Bulletin of the *Transilvania* University of Braşov
Series III: Mathematics and Computer Science, Vol. 1(63), No. 2 - 2021, 129-138
https://doi.org/10.31926/but.mif.2021.1.63.2.11

DETERMINISTIC FINITE AUTOMATA FOR BRACHYCEPHALY RISK ESTIMATION IN DOG BREEDING

Timea ARANYI[1], Alexandra BĂICOIANU[*,2] and Ioana PLAJER[3]

Abstract

Afflictions, or predispositions to have particular disorders in an animal are often caused by specific genes. However, some of these conditions could be avoided using genetic alterations during the breeding process. A good example is brachycephaly in dogs, which, in many cases, causes dyspnoea. In this research paper, we focused on using Finite Deterministic Automata for pattern recognition in dog genes in order to improve the breathing problems of French bulldogs, and other brachycephalic breeds.

2020 *Mathematics Subject Classification:* 68Q45, 68Q70, 03D05, 20F10, 18B20, 68Q07.

Key words: Formal languages theory, automata theory, deterministic finite automata, nucleotide, DNA, French Bulldog.

1 Introduction

What do computer science, a French bulldog and a roughly eighty year old “machine”, described by two neurophysiologists have in common? Seemingly, nothing. But after thinking about it, one can conclude that both computer science and genetics work based on a set of rules. Thus, using Formal Languages Theory should come as a natural choice.

This research was motivated by observing the breathing difficulties of Dop, a French bulldog we are fond of and is aimed at helping not only French bulldogs, but other breeds that face the same conditions because of their genetic predispositions.

The French bulldog is a ”brachycephalic” breed, meaning that these dogs have shortened heads. Because of the way their heads are shaped, they are genetically

[1] Faculty of Mathematics and Computer Science, *Transilvania* University of Braşov, Romania, e-mail: timea.aranyi@student.unitbv.ro

[2*] *Corresponding author*, Faculty of Mathematics and Computer Science, *Transilvania* University of Braşov, Romania, e-mail: a.baicoianu@unitbv.ro

[3] Faculty of Mathematics and Computer Science, *Transilvania* University of Braşov, Romania, e-mail: i.plajer@unitbv.ro

inclined to have respiratory issues. Since their snout is so short, they can have a few anatomical abnormalities and tend to have difficulty breathing, especially during longer physical activity or in a hot summer. A study [4] conducted between 2011 and 2015 shows that 66% of French bulldogs are affected by respiratory issues, due to "upper airway obstruction", because they are brachycephalic. And that is a very high percentage. Their condition affects their, and their owner's well being and everyday life. Since they have trouble breathing while exercising, they tend to be less active throughout the day. As these dogs really enjoy food, being less active often leads to them being overweight, or even obese, with effect on their heath and their quality of life. If their snouts weren't that short, they would be less likely to develop all those conditions, and they could lead a healthier and happier life.

This research paper aims to help improve the French bulldog's quality of life by modifying the breed using genetic alteration, thus helping them and their owners to lead a less complicated life.

Deterministic Finite Automata (DFA) have already been used for pattern recognition. As a DNA sequence can be regarded as a finite pattern, some approaches of DNA recognition using DFAs can be found in literature. In [7], the authors propose to first create a non-deterministic finite automaton (NFA)to verify if an input string is present in a DNA pattern, then, based on this NFA, construct a deterministic finite automaton, in order to decrease the analysis time of a DNA sequence.

Another related research paper [1] the DNA sequences are being represented numerically, and the Alergia Algorithm is used to create stochastic finite automata, then these automata are used to compare two or more living organisms DNA sequences.

In this research paper we will discuss a way in Automata Theory that can be used to check if an animal has a particular characteristic. To validate our work, a C++ program which simulates a deterministic finite automaton was implemented.

2 Materials and Methods

All the genetic information of an organism is encoded by the DNA. Richard Dawkins calls the DNA molecules "the modern equivalents of the first replicator" [2], emphasizing their self-copying property. A DNA molecule consists of two polynucleotide chains that together form a spiral, called a double-helix.

The nucleotides, which make up the DNA, and play an essential part in this research project, are of four types, and they are the same in every creature on our planet. There is no difference between our nucleotides, and those of a snail; as Richard Dawkins emphasises in his work, "The Selfish Gene" [2]. This means, we are all the same at our core. But then, the question is, what makes all these earthly creatures so different? It all comes down to one seemingly simple thing: the order in which these nucleotides follow each other.

