A Reusable Non-Complementary-DNA-Based Neural Network

Chengjie Sun[a], Xiaoyang Liu[a], Jiafeng Zhong[a], Qin Zhou[a], and Jianjun Cheng[a,b,*]

[a]Research Center for Industries of the Future, Westlake University and School of Engineering, Westlake University, Hangzhou, Zhejiang 310030, China

[b]Institute of Advanced Technology, Westlake Institute for Advanced Study, Hangzhou, Zhejiang 310024, China

*To whom correspondence may be addressed. E-mail: chengjianjun@westlake.edu.cn

Key words: DNA-based neural network, reusable DNA computing, non-complementary computation, pattern recognition materials, lipid-oligonucleotide conjugate

Abstract

The emulation of intelligence across diverse domains of the human brain has spurred the development of neural network based artificial intelligence. The computation of the DNA-based neural network has recently emerged as a focal point of research due to its versatility, scalability, energy efficiency and potentially other huge benefits and implications, as compared to electronic computation. Despite notable advancements, the development of the current DNA neural networks, based on the complementary pairing of DNA nucleobases, is restricted by the lack of reusability of the DNA computing materials, one of key bottlenecks impeding their progression towards neural network learning and evolution. As a result, even the state-of-the-art DNA neural network computations are limited to one-time use currently. Here we report the design of an unprecedented, reusable DNA based non-complementary perceptron (NCP) strategy that implements thresholding and weighted summation functions like neurons and the corresponding neural networks capable of 4-bit molecular pattern recognition. To facilitate the scaling-up of the non-complementary circuits, a modulated concept employing "tagging" domains is also coined. We demonstrate that non-complementary "winner-take-all" circuit can be rationally constructed with a non-complementary annihilator strand. Such NCP based neural network architecture is capable of 4-bit pattern recognition, evidenced by its success in playing the "I Spy" game. Most importantly, when removable input strands (lipid-oligonucleotide conjugates) are utilized, this NCP-based pattern recognition neural network shows high fidelity in multiple-cycle computing. This suggests a reusable DNA based NCP computation strategy as a potential conceptual breakthrough for the design of next-generation DNA computers.

Introduction

The neural network, serving as the foundational structure of human intelligence, has been acknowledged as a potent computational paradigm for attaining artificial intelligence (AI). The past few years have witnessed the explosive advancement of electronic AI programs, evidenced by the emergence of AlphaGo, AlphaFold, ChatGPT, Sora, and many others, that have global and disruptive impact in corresponding fields. All these deep neural networks based models have demonstrated their capability of surpassing human intelligence in specific domains. Albeit great progress has been made in electronic computers, the out-of-memory approach for information processing suffers from high energy consumption, catastrophic forgetting, and low learning efficiency(1). To advance computing architecture designs with lower energy consumption, neuromorphic computing has emerged as an area of intensive investigations(2). The logic behind neuromorphic computing is to realize artificial intelligence by mimicking the structure and function of the human brain neural network. However, the lack of efficient hardware to implement neuron-like behaviors impedes the further development of neuromorphic computing(3).

As the core of the “central dogma of molecular biology”, DNA defines the form, longevity, and evolution of life. The combinations of A/T/C/G four nucleotides in the long DNA chains encode unlimited genetic information, which produces countless organisms on Earth. Generally, DNA message is processed through the perfect "Watson–Crick" pairing (A–T and G–C), which guarantees fidelity of storage, reading, and replication of genetic information. This naturally evolved biological computation principle has been employed to construct artificial, DNA-based computing systems that can mimic human brain functions to some extent(4-8). Since the concept of DNA-based neural network was introduced approximately a decade ago, tremendous progress has been made in the algorithm, advancing the scalability, versatility and precision in molecular information processing(8-10). Benefited from the fast-developing automated synthesis of oligonucleotide(11, 12), DNA computing has gradually emerged as a

powerful and practical strategy for “smart” diagnosis and therapy(13-17).

Although these DNA computers could possess neural network-like functions, various questions remain to be overcome. For instance, once the input strand being used in computing, the DNA “hardware” is non-recoverable as the computing process causes destructive change to the DNA architectures, preventing its reuse for further input information processing. This deficiency severely restricts current DNA-based neural networks from mimicking human brain neural networks that can handle nearly infinite number of input messages (**Figure 1a**). While multiple input information processing is the most basic demand of neural network learning, which reserves the potential of molecular AI, the availability, and in particular reusability, of the DNA computing “hardware” is a major bottleneck that hinders the DNA based AI from rapid advancement(18). Although the DNA program could be regenerated by muting input strands or introduction of the “regenerator” module(19, 20), these systems require complicated sequence design, increased strand length, and accurate control of strand equivalent, which go against the scaling up of the circuits and substantiate the importance of the recovery and reuse of the DNA “hardware”. Until now, no reusable DNA-based neural network has been reported. In fact, the issue of DNA “hardware” reusability has become an outstanding challenge that limits the expedited development of the DNA neural network based biological computing(18).

Recently, a non-complementary computation strategy was reported as an alternative approach for DNA computing by purposely introducing “defects” to the complementary double strands of DNA(21). This strategy outperforms complementary DNA computation for its accelerated computing speed, simplified gating framework, capability of computing with continuously changing variables, while still keeping its excellent biological system compatibility, which makes non-complementary computation a highly attractive computing principle for the design of the next-generation biological computers. Moreover, this computation approach is based on thermodynamic control of information flow, which is based on chemical equilibrium

shifting of the strand-displacement reactions. As the chemical equilibrium of a reversible reaction is governed by the concentrations of the reactants, this non-complementary computation system could recover to the initial state if the input strands are cleared. Therefore, we hypothesize that non-complementary computation reserves great potential in achieving reusable DNA-based neural network computing (**Figure 1b**).

In this work, we demonstrate that the thresholding and weighting functions of neurons could be mimicked by a non-complementary DNA-based perceptron (NCP). Moreover, to scaling up the non-complementary circuit, a "tagging" domain is introduced into the non-complementary strands. Wiring of multiple non-complementary gates generates "winner-take-all" neural network, which further supports a 4-bit pattern-recognition neural network that can play an "I Spy" game with human. Delicately, using lipid-oligonucleotide conjugates (LOCs) as removable input, this DNA-based neural network becomes reusable, potentially offering a universal approach to achieve reusable DNA computation, which can be further applied in molecular neural network learning. With aforementioned advantages, we believe that our design may substantially accelerate the development of DNA-computing based molecular artificial intelligence.

