Univerzita Komenského v Bratislave
Fakulta matematiky, fyziky a informatiky

Sampling as an approach to sequencing MinION data

Bakalárska práca

2017
Matúš Zeleňák

Univerzita Komenského v Bratislave
Fakulta matematiky, fyziky a informatiky

Sampling as an approach to sequencing MinION data

Bakalárska práca

Študijný program:	Informatika
Študijný odbor:	2508 Informatika
Školiace pracovisko:	Katedra informatiky
Školiteľ:	Mgr. Tomáš Vinař PhD.

Bratislava, 2017
Matúš Zeleňák

Poďakovanie: Týmto by som sa chcel poďakovať Tomášovi Vinařovi a Broni Brejovej za odbornú, svojej priateľke za morálnu a stránke LOTR memes za humornú podporu.

Abstrakt

MinION je platforma sekvenovania DNA, ktorá produkuje dlhé čítania s vysokým stupňom chybovosti. Jednou z príčin vysokej chybovosti je to, že elektrický signál, ktorý prístroj produkuje, je potrebné preložiť do DNA sekvencií a tento proces je zdrojom veľkého množstva chýb. Je známe, že skryté Markovove modely (HMM) sú rozumnou aproximáciou fungovania platformy MinION. Cieľom tejto práce je využiť vzorkovanie v HMM a vyhnúť sa tak prekladu do sekvencií pri úlohách, ako je zarovnávanie čítaní ku genómu, či zarovnávanie čítaní navzájom. Takýto prístup môže pomôcť zvýšiť sensitivitu zarovnaní.

Kľúčové slová: skrytý Markovov model, vzorkovanie, DNA , seeding

Abstract

MinION DNA sequencing platform produces long reads with high error rates. One of the reasons for high error rates is that electrical signals produced by the sequencing machine need to be first translated into DNA sequences by a process called base calling, and this process is error prone. It is known that hidden Markov models (HMMs) are a reasonable model for this platform. The goal of this thesis is to use sampling from HMMs to avoid base calling in tasks such as read to genome and read to read alignment, which could help to improve sensitivity of these tasks.

Keywords: hidden Markov model, sampling, DNA, seeding

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Lorem ipsum

Chapter 2

Background

This chapter should enlighten the reader in relevant topics that will be used throughout the thesis and should serve as a reference point for basic definitions.

2.1 DNA

Organisms have most of their working principles encoded into a deoxyribonucleic acid (abbreviated DNA). This molecular structure made of two complementary strands of nucleotides contains all the relevant information necessary for growth, functioning and reproduction of the carrier organism.

From the informatics standpoint, a single strand of DNA can be viewed as a finite length string over the four symbol alphabet. The symbols represent the DNA nucleotides (bases) which are the building blocks of nucleic acid and are labeled A, C, G, T for adenine, cytosine, thymine and guanine respectively.

Given one strand of the DNA, it is simple to determine the complementary strand by utilizing base-pairing rules: A always bonds with C and G bonds with T. These bonding rules are symmetrical.

2.2 Sequencing DNA

Given the importance of DNA, it is only natural that we attempt to retrieve its structure from organisms in a process called DNA sequencing. The result of a sequencing run is a sequence *read* – a string consisting of the aforementioned nucleotide symbols. Considering the size of an average genome (e.g. 3 giga base pairs for human genome) and the delicacy of such process, it is difficult to obtain long uninterrupted sequences of DNA in one pass.

Over the past decades, technological progress has given rise to three generation of sampling approaches.

2.3 Nanopore sequencing and MinION platform

As of May 2015 a new platform released by Oxford Nanopore became available for public use - MinION. The core of the platform is a phone-sized device which is connected to a computer via USB. Sequencing is performed by threading a single strand of DNA through a protein nanopore. The nanopore is embedded in a membrane made of synthetic polymers to which an electric current is applied. There are hundreds of such nanopores in one device, each of which generates a stream of data.
As a part of the DNA strand passes through the nanopore, it induces a characteristic disruptions of the current. These disruptions are measured thousands of times per second and based on the data *events* are identified. Events describe a specific *k-mer* (a continuous sequence of k nucleotide bases). The process of assigning k to events is called base calling. In ideal conditions, the speed at which the strand passes through the nanopore would be constant and there would be a deterministic way to map the disruptions to *kmers* that caused them. Unfortunately, that is not the case and even the latest version of MinION (the R9.4 released October 2016) has up to 13% error rate.
In order to deal with the errors, processing of reads (sequences of events) by probabilistic models is necessary. Even though the R9.4 uses recurrent neural networks, using hidden Markov models (which were officially used in R7) is still considered sufficient.

