text{x}10^{-8}$s	$2{,}64\text{x}10^{-8}$s	$4{,}993\text{x}10^{-7}$s	$4{,}479\text{x}10^{-7}$s	$4{,}469\text{x}10^{-7}$s

Table 7
Total Time to find the search interval ϵ for file10 using a CUDA C++ implementation with global memory. Prediction time is the sum of time spent on operations and copying. Mean Prediction Time is calculated as ration between Prediction Time and Elements

	*NN*1			*NN*2		
	10%	50%	100%	10%	50%	100%
#Elements	104857	524288	1048576	104857	524288	1048576
Prediction Time	$2{,}755\text{x}10^{-2}$s	$3{,}313\text{x}10^{-2}$s	$4{,}097\text{x}10^{-2}$s	$8{,}7110^{-1}$s	2,014s	4,545s
Mean Prediction Time	$2{,}628\text{x}10^{-7}$s	$6{,}319\text{x}10^{-8}$s	$3{,}907\text{x}10^{-8}$s	$8{,}307\text{x}10^{-6}$s	$3{,}841\text{x}10^{-6}$s	$4{,}334\text{x}10^{-6}$s

Neural Nets Solution
bla

Solution by use of data structures
[RG: *Fare riferimento ai lavori di Ferragina*]

Experiments
bla

1.3 Open Problems

Problems from Section 1.1
[RG: *Dalla formalizzazione delle neural netowrks data in Sezione 1.1, si evince*

che un vantaggio semplice da ottenere in termini di spazio, ma anche di tempo è che la codifica del nostro universo possa avere risoluzione inferiore a d bits, semmai sfruttando informazione statistica circa i dati. Ciò porta naturalmente a chiedersi se la nozione di order-preserving coding di GilberT_m and Moore [4] possa beneficiare di quanto sappiamo oggi circa la compressione. Si vedano [5, 6]].

Problems from Section 1.2
bla

References

1. D. Belazzougui, P. Boldi, R. Pagh, S. Vigna, "Monotone minimal perfecT_m hashing: Searching a sorted table with O(1) accesses". In Proceedings of the 20th Annual ACM-SIAM Symposium On Discrete Mathematics (SODA), pp. 785-794, 2009, ACM Press.
2. D. Belazzougui, P. Boldi, R. Pagh, S. Vigna, "Theory and Practice of Monotone Minimal PerfecT_m Hashing.". *Journal of Experimental Algorithmics*, Article N.2025378, 2011, ACM Press.
3. D. Belazzougui, P. Boldi, S. Vigna, "Dynamic Z-FasT_m Tries". In Proceedings of the 17th International Symposium on String Processing and Information Retrieval (SPIRE), pp. 159–172, 2010, Springer.
4. E.N. Gilbert, E.F. Moore," Variable-length binary encoding", Bell Systems Technical Journal 38, pp. 933968, 1959.
5. A. López-Ortiz, M. Mirzazadeh, M.A. Safari, S. Hossein, "FasT_m String Sorting Using Order-preserving Compression", Journal of Experimental Algorithmics, Vol 10, Article N. 1.4, 2006, ACM Press.
6. G. Antoshenkov, D.B. Lomet, J. Murray, "Order preserving string compression", Proceedings of the Twelfth International Conference on Data Engineering, pp. 655-663, 1996.