Results

Although how neural network generates human intelligence is not totally understood, it is widely accepted that neural network processes information through an in-memory computing strategy(22). The core thought of neuromorphic computing is mimicking the structure and function of neurons, the basic building blocks of neural network. Neurons receive information at their dendrites, transmit the information through their axons, and release the information at their axon termini (**Figure S1**). Only when the weighted sum of the input information exceeds the neuron's threshold, the neuron will be fired to generate an output. It has been reported that this linear threshold gating function could be mimicked by DNA circuit utilizing the "seesaw" gate motif(9). In this work, we

proposed that a similar information-processing task could also be performed by a non-complementary DNA-based perceptron (NCP) (**Figure 2a**). This perceptron contains two parts: a non-complementary duplex for weighting function and a single strand for thresholding function. The non-complementary duplex is formed by a weighting strand for binding with input strand and an output strand to transmit the information of the input strand. The relative concentration of the duplex controls the concentration of the output strand, which finally determines the weighting of the input strand. The thresholding strand is capable of capturing the input strand before the input strand interacting with the duplex. To avoid the problem that the thresholding strand may compete with the weighting strand to bind with the output strand, a thresholding strand with the same sequence as the weighting strand is employed. Given that the input strand will preferentially hybridize with the single strand, only when the concentration of the input strand is larger than that of the thresholding strand, it can interact with the duplex to release the output strand from the perceptron.

To illustrate this concept, an NCP with three mismatches was constructed (NCP_1, **Figure 2b** and **Table S1**). For simplicity, here we directly adopted the same strand sequences of the non-complementary "YES" gate reported in the previous work(21). The output information was read through fluorescence resonance energy transfer (FRET). Both the weighting strand and the thresholding strand were labeled with BHQ2 at their 3' end, while the output strand was labeled with Cy3 at its 5' end. With this setting, the Cy3 fluorescence would be turned on when the output strand is excreted out from the perceptron. We measured the fluorescence output signals with different concentrations of the input strand (**Figure 2c**). When there is a lack of the thresholding strand, the addition of the input strand will directly turn on the Cy3 fluorescence, indicating the release of the output strand. When the thresholding strand is employed (0.7 μM), the concentration of the input strand should be large enough (> 0.7 μM) to initiate the release of the output strand. The observed profiles of fluorescence intensity are highly consistent with those predicted by NUPACK(23) (**Figure S2**), indicating that NCP is highly designable.

Another key function of neurons is the weighted summation of input value(9). In NCP, the weighted summation function could be constructed with different weighting strands associated with different input strands. For example, to construct an NCP capable of processing two inputs (I_1 and I_2), we can introduce two weighting strands, forming two sets of non-complementary duplexes (*i.e.* W_1:O and W_2:O in NCP_2, **Figure 2d**). The weighting value is dependent on the relative concentration of the duplex with respect to the two duplexes. To illustrate this concept, we studied the computing performance of NCP_2 under different weighting parameters for I_1 and I_2 (**Table S2**). As anticipated by NUPACK, the normalized concentrations of the released output strand should approximate the weighted sum of the values of I_1 and I_2 (**Figure S3**). We then experimentally tested the weighted summation performance of NCP_2, where the output values were read through the fluorescence labeled on the output strand (**Table S2**). As expected, the contribution of an input to the output value depends on the corresponding weighting value, and the output signal is the sum of the contributions from different inputs (**Figure 2e**). Since the output is based on the chemical equilibrium shift of the non-complementary strand-displacement reaction, the output value does not linearly depend on the weighting values. Nevertheless, these results indicate that NCP can selectively respond to the input strand that is "remembered" through the relatively high weighting value, demonstrating the capability of NCP in mimicking the function of neuron.

However, in non-complementary hybridization, a strand with a specific sequence can hybridize with many other strands having different sequences at a similar affinity(21). As a result, scaling up non-complementary computation significantly amplifies the challenge of sequence design to prevent non-specific hybridization. To address this problem, a tagging domain is introduced to simplify the sequence design (**Figure 3**). This tagging domain could complementarily hybridize with the corresponding tagging domains of the upstream or downstream associating strands. With the help of the tagging domains, one non-complementary hybridization pattern could be used for the

construction of different NCPs (**Figure 3a**). This design is supported by NUPACK simulation. For example, four strands with the same 15-nt non-complementary hybridization pattern (W:O) and two pairs of 3-nt tagging domain (T_{W3}:T_{O3} and T_{W4}:T_{O4}) could self-sort into two duplexes (WT_{W3}:OT_{O3} and WT_{W4}:OT_{O4}), selectively producing the desired NCP_3 and NCP_4, respectively (**Figure 3b**). Furthermore, by carefully selecting the sequences of tagging domains, several NCPs could be easily constructed with one non-complementary hybridization pattern (**Figures 3c** and **3d**). Like the original version (**Figure 2d**), the weighted summation function for different input strands could also be constructed with corresponding weighting strands (NCP_5, **Figure 4a** and **Table S3**). As the introduction of the tagging domain enhances the binding affinity of the strands, the tagging domains should also be added to the upstream and downstream strands to balance the hybridization energy change, ensuring the equilibrium shift of non-complementary strand-displacement reactions. For the simplicity of description, the tagging domains designated to bind with the upstream or downstream strands are denoted as T_{Xiu} and T_{Xid}, respectively, where "X" represents the function of the strand and "i" represents the identity number of the NCPs (i = 1, 2, …, n). To illustrate the input pattern processing function of NCP with tagging domains, we simulated the output performance of NCP_5 with varied weighting parameters (**Figure 4b**). As anticipated, the normalized concentrations of the released output strand approximate the weighted sum of the input values, illustrating the capability of the NCP in processing 2-bit input information. To further explore the capability of NCP in processing more complicated input pattern, we then studied the output performance of NCP_5 with four kinds of weighting strands (W_{15}, W_{25}, W_{35}, and W_{45}) in processing 4-bit input information (T_dI_1, T_dI_2, T_dI_3, and T_dI_4) through NUPACK simulation (**Figure 4c and Table S5**). The memories of the different input patterns of the NCPs are generated by setting high concentration values of the corresponding weighting strands and low concentration values of other weighting strands. For example, the concentrations of the W_{15}, W_{25}, W_{35} and W_{45} strands are set at 0.9, 0.9, 0.1 and 0.1 μM, respectively, for the writing of the "T_dI_1 and T_dI_2" memory. Meanwhile, the concentration of the output strand (O_5) is fixed at 2 μM. With this setting, the addition

of T_dI_1 or T_dI_2 strands will lead to a greater amount of the O_5 strand being released as compared to that of T_dI_3 or T_dI_4, resulting in higher weighting values of T_dI_1 and T_dI_2 than those of T_dI_3 and T_dI_4. Similarly, other NCP memories can be readily edited by changing the relative concentrations of the weighting strands (**Figure 4c**). Moreover, another NCP that can also process the same set of input information can be easily constructed with another pair of tagging domains (T_{W6}:T_{O6}) (NCP_6, **Figure S4**), demonstrating the versatility of this "tagging" strategy in scaling up of NCP construction.