There are four types of nucleotides: adenine (A), thymine (T), cytosine(C) and guanine(G). These nucleotides form pairs, called base pairs: A –T and C – G. Adenine and thymine form two hydrogen bridge bonds, while cytosine and guanine form three. These bonds are the reason the two chains are connected. The genes are meaningful information carrying nucleotide sequences, encoding one or more proteins, as well as their functionality. These genes can be found on chromosomes, which are very long DNA sequences that encode one or more physical traits.

Formal Language Theory is a field of science that studies the patterns and rules in a set of words over a given alphabet and Automata Theory is a subset of Formal Language Theory. An automaton is an abstract mechanism, composed of states, which follows a predetermined sequence of operations. Automata enable the analysis of a system, for instance of a language.

A finite automaton or finite state machine works with a set of symbols, and "jumps" through a set of states using a transition function. It gets a word as input, and iterates through the states, according to the transition function. According to the state it reaches in the end, it either accepts or rejects the word. An accepted word is part of the language the machine can recognize, while a rejected word is not. The automata used in this paper are deterministic finite automata (DFA).

A DFA can be formally described by a tuple $M = (Q, \Sigma, \delta, q_0, F)$, in which Q is a finite, not empty set of states, Σ is a finite, not empty set of symbols, called an alphabet, δ is called transition function and defines how the state of the automaton changes while processing an input word, $q_0 \in Q$ represents the initial state of the automaton and $F \subseteq Q$ is the set of final states i.e., the states of acceptance for the words of the language described by the DFA.

In the following we will illustrate how to build a DFA in order to recognize a pattern in a certain genetic sequence, allowing for the estimation of brachycephaly risk in dog breeding. For this purpose, we are going to assume the following things: the genome or a DNA sequence to work with is known as well as the DNA sequence that encodes the desired characteristic. Our task is to check if this DNA sequence is present in the sample or not. If the sequence is found, it means that the subject has the required trait, if not, it doesn't. This is where DFAs come into action. We can design a finite deterministic automaton to check if a given substring can be found in an input string (in this case a DNA) as is shown in the following two examples.

However, before studying the examples, it should be taken into consideration that the two strands that create a DNA molecule are complementary to each other. Nevertheless, it is enough to only consider one strand, since we can write down the other at any given time, based on the first strand and the pairs presented in the beginning of this section. Another possibility would be, to represent the two strands as one string as T. Head does in [2], in which he analyses the generative capacity of specific recombinant behaviours of the DNA. This can be done, for example by expressing the pair of strands

$$\left\{ \begin{matrix} A & T & G & C & C & G \\ T & A & C & G & G & C \end{matrix} \right\}$$

by the string $[A/T][T/A][G/C][C/G][C/G][G/C]$. In our work, we use only the first strand.

The alphabet considered for the DFAs with all the examples is $\{A, T, C, G\}$, representing the DNA nucleotides. The initial state is q0 in each case, and the number of states is going to be equal to the length of the DNA sequence to find, plus one.

The basic idea of these automata is fairly simple: each of them stays in the initial state until the input equals the first nucleotide from the searched sequence, then it transits to the next state. From there, it either jumps to the next state, if the new input equals the next nucleotide, or goes back to the initial or a previous state, if it doesn't.

A simple example, in which a DNA strand ATGC**CGAA**TTTAGGA and a searched sequence **CGAA** are considered, is illustrated in Figure 1.

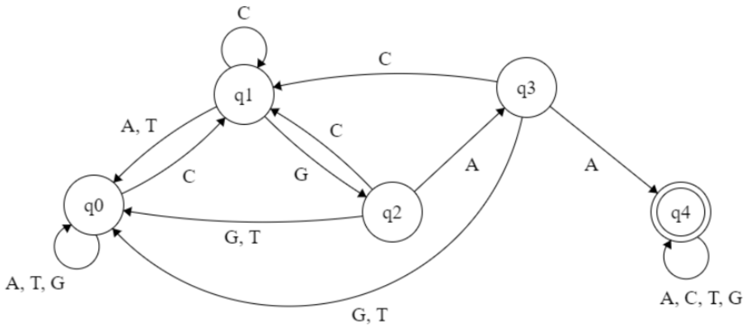

Figure 1: DFA which accepts DNA strands containing **CGAA** string.