2.4 Hidden Markov model

We shall now introduce a probabilistic model that lets us simulate the behaviour of MinION platform. Hidden Markov models are a machine learning concept that finds its use in many areas : speech and handwriting recognition, image processing and bioinformatics to name a few.

Definition [3]: Let $H = (Q, S, E, q_0, e, t)$ where Q is a non-empty finite set of states, $S \subseteq Q$ is a set of silent states, E is a an uncountably infinite set of emissions, $q_0 \in S$ is a initial state, $e : Q \times E \to [0, 1]$ is an emission probability function and $t : Q \times Q \to [0, 1]$ is a transition probability function. H is a hidden Markov model when the following criteria are satisfied:

1. $\forall u \in Q : t(u, q_0) = 0$
2. $(\forall u \in S)(\forall e \in E) : e(u, e) = 0$

3. $\forall u \in Q : \sum_{v \in Q} t(u, v) = 1$

4. $\forall u \in Q \setminus S : \int_{x \in E} e(u, x)dx = 1$

This model is structurally similar to a finite automaton and can be visually represented as a directed graph. In such graph the states would be the members of set Q and arrows would mark transitions between states u, v that satisfy $t(u, v) > 0$. Initial state is drawn as two concentric circles while other states are just a circle.
The HMM we use is a generative model. It is used to generate two sequences - a sequence of states and a sequence of emissions. This Markov model is called hidden because we do not actually get to see the state sequence, only observe the resulting emissions.

The HMM works as following: starting from the initial state q_0 (which is silent) it transitions to some random state $q_i \in Q \cup S$ with a probability defined by $t(q_0, q_i)$. If the state $q_i \notin S$, it emits a random emission $e \in E$ according to the probability defined by $e(q_i, e)$. If the state $q_i \in S$, it emits nothing. Afterwards, the process of transitioning into next state happens and the automaton can continue to run this way indefinitely. For practical purposes we are only interested in transitional paths of finite length.

By using HMM defined above we can determine the joint probability over the pairs $P(e \wedge s)$ where e is the sequence of observed emissions and s is a description of the path the HMM took in order to emit e. The probability can be calculated as

$$P(e \wedge s) = (\prod_{i=1}^{m} t(s_{i-1}, s_i)) \cdot (\prod_{i=1}^{n} e(v_i, e_i))$$

where $e = e_1, e_2, \ldots, e_n$ is the sequence of emissions , $s = s_0, s_1, s_2, \ldots, s_m$ is the path (sequence of hidden states) and $v = v_1, v_2, \ldots, v_n$ is a subsequence (not necessarily continuous) of all non-silent states of sequence s. For future use we define $s_0 := q_0$.

Let us now construct a simple HMM with states that represent all possible DNA *kmers* of a particular length. The emissions would in turn match the electric current levels from the MinION. The transitional and emission probabilities are gathered from the already known data of MinION sequencing. One can see that such HMM would relatively accurately model the behaviour of the MinION platform during sequencing. We can now begin to extract useful informations from such HMM. Suppose we obtained a sequence of electric current levels from a MinION sequencing run. In order to determine the DNA sequence that would best match with the current levels we can find a path of states in the HMM that would emit the sequence of current levels and at

the same time is the most probable of all the paths emitting such sequence. Finding such path is usually done by using dynamic programming algorithms, such as Viterbi or Forward algorithm. [2]

2.5 Sampling from HMM

While finding the most probable path of hidden states is useful, many times we would like to take *samples* from the HMM instead. Samples are sequences of hidden states with suboptimal probabilities that still retain biological significance.
Formally, given a $H = (Q, S, E, q_0, e, t)$ and an emission sequence f we are constructing a set $R = \{s_1, s_2, \ldots, s_k\}$ where s_i is a path of hidden states in H such that:

$$\forall s_i \in R : P(s_i \wedge f) > 0$$

Moreover, the probability that path s_i belongs to the generated set R equals the probability of the HMM taking the path s_i during its computation.