Furthermore, based on this tagging strategy, non-complementary "winner-take-all" neural network could be constructed. "Winner-take-all" neural network is an important computing principle in decision making, machine learning, and cognitive modeling. In the classic work of Qian *et al.*, this "winner-take-all" neural network is realized based on an "annihilator" gate(8). The fundamental role of an "annihilator" gate is to concurrently hybridize with two distinct input strands to be compared when both input strands are presented in the solution. Inspired by their work, we proposed that similar annihilating function could also be realized based on non-complementary hybridization with a hairpin structure (A_5A_6, **Figure S5**). When two strands (O_5 and O_6) are simultaneously presented, they will concurrently hybridize with A_5A_6. Concretely, A_5A_6 is composed of two domains ($T_{A5u}A$ and $T_{A6u}A$), which are designed for the hybridization with two distinct strands (O_5 and O_6), respectively (**Figure 5a**). This procedure could efficiently amplify the concentration difference between the two strands through a subtraction way (**Figure 5a**). For example, the direct comparison between the normalized concentrations of O_5 and O_6 shows a large area with subtle values (**Figure 5b**, top), which can result in a large decision margin. In contrast, when half of the O_5 and O_6 strands are annihilated, the fuzzy area is notably decreased, resulting in substantially increased concentration contrast (**Figure 5b**, bottom). The reporting gate for O_5 or O_6 is a non-complementary duplex consisting of a receipting strand (A_i, i = 5 or 6) and a reporting strand (R_i, i = 5 or 6) (**Figure 5a**). The structure of A_i is similar to that of the corresponding domain of A_5A_6. An additional 3-nt tag

domain (T_{Ai}, i = 5 or 6) is appended to its 3' end to establish the connection with the corresponding reporting strand (R_i, i = 5 or 6). Accordingly, the structure of R_i strand is composed by two parts, one is a 15-nt domain non-complementarily ("R" domain) hybridizes with the A domain of A_i strand, another is a 3-nt domain (T_{Ri}, i = 5 or 6) complementarily hybridizing with the T_{Ai} domain of A_i strand. By labeling BHQ2 quencher at A_i's 5' end and Cy fluorophore at R_i's 3' end (Cy3 at R_5 and Cy5 at R_6), the release of the two reporting strands could be simultaneously detected through FRET. As a proof-of-concept, a non-complementary "winner-take-all" system capable of processing the output strand information of NCP_5 and NCP_6 was designed, as shown in **Figure 5c** and **Table S7**. The secondary structure of A_5A_6 strand could be predicted by NUPACK. Such result indicates that A_5A_6 strand could stably form a hairpin-like structure through 9 G-C base pairing under our experimental conditions (1 M NaCl, 25 ^oC) (**Figure 5d**). To experimentally verify this design of non-complementary "winner-take-all" neural network, we prepared and implemented this program with different concentration patterns of O_5 and O_6 (**Figure 5e**). As expected, when the concentration of O_5 is larger than that of O_6 (the pink cells), the normalized signal of R_5 (Cy3 fluorescence) is higher than that of R_6 (Cy5 fluorescence). Conversely, the normalized signal of R_6 is higher than that of R_5 (the blue cells). The decision margin is $0.95 < R_5/R_6 < 1.1$ (the white cells on the diagonal). These results are in consistent with those predicted by NUPACK (**Figure S6**), further demonstrating that non-complementary neural network is highly predictable. It should be noted that if A_5A_6 strand is not employed, the "winner-take-all" computing function would have system errors (**Figure S7**). Therefore, the "annihilator" strand is necessary to construct the "winner-take-all" neural network based on non-complementary strands.

A fundamental function of neural networks is pattern recognition(8-10). With the abovementioned framework, a 4-bit pattern-recognition non-complementary neural network can be constructed by wiring NCP_5 and NCP_6 with the "winner-take-all" neural network (**Figures 6a** and **S8**). In this neural network, the input information is firstly parallelly processed by the NCPs to yield weighted summations. Then the weighted

summations are subject to pairwise annihilation to amply the difference between the summations. Finally, the annihilated result is reported by the reporting gates. As illustrated above, different molecular pattern could be remembered by different NCPs through the regulation of the corresponding weighting strands (**Figures 4** and **S4**). We then encoded the input pattern "T_dI_1 and T_dI_2" into Pattern 1 and "T_dI_1 and T_dI_4" into Pattern 2. Through the assignment of the meanings to the input strands, an "I Spy" game can be played between human and the non-complementary neural network (**Figure 6a**). For instance, let T_dI_1, T_dI_2, T_dI_3, and T_dI_4 represent "Spherical", "Red", "Corticate", and "Yellow", respectively. Correspondingly, the recorded Pattern 1 and 2 mean "Apple" and "Lemon", respectively. The neural network could guess the object we refer to by offering it some "clues". For example, when the input pattern is "T_dI_1 and T_dI_2", which means "Spherical" and "Red", the computing solution can give a higher R_5 signal than R_6, telling the answer "Apple" (**Figure S9**). We then experimentally tested this "I Spy" game (**Figure 6b** and **Table S8**). The normalized fluorescence results showed a great correlation of R_5 and R_6 with the corresponding input Pattern 1 and 2, substantiating a similar pattern-recognition function to the previously reported complementary DNA-based neural network(8). Taken together, these results demonstrate the successful realization of 4-bit molecular pattern-recognition based on non-complementary computation.

As mentioned above, in biocomputing, neurons can repetitively and reversibly switch between the resting state and the firing state (**Figure S1**). An ideal DNA-based perceptron should also be reusable to accommodate multiple batches of input, which is technically challenging based on the current state-of-the-art design of complementary DNA computation. For example, in the classic works by Qian *et al.*(8, 9), fuel strands are employed to drive the strand-displacement reactions. When resetting the computing system to the initial state, these fuel strands could compete with initial gating strands to form inactive duplexes, excluding the processing of another batch of input (**Figure S10a**). On the contrary, non-complementary computation uses thermodynamic control of strand-displacement reactions. Therefore, once the input strand is removed from the

reaction, the system would return to its initial state (**Figure S10b**). Thus, the removal of the input strands would facilitate the recovery and the reuse of the non-complementary DNA-based neural networks. There are two strategies to clear input strands from the computing system. One strategy is muting the input strands with their complementary strands(19, 24, 25). This strategy requires precise control of equivalent, otherwise the error in equivalent would be enhanced during cycling. Another strategy is to remove the input strands from the computing system(26-28). This strategy is more like the way employed by neurons, which has been employed in some previous studies to achieve reusable DNA circuits(26-28). In fact, as early as in 1998, a "sticker-based model" has been proposed to reuse DNA computer by removing input strands(29). However, these works need special devices that are not easily available(26, 28, 29) or based on special responsive sequences,(27) preventing these strategies from wide spread development and applications.