The final state of this automaton is q_4, and the transitions are given by the labelled connecting arcs between states. Assuming that the input string is ATGC**CGAA**TTTAGGA, the machine would process it as follows: The first input is A, so the automaton stays in the initial state. For T and G it does the same. Then, the next input is C, so it jumps to q_1. Since the next input is also C, it stays in q_1. Next comes G, and the automaton jumps to q_2, and after that it reads A, so it jumps to q_3. The next input is also A, so it jumps to the final state. Here, it doesn't matter anymore what the input is, as long as it is part of the given alphabet, the automaton stays in q_4 until there is nothing left on the input tape, and it accepts the DNA. So, in q_0 we haven't found any of the nucleotides, in q_1 we've found C, in q_2 CG, in q_3 CGA and in q_4 $CGAA$.

This can be illustrated by following configuration chain:

$$(q_0, ATGCCGAATTTAGGA) \rightarrow (q_0, TGCCGAATTTAGGA) \rightarrow$$
$$(q_0, GCCGAATTTAGGA) \rightarrow (q_0, CCGAATTTAGGA) \rightarrow$$
$$(q_0, CGAATTTAGGA) \rightarrow (q_1, GAATTTAGGA) \rightarrow$$
$$(q_2, AATTTAGGA) \rightarrow (q_3, ATTTAGGA) \rightarrow (q_4, TTTAGGA) \rightarrow$$

$(q_4, TTAGGA) \rightarrow (q_4, TTAGGA) \rightarrow (q_4, TAGGA) \rightarrow$
$(q_4, AGGA) \rightarrow (q_4, GGA) \rightarrow (q_4, GA) \rightarrow$
$(q_4, A) \rightarrow (q_4,)$.
As q_4 is a final state, the sequence is accepted.

In a similar manner, we can design a DFA for real world application, using a DNA sequence susceptible to causing brachycephaly in dogs.

There are a few research articles that talk about genetic health issues in dogs, one of these conditions being brachycephaly. In [6] and [5] some of the genes in which mutations are connected with brachycephaly are discussed. In our paper we will consider one of these, the DVL2 gene, for which a genetic primer subsequence is presented in [5].

CGGCTAGCTGTCAGTTCTGG.

The automaton which accepts this subsequence is presented in Figure 2.

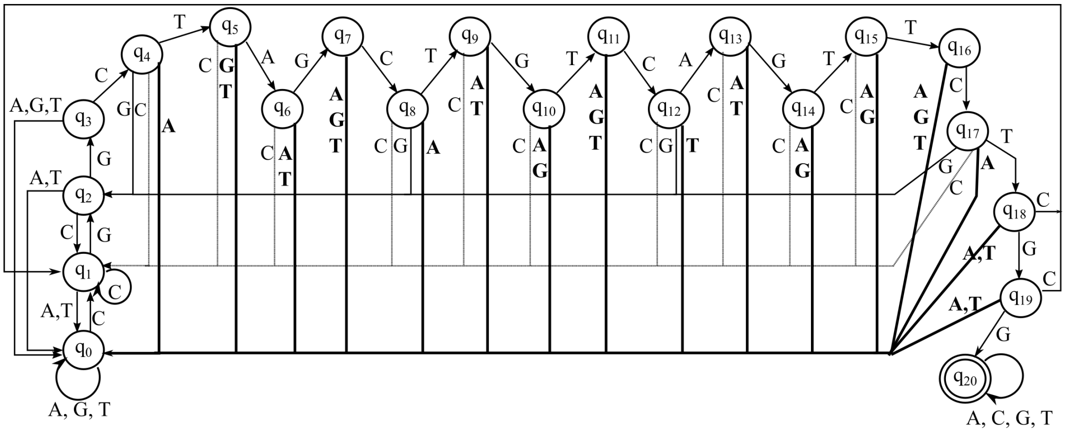

Figure 2: DFA for the DVL2 gene.

The configuration chain, which illustrates the acceptance is given by:

$(q_0, CGGCTAGCTGTCAGTTCTGG) \rightarrow$
$(q_1, GGCTAGCTGTCAGTTCTGG) \rightarrow$
$(q_2, GCTAGCTGTCAGTTCTGG) \rightarrow$
$(q_3, CTAGCTGTCAGTTCTGG) \rightarrow$
$(q_4, TAGCTGTCAGTTCTGG) \rightarrow$
$(q_5, AGCTGTCAGTTCTGG) \rightarrow$
$(q_6, GCTGTCAGTTCTGG) \rightarrow$
$(q_7, CTGTCAGTTCTGG) \rightarrow$
$(q_8, TGTCAGTTCTGG) \rightarrow$
$(q_9, GTCAGTTCTGG) \rightarrow$
$(q_{10}, TCAGTTCTGG) \rightarrow$

$(q_{11}, CAGTTCTGG) \to$
$(q_{12}, AGTTCTGG) \to$
$(q_{13}, GTTCTGG) \to$
$(q_{14}, TTCTGG) \to$
$(q_{15}, TCTGG) \to$
$(q_{16}, CTGG) \to$
$(q_{17}, TGG) \to$
$(q_{18}, GG) \to$
$(q_{19}, G) \to$
$(q_{20},)$.
As q_{20} is a final state, the sequence is accepted.