We reason that if the input strand is largely different from the DNA hardware in polarity, we can remove the input strand through reversed phase chromatography. Lipid conjugation is a widely used modification method for oligonucleotides, which could dramatically increase the hydrophobicity of oligonucleotides(30). Therefore, we proposed an input-computing-chromatography-recovery process to achieve the reusing of non-complementary-DNA-based computation system with lipid-oligonucleotide conjugates (LOCs) as the removable input strands (**Figure 7a**). To efficiently synthesize a series of LOCs with different sequences, we employed the recently-developed P(V) chemistry to achieve automated synthesis, eliminating the need for preparation of activated lipid phosphorous intermediates(12) (**Figures S14-S32**). To check whether the introduction of lipid will influence the validity of logic judgement or not, we implemented non-complementary "YES", "NOT", "AND", and "OR" gates with LOCs as the input (**Table S9**). All of the gates functioned correctly, indicating that the lipid modification does not alter the gating performance of DNA strands (**Figure S11**). After computing, the input strand should be de-hybridized from the Input:Weighting duplexes to make the DNA hardware available for separation. With the help of NUPACK

simulation, we realized that the hybridization of non-complementary DNA strands is highly Na^+-dependent (**Figure S12**). The hybridization extent of a non-complementary DNA gate significantly decreases with the decrease in Na^+ concentration. Therefore, the de-hybridization and separation of LOC inputs from the DNA hardware could be achieved through a standard Sep-Pak® chromatography (please refer to the Supplementary Information for further details). To demonstrate this method, the chromatography process for removing the LOC input of an NCP was monitored employing a Cy5-labeled LOC input and a Cy3-labeled output strand through the fluorescence signals of the collected fractions (**Figure 7b**). As analyzed from the fluorescence signals, the recovery efficiency of the DNA hardware increases with ACN concentration. However, the leaking of LOCs would also increase with ACN concentration (**Figure 7c**). Therefore, 20% ACN, which could achieve ~80% recovery of the DNA hardware and negligible leaking of LOC input, was chosen as the eluent for recycling of the computing system. It is worth noting that although not all DNA hardware are recovered, the computing performance would not be compromised, since the non-complementary DNA gating is dependent on the relative concentration of the DNA hardware strands. Theoretically, with 80% recovery, the gating performance could be well maintained for at least 20 cycles (**Figure 7d**). Experimentally, we conducted three cycles of NCP firing to preliminarily demonstrate the feasibility of the recycling (**Figure 7e**). Inspired by these results, we further implemented the reusing of the non-complementary 4-bit pattern-recognition neural network (**Figure 6**) with lipid-modified T_dI_1, T_dI_2, and T_dI_4 as the removable input strands (**Figure 7f** and **Figure S13**). This process could be likened to a repeated "I Spy" game between human and the non-complementary neural network. We tested two different playing scenarios: same clues in different cycles and different clues in different cycles. In the first scenario, with the molecular clues "Spherical" and "Red", the neural network consistently provided the correct answer "Apple" for at least three cycles ($R_5/R_6 > 1.1$, **Figure S13**). Similarly, in the second scenario, the computing solution consistently provided the correct answer across different batches of clues (Cycle 1, Cycle 3, Cycle 4, and Cycle 5), and correctly recognized the unclear clue "Spherical" (Cycle 2) ($0.95 < R_5/R_6 < 1.1$, **Figure 7f**). These

results demonstrate the high fidelity of this reusing approach for DNA-based neural networks. To the best of our knowledge, this is the first time that reusable DNA-based neural network is achieved.

Discussion

In the pursuit of artificial neural networks to mimic, or even surpass human intelligence, numerous approaches have been proposed (1, 2, 18, 31-34). As of now, only electronic-based computation has achieved widespread success. However, although the success, the electronic computations are not energy efficient and has significantly exacerbated the energy crisis. In contrast, biological neural networks can process information at similar efficiency with energy consumption orders of magnitude less than that of the electronic computers(1). To this end, neuromorphic computing is emerging as an appealing alternative approach for information processing. However, efficient hardware to construct neuromorphic computers at a macroscopic level is still substantially lacking, primarily due to difficulties in interconnectivity between chips(1). Instead, at the molecular level, interconnectivity between computing units becomes less challenging. Particularly in DNA-based computation, multiple interconnections can be easily achieved through sequence design. Although significant progress has been made in algorithms(8-10), DNA-based neural network computing systems are still limited to one-time use, significantly impeding the further development of neural network learning(18).

From the perspective of reusability, non-complementary DNA-based computation becomes more favorable than traditional complementary DNA-based computation due to its thermodynamic-controlled nature(21) (**Figure S10**). This nature ensures computing fidelity after recycling, even with a relatively noticeable loss of DNA hardware (**Figure 7d**). Our work elucidates, for the first time, that non-complementary computation can process information in a microscopic neuromorphic manner. This microscopic neuromorphic manner is proved by the 4-bit pattern-recognition and the

high-fidelity multi-cycle computing capability of the non-complementary neural networks (**Figures 6** and **7**). Although a huge challenge in sequence design for the construction of more complicated non-complementary neural networks still exists, our results strongly support this expectation: reusable DNA-based neural network could be achieved through non-complementary computation with removable input strands. Taken together, these results demonstrate a novel neuromorphic computing system that is attractive for achieving practical artificial intelligence at the molecular level.

Conclusions

In this work, we hypothesized that reusable DNA-based neural network could be realized based on non-complementary computation. To achieve this goal, we introduced the concept of non-complementary DNA-based perceptron (NCP), which could be used to build non-complementary DNA-based neural network. To simplify the sequence design and scale up the non-complementary neural network, a tagging strategy is further introduced. Based on these frameworks, non-complementary "winner-take-all" and 4-bit pattern-recognition neural networks were successfully constructed. Most importantly, we established a recycling process for non-complementary DNA computing systems using lipid-oligonucleotide conjugates (LOCs) as the removal input strands and Sep-Pak® chromatography. These studies demonstrate that reusing of DNA-based neural network like biological neural network is possible. We expect that such reusable DNA-based neural networks may be further applied in neural network learning, which can pave the way for more powerful and intelligent DNA computers.

Data, Materials, and Software Availability

All study data are included in the article and/or supporting information.

Acknowledgements

This work was supported by the National Natural Science Foundation of China (52233015 and 52203187), Tencent New Cornerstone Investigator Program, and Westlake Education Foundation.

Author contributions

C.S. and J.C. designed research; C.S., X.L., and J.Z. performed research; C.S., X.L., Z.J., Q.Z., and J.C. analyzed data; and C.S., Q.Z., and J.C. wrote the paper.

Competing interests

C.S., X.L., J.Z., Q.Z. and J.C. are inventors on a China patent application 2024107708382 that covers reusable non-complementary-DNA-based neural networks.

References

1. D. Marković, A. Mizrahi, D. Querlioz, J. Grollier, Physics for neuromorphic computing. *Nat. Rev. Phys.* **2**, 499-510 (2020).
2. P. A. Merolla *et al.*, Artificial brains. A million spiking-neuron integrated circuit with a scalable communication network and interface. *Science* **345**, 668-673 (2014).
3. S. Kumar, R. S. Williams, Z. Wang, Third-order nanocircuit elements for neuromorphic engineering. *Nature* **585**, 518-523 (2020).
4. L. M. Adleman, Molecular computation of solutions to combinatorial problems. *Science* **266**, 1021-1024 (1994).
5. B. Yurke, A. J. Turberfield, A. P. Mills, Jr., F. C. Simmel, J. L. Neumann, A DNA-fuelled molecular machine made of DNA. *Nature* **406**, 605-608 (2000).
6. L. Qian, E. Winfree, Scaling up digital circuit computation with DNA strand displacement cascades. *Science* **332**, 1196-1201 (2011).
7. S. M. Douglas, I. Bachelet, G. M. Church, A logic-gated nanorobot for targeted transport of molecular payloads. *Science* **335**, 831-834 (2012).
8. K. M. Cherry, L. Qian, Scaling up molecular pattern recognition with DNA-based winner-take-all neural networks. *Nature* **559**, 370-376 (2018).
9. L. Qian, E. Winfree, J. Bruck, Neural network computation with DNA strand displacement cascades. *Nature* **475**, 368-372 (2011).
10. S. Okumura *et al.*, Nonlinear decision-making with enzymatic neural networks. *Nature* **610**, 496-501 (2022).
11. M. Madsen, K. V. Gothelf, Chemistries for DNA nanotechnology. *Chem. Rev.* **119**, 6384-6458 (2019).
12. Y. Huang *et al.*, A P(V) platform for oligonucleotide synthesis. *Science* **373**, 1265-1270 (2021).
13. Q. Ma *et al.*, An automated DNA computing platform for rapid etiological diagnostics. *Sci. Adv.* **8**, eade0453 (2022).
14. Y. Zhang *et al.*, Activating a DNA nanomachine via computation across cancer cell membranes for precise therapy of solid tumors. *J. Am. Chem. Soc.* **143**,

15233-15242 (2021).

15. Y. Wu, L. Zhang, S. Berretti, S. Wan, Medical image encryption by content-aware DNA computing for secure healthcare. *IEEE Trans. Ind. Inform.* **19**, 2089-2098 (2023).
16. F. F. Yin *et al.*, DNA-framework-based multidimensional molecular classifiers for cancer diagnosis. *Nat. Nanotechnol.* **18**, 677-686 (2023).
17. A. Ishaqat, X. Zhang, Q. Liu, L. Zheng, A. Herrmann, Programming DNA circuits for controlled immunostimulation through CpG oligodeoxynucleotide delivery. *J. Am. Chem. Soc.* **145**, 12465-12474 (2023).
18. S. Yang *et al.*, DNA as a universal chemical substrate for computing and data storage. *Nat. Rev. Chem.* **8**, 179-194 (2024).
19. A. J. Genot, J. Bath, A. J. Turberfield, Reversible logic circuits made of DNA. *J. Am. Chem. Soc.* **133**, 20080-20083 (2011).
20. N. V. DelRosso, S. Hews, L. Spector, N. D. Derr, A molecular circuit regenerator to implement iterative strand displacement operations. *Angew. Chem. Int. Ed.* **56**, 4443-4446 (2017).
21. M. P. Nikitin, Non-complementary strand commutation as a fundamental alternative for information processing by DNA and gene regulation. *Nat. Chem.* **15**, 70-82 (2023).
22. X. Huang *et al.*, An ultrafast bipolar flash memory for self-activated in-memory computing. *Nat. Nanotechnol.* **18**, 486-492 (2023).
23. J. N. Zadeh *et al.*, NUPACK: Analysis and design of nucleic acid systems. *J. Comput. Chem.* **32**, 170-173 (2011).
24. S. Garg *et al.*, Renewable time-responsive DNA circuits. *Small* 10.1002/smll.201801470, e1801470 (2018).
25. J. Hahn, W. M. Shih, Thermal cycling of DNA devices via associative strand displacement. *Nucleic Acids Res.* **47**, 10968-10975 (2019).
26. T. E. Tomov *et al.*, DNA bipedal motor achieves a large number of steps due to operation using microfluidics-based interface. *ACS Nano* **11**, 4002-4008 (2017).
27. Y. Guo *et al.*, pH-Controlled detachable DNA circuitry and its application in

resettable self-assembly of spherical nucleic acids. *ACS Nano* **14**, 8317-8327 (2020).

28. Y. Pei *et al.*, Single-molecule resettable DNA computing via magnetic tweezers. *Nano Lett.* **22**, 3003-3010 (2022).
29. S. A. M. Roweis *et al.*, A sticker-based model for DNA computation. *J. Comput. Biol.* **5**, 615-629 (1998).
30. X. Li *et al.*, Lipid-oligonucleotide conjugates for bioapplications. *Natl. Sci. Rev.* **7**, 1933-1953 (2020).
31. H. Jaeger, Towards a generalized theory comprising digital, neuromorphic and unconventional computing. *Neuromorph. Comput. Eng.* **1**, 012002 (2021).
32. D. V. Nicolau *et al.*, Parallel computation with molecular-motor-propelled agents in nanofabricated networks. *Proc. Natl. Acad. Sci. U. S. A.* **113**, 2591-2596 (2016).
33. H. Yasuda *et al.*, Mechanical computing. *Nature* **598**, 39-48 (2021).
34. Y. Kim *et al.*, Evidence for the utility of quantum computing before fault tolerance. *Nature* **618**, 500-505 (2023).

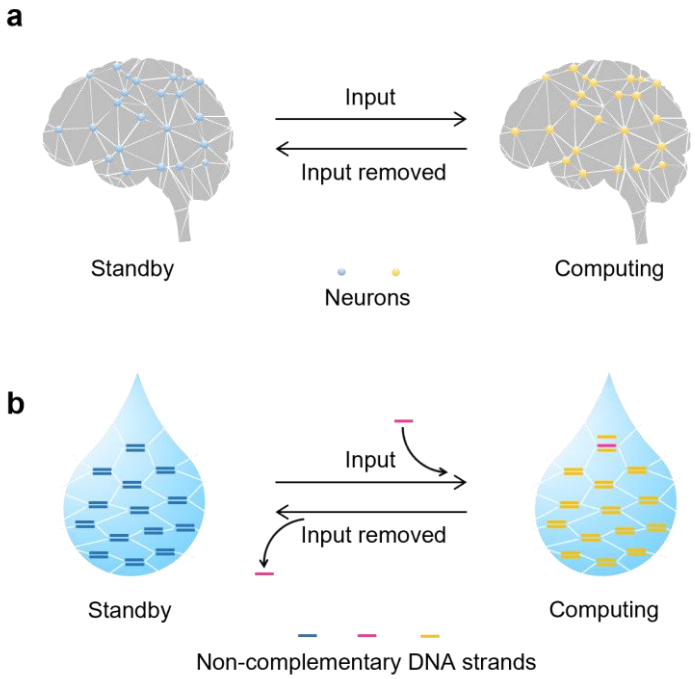

Figure 1. Biological neural network and its non-complementary-DNA-based computation mimic. (a) Biological neural network could be reused once the input signal is removed. **(b)** Non-complementary-DNA-based artificial neural network could be reused once the input strand is removed.