In case of mutations, the automaton does not accept the sequence and outputs a non valid sequence. This can be illustrated for some hypothetic mutations. By example if a deletion / frameshift mutation [5] occurs, by deleting the 12th nucleotide, we obtain the subsequence:

CGGCTAGCTGTAGTTCTGG.

For this subsequence, the automaton will produce the following result:

$(q_0, CGGCTAGCTGTAGTTCTGG) \to$
$(q_1, GGCTAGCTGTAGTTCTGG) \to$
$(q_2, GCTAGCTGTAGTTCTGG) \to$
$(q_3, CTAGCTGTAGTTCTGG) \to$
$(q_4, TAGCTGTAGTTCTGG) \to$
$(q_5, AGCTGTAGTTCTGG) \to$
$(q_6, GCTGTAGTTCTGG) \to$
$(q_7, CTGTAGTTCTGG) \to$
$(q_8, TGTAGTTCTGG) \to$
$(q_9, GTAGTTCTGG) \to$
$(q_{10}, TAGTTCTGG) \to$
$(q_{11}, AGTTCTGG) \to$
$(q_0, GTTCTGG) \to$
$(q_0, TTCTGG) \to$
$(q_0, TCTGG) \to$
$(q_0, CTGG) \to$
$(q_1, TGG) \to$
$(q_0, GG) \to$
$(q_0, G) \to$
$(q_0,)$.
As state q_0 is not a final state, the sequence is rejected and the conclusion is, that it is not a valid configuration of the gene.

In case of a substitution mutation, in which a nucleotide is altered, for example the 4th nucleotide C is replaced by G, then the automaton produces the following output:

$(q_0, CGGGTAGCTGTCAGTTCTGG) \rightarrow$
$(q_1, GGGTAGCTGTCAGTTCTGG) \rightarrow$
$(q_2, GGTAGCTGTCAGTTCTGG) \rightarrow$
$(q_3, GTAGCTGTCAGTTCTGG) \rightarrow$
$(q_0, TAGCTGTCAGTTCTGG) \rightarrow$
$(q_0, AGCTGTCAGTTCTGG) \rightarrow$
$(q_0, GCTGTCAGTTCTGG) \rightarrow$
$(q_0, CTGTCAGTTCTGG) \rightarrow$
$(q_1, TGTCAGTTCTGG) \rightarrow$
$(q_0, GTCAGTTCTGG) \rightarrow$
$(q_0, TCAGTTCTGG) \rightarrow$
$(q_0, CAGTTCTGG) \rightarrow$
$(q_1, AGTTCTGG) \rightarrow$
$(q_0, GTTCTGG) \rightarrow$
$(q_0, TTCTGG) \rightarrow$
$(q_0, TCTGG) \rightarrow$
$(q_0, CTGG) \rightarrow$
$(q_1, TGG) \rightarrow$
$(q_0, GG) \rightarrow$
$(q_0, G) \rightarrow$
$(q_0,)$.
Again q_0 is not a final state and the string is rejected.

These mutations are not real case ones, as we do not have enough insight into the genetic details, but in our opinion, they illustrate well enough the mechanism of acceptance, respectively rejection.

3 Discussions

Automata Theory could help identify genetic problems in dogs and so contribute to the breeding of dogs in general and French bulldogs in particular, in a way that reduced the probability of health problems like those induced by brachycephaly. Assuming that we know the genome of both parents, and we can deduce the genome of the offspring, we could use an automaton to check if the offspring is going to be affected by upper airway abnormalities. If the automaton finds that the sequence which causes the malformations is present in the DNA of the offspring, the breeder can choose not to breed the two dogs.

Another way in which Automata Theory can help these dogs is by helping to modify the breed, so that their head isn't shortened anymore. This could be done with crossbreeding. For this to work we would need to know the DNA of the offspring of the two different breed dogs, in this case a French bulldog and another small breed dog, and what the DNA sequence that encodes the characteristics of the nose looks like. Then we could design an automaton to search for that sequence and to determine if the nose would be as we want it or not.