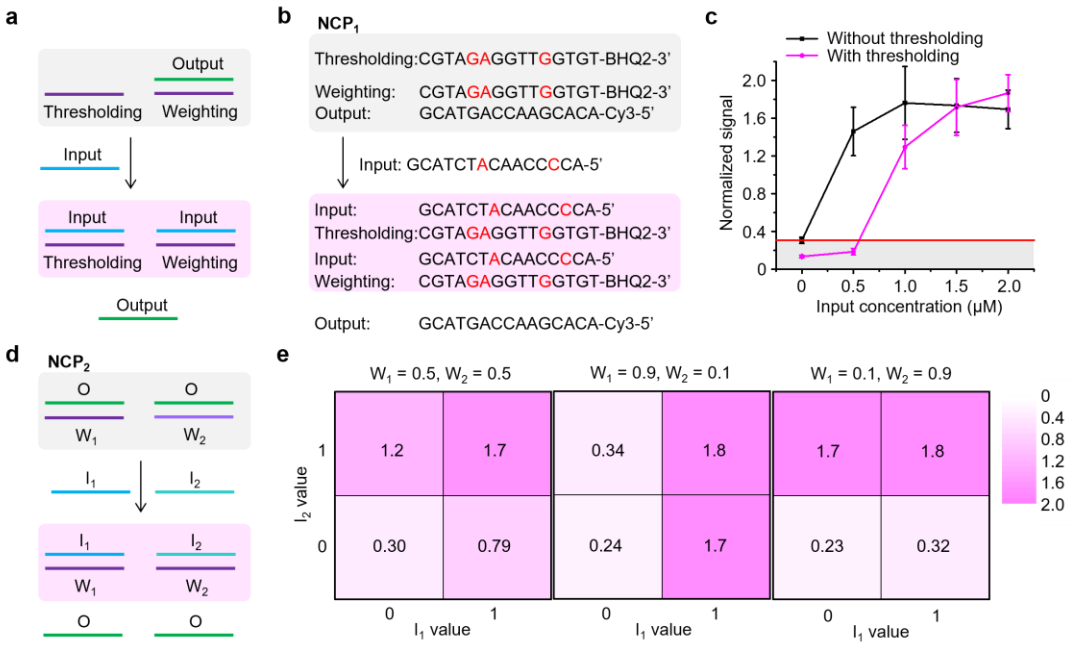

Figure 2. Construction of the non-complementary-DNA-based perceptron (NCP). **(a)** Schematic illustration of the input-output process of an NCP. **(b)** Strand sequences of a typical NCP. The double strand with 3 mismatches could interact with the input strand to produce a double strand with 2 mismatches and release the output strand, which procedure would turn on the fluorescence labeled on the output strand. **(c)** Thresholding function of the NCP. **(d)** Schematic illustration of an NCP executing the weighted summation function for two input strands (I_1 and I_2). **(e)** Weighted summation performance of NCP_2 with different weighting values for I_1 and I_2.

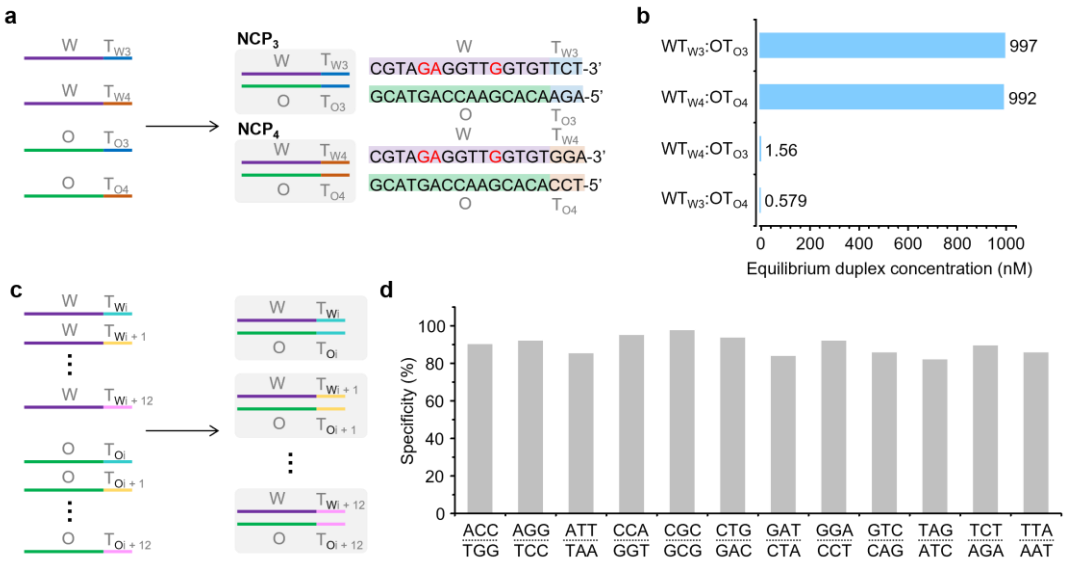

Figure 3. Scaling up the construction of NCP with the tagging strategy. (a) Schematic illustration and the sequences of two NCPs (NCP_3 and NCP_4) formed through self-sorting of four strands with two pairs of tagging domains (T_{W3}:T_{O3} and T_{W4}:T_{O4}, respectively) and one non-complementary sequence pattern (W:O). **(b)** Simulated equilibrium concentration of the desired NCP duplexes (WT_{W3}:OT_{O3} and WT_{W4}:OT_{O4}) and the undesired duplexes (WT_{W4}:OT_{O3} and WT_{W3}:OT_{O4}), which indicates the high formation specificity of NCP_3 and NCP_4. **(c)** Schematic illustration of 12 NCPs' formation with 12 pairs of tagging domains and one non-complementary sequence pattern. **(d)** 12 NCPs can form through 12 pairs of tagging domains and one non-complementary sequence pattern at high specificity.