There are three main types of genetic mutations [2, 8], with a wide variety of consequences, that cause diversity among living organisms. These mutations are: single base substitutions, deletions and insertions. Single base substitutions or point mutations are the most common mutations, and can be separated in two categories: transition and transversion. A transition is when a purine replaces another purine, or when a pyrimidine replaces a pyrimidine. When a purine replaces a pyrimidine or vice versa, the mutation is called a transversion.

Point mutations are of three types: silent, missense and nonsense. Silent mutations, as the name suggests, are mostly harmless. When a mutation doesn't change the amino acid sequence, we talk about a silent mutation. Missense mutations can be damaging or have little effect, depending on the type of amino acid substitution. Nonsense mutations most likely lead to nonfunctional proteins.

Deletions and insertions are pretty self-explanatory, and depending on the number of lost or inserted base pairs, they can have deleterious to no effect.

The main causes of mutations are errors in DNA replication or in DNA recombination, radiations or chemical damage to the DNA [8]. However, the most common causes are errors in the DNA recombination or replication.

Knowing that not all mutations have an effect on the health of an organism (in this case, a dog), there can be certain mutations that the automaton accepts. However, major and harmful mutations won't be accepted. Considering this, the automaton can accept or reject a sequence, or it could be modified by adding a new final state in which it gets due to a specific mutation in the nucleotide sequence. In this case, if there is enough data, different mutations can be recognized and by adding further functionality and variables to each state of the automaton, the exact position of the mutation can be found. In collaboration with geneticists the automata could be adjusted as described above.

4 Conclusions

Languages and automata are elegant and robust concepts that can be found in every area of computer science. Automata theory is important conceptually as a simple computational model that we understand well, and regular expressions and automata have many real-life applications. The automata-theoretic approach to decision procedures is one of the most fundamental approaches to decision procedures. Recently, this approach has found biological and biomedical applications. In this context, we focused on using automata for pattern recognition in dog genes. The proposed method does not completely solve the problem because it doesn't review every change that happens when crossbreeding. Crossbreeding won't only change the look of the nose; it is going to change every other physical aspect of the animal, yet it might cause other genetic predispositions, and none of these are considered by our automata. There is an entire field specialized in genetic alterations, called genetic engineering, which offers researchers real opportunities to help French bulldogs and every other brachycephalic breed. This paper describes a case study performed using DNA sequences, in order to evaluate and improve breathing problems of the French bulldogs, and other brachycephalic breeds. It proposes a new way of using finite deterministic automata, which might lead to further research in the future.

References

[1] Achrekar, P.P. , *Pattern Recognition of DNA Sequences using Automata with emphasis on Species Distinction*, master thesis, San Jose State University, 2013.

[2] Dawkins, R., *The Selfish Gene*, Oxford University Press, 1976.

[3] Head, T., *Formal language theory and DNA: An analysis of the generative capacity of specific recombinant behaviours*, Bulletin of Mathematical Biology **49** (1987), no. 6, 737-759.

[4] Liu, N-C., Adams, V.J., Kalmar, L., Ladlow,J. F. and Sargan, D.R., *Whole-body barometric plethysmography characterizes upper airway obstruction in 3 brachycephalic breeds of dogs*, Journal of Veterinary Internal Medicine, **30** (2016), no. 3, 853-865

[5] Mansour, T. A., Lucot, K., Konopelski, S.E., Dickinson, P.J., Sturges, B.K., Vernau, K.L., Choi, S., Stern, J.A., Thomasy, S.M., Döring, S., Verstraete, F.J.M., Johnson, E.G., York, D., Rebhun, R.B., Ho, H., Brown, C.T. and Bannasch, D.L., *Whole genome variant association across 100 dogs identifies a frame shift mutation in DISHEVELLED 2 which contributes to Robinow-like syndrome in Bulldogs and related screw tail dog breeds* PLoS Genetics **14** (2018), no. 12, https://doi.org/10.1371/journal.pgen.1007850

[6] Niskanen, J., Reunanen, V., Salonen, M., Bannasch, D., Lappalainen, A.K., Lohi, H. and Hytönen, M.J., *Canine DVL2 variant contributes to brachycephalic phenotype and caudal vertebral anomalies*, Human Genetics **140** (2021), article 4106, 1535-1545, https://doi.org/10.1007/s00439-021-02261-8

[7] Qura-Tul-Ein, Saeed, Y., Naseem, S., Ahmad, F., Alyas, T. and Tabassum, N., *DNA Pattern analysis using finite automata*, International Research Journal of Computer Science (2014), October.

[8] ***, DNA Mutation and Repair, http://www2.csudh.edu/nsturm/\\CHEMXL153/DNAMutationRepair.htm.