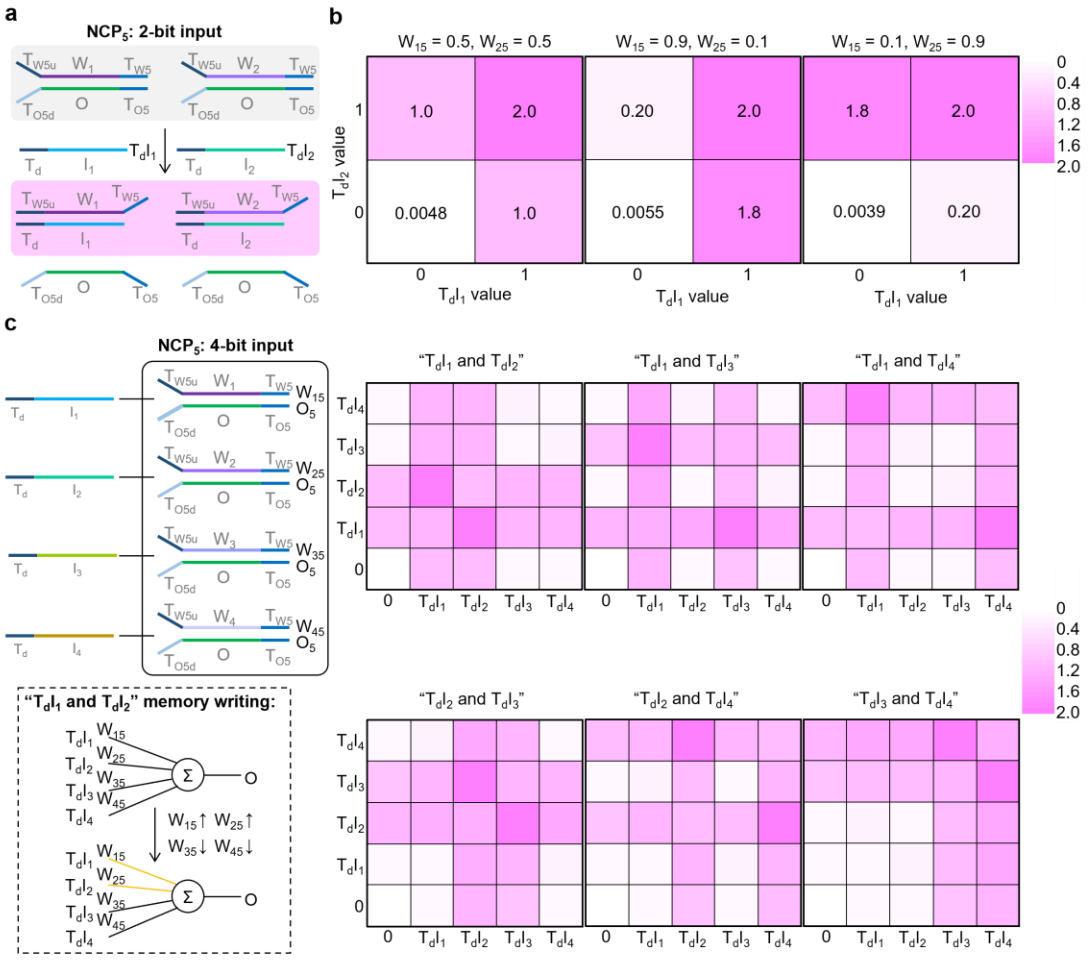

Figure 4. Construction of an NCP utilizing the tagging strategy for weighted summation of 2-bit or 4-bit input information. (a) Schematic illustration of an NCP with tagging domains for weighted summation of 2-bit input information (T_dI_1 and T_dI_2). **(b)** Simulated weighted summation performance of the NCP with different weighting values for T_dI_1 and T_dI_2. **(c)** Schematic illustration and simulated weighted summation performance of an NCP with tagging domains processing 4-bit input information (T_dI_1, T_dI_2, T_dI_3, and T_dI_4). The weighting values are set by regulating the concentrations of the corresponding weighting strands. Data are predicted by NUPACK.

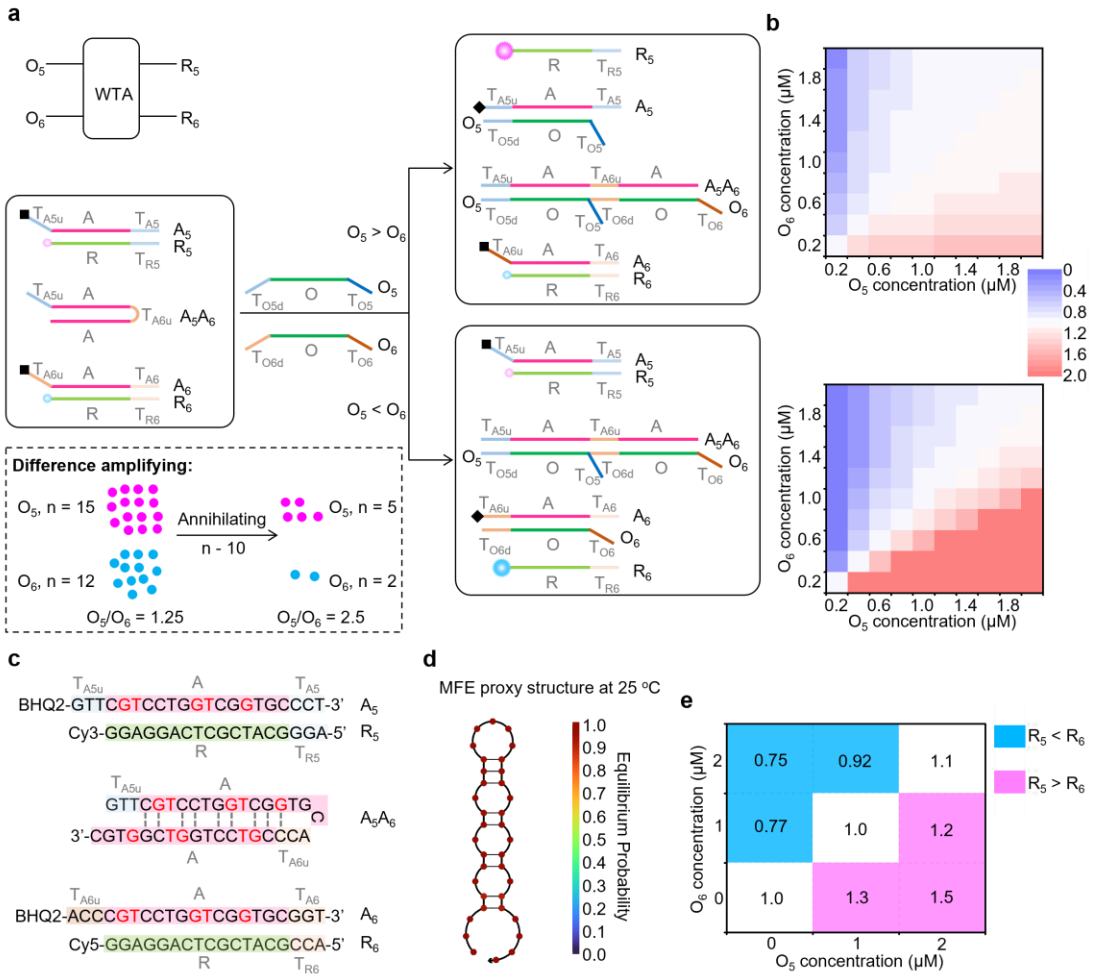

Figure 5. Non-complementary DNA computation-based winner-take-all neural network. (a) Schematic illustration of the strand-displacement reactions in non-complementary-DNA-based winner-take-all neural network. With the help of the annihilator strand (A_5A_6), the concentration difference between two strands (O_5 and O_6) could be amplified through a subtraction way. **(b)** Normalized O_5/O_6 concentration ratio with no (top) or 50% (bottom) strand annihilating operation. **(c)** Sequences of a non-complementary-DNA-based winner-take-all neural network processing the O_5 and O_6 strands. **(d)** Simulated secondary structure of A_5A_6 strand. **(e)** Two-species winner-take-all behavior of the non-complementary DNA neural network in **(c)**. The decision margin is $0.95 < R_5/R_6 < 1.1$.

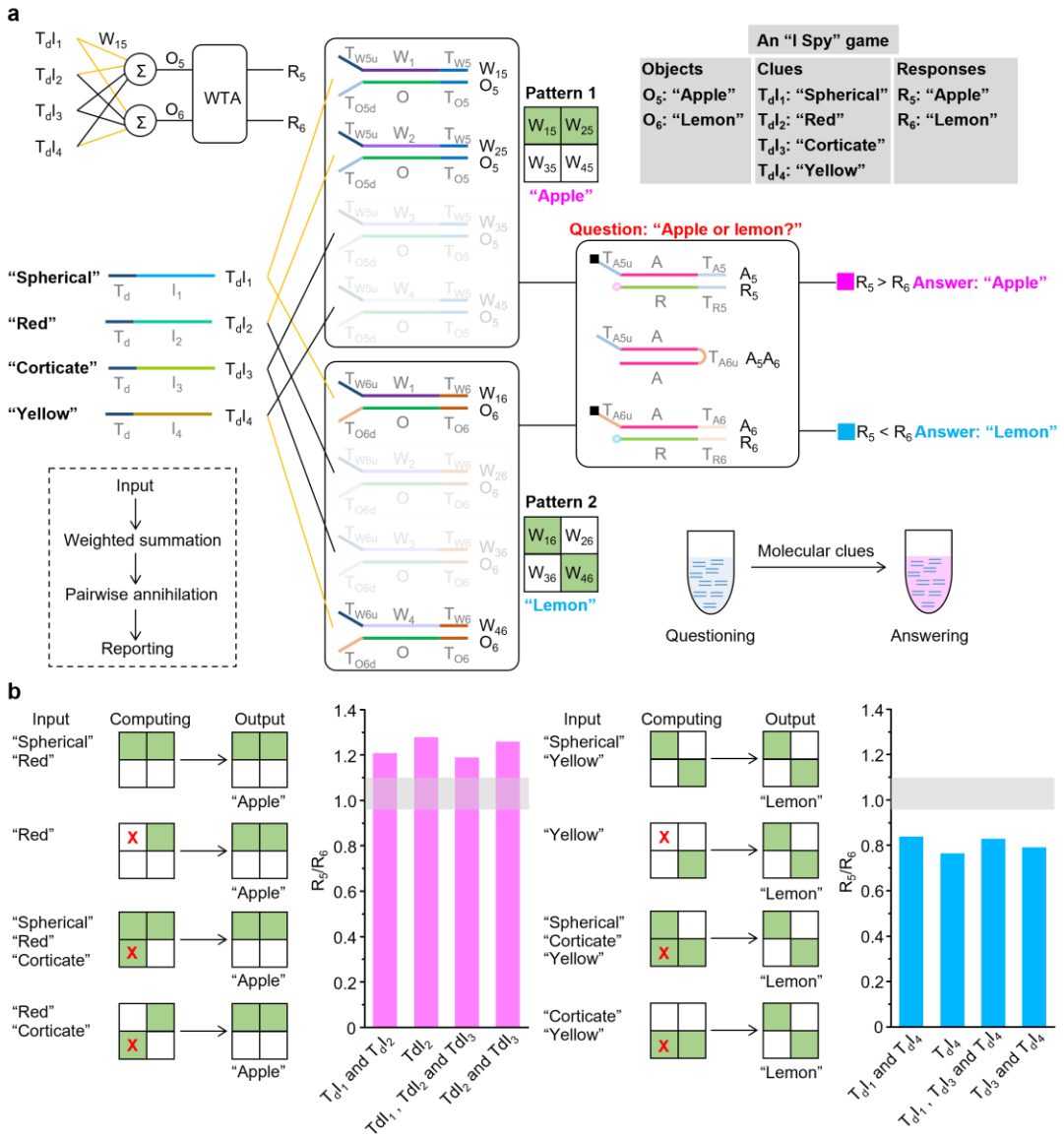

Figure 6. An "I Spy" game based on the 4-bit pattern-recognition non-complementary-DNA-based neural network. (a) Schematic illustration of the non-complementary DNA-based "I Spy" game. **(b)** Experimental implementation of the game with different molecular clues. The results demonstrate that this neural network could recognize input patterns similar to its pattern memories. The decision margin is $0.95 < R_5/R_6 < 1.1$.

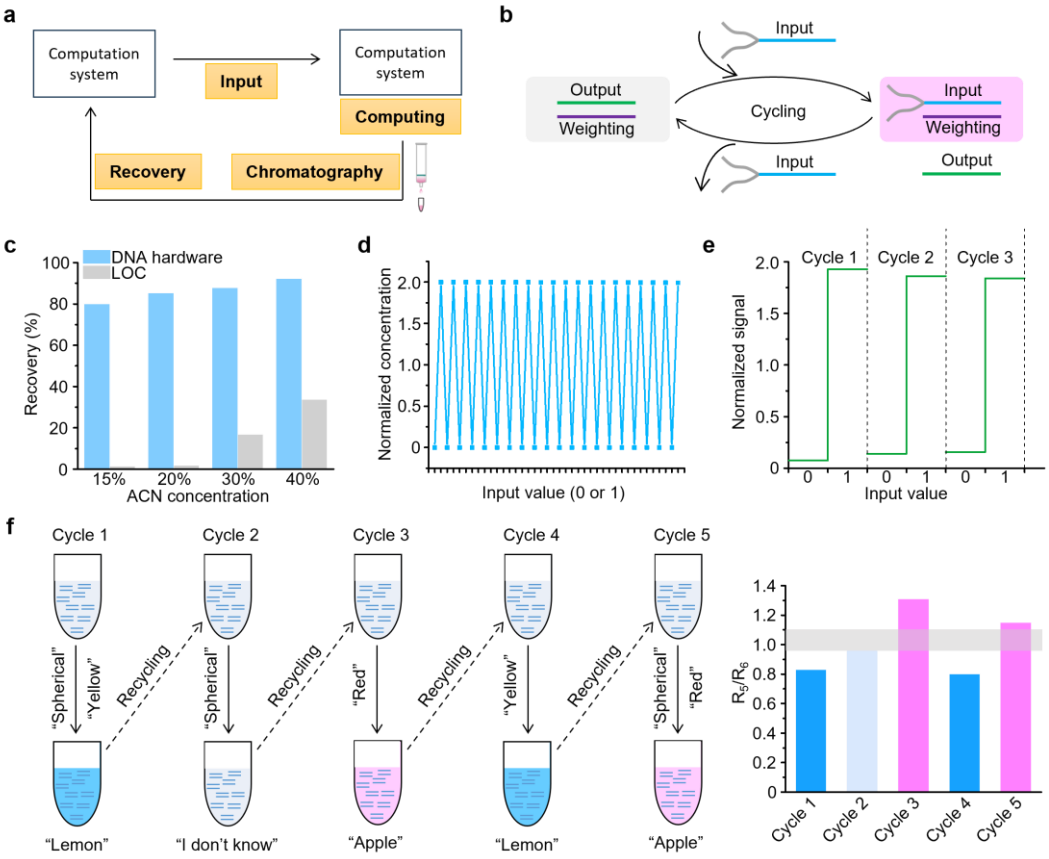

Figure 7. Reusing of the non-complementary-DNA-based neural network. (a) Schematic illustration of the process for reusing of non-complementary DNA-based computation system through chromatography separation. **(b)** Schematic illustration of the NCP reusing with LOC as the removable input strand. **(c)** Chromatography recoveries of the DNA hardware and the LOC input strand with different concentrations of ACN as the eluent. **(d)** Simulated firing performance of the NCP at a recovery level of 80% and constant input concentration. **(e)** Experimental results of NCP_1 reusing is consistent with the simulated results. **(f)** Schematic illustration and experimental implementation of the repeated "I Spy" game with different input clues. The decision margin is $0.95 < R_5/R_6 < 1.1$.