Nowadays, there are algorithms capable of generating such set given a HMM [1], however their running time is slow and are therefore ill-fit for datasets that are as vast as the ones generated by MinION platform. The aim of this thesis is to speed up the process of sampling by utilizing heuristic methods and hardware acceleration.

2.6 Alignment of sequences

The alignment of DNA sequences is a process of rearranging the given sequences in order to find similarities. The rearrangement is not arbitrary – the relative order of nucleotides in the sequence must remain the same and the sequence itself is only modified by inserting gap symbols. Most usually the alignment is performed by selecting a scoring function that rewards long consecutive matches between sequences and punishes substitutions and gap openings, then finding a rearrangement that maximizes the score of the whole sequences (global alignment) or a subsequences (local alignment)
The idea of sequence alignment is to rearrange the sequences in such a way, that they look as similar as possible when comparing them by symbols. Such alignment is helpful when comparing the genome to a database, determining the function of the sequence or study of the evolutionary processes.

In order to gain insight consider this simple version of the problem for finding the optimal global alignment formalized as following:
Let $S_1 = s_1^1 s_2^1 \ldots s_n^1$ and $S_2 = s_1^2 s_2^2 \ldots s_m^2$ where $S_1, S_2 \in \{A, C, G, T\}^*$ be the two genome sequences we would like to align. Let $f(x, y)$ where $x, y \in \{A, C, G, T, -\}$ be a scoring function for the alignment of two individual bases or gaps (annotated $-$).

We are looking for $A_1 = a_1^1 a_2^1 \dots a_k^1, A_2 = a_1^2 a_2^2 \dots a_k^2$ where $A_1, A_2 \in \{A, C, G, T, -\}^*$ and $k = max(n, m)$ such that S_1 is a subsequence (not necessarily continuous, the the remaining symbols are gaps) of A_1, likewise for S_2 and A_2. Moreover, the A_1, A_2 should satisfy the condition:

$$A_1, A_2 = \arg\max_{A_1, A_2} \psi(A_1, A_2)$$

where $\psi(X, Y)$ is an aggregation function which takes the results of $f(A_i^1, A_i^2) \forall i \in 1..k$ and calculates the final score for the alignment of A_1 and A_2.

In a more concrete example, the f could give a positive number if the input arguments are equal, and negative value otherwise. The aggregation function ψ can simply be a summation over all $f(x_i, y_i)$. In practice, more elaborate scoring schemes can be used, e.g. ones that punish continuation of gap less than opening a new one etc.

The problem of finding the best alignment can be solved by dynamic programming algorithms such as Smith-Waterman for local alignment or Needleman-Wunsch for global alignment.

The advantage of the above mentioned algorithms is that they always find the optimal alignment. Their hindrance though is their time and space complexity of $O(nm)$ where n, m are the lengths of the aligned sequences. For common genomes with length in order of millions base pairs (recall that human genome has 3Gbp) such slow and consuming algorithms are of little to no use.

As a consequence we are forced to use heuristic methods instead. One of such is called *seed and extend* strategy and typically follows the process of these steps:

1. Finding the seeds : the algorithm finds *kmers* of fixed size that occur in both of the aligned sequences. This is usually done by using hash tables with the *kmers* as a key and the position in the sequence as the value. We call these exact matches *seeds*

2. Extension without using gaps : starting at the position of a particular seed in a sequence, the algorithm attempts to extend them in both ways to get longer alignment without inserting gaps in either sequence. It is also beneficial to connect two nearby seeds by extending them

3. Extension with using gaps: take the already generated alignment and extend it by using gaps to concatenate it with neighboring alignments

4. Extension by dynamic programming: Use a deterministic algorithm (such as Smith-Waterman or Needleman-Wunsch) to further expand the alignment

This strategy is employed in the widely-used NCBI-BLAST. Compared to the exact algorithms, seed and extend offers significant speed boost at the expense of alignment precision.

In our thesis, we aim to investigate alternative approaches to sampling by seeding that would enable us to efficiently align not just pairs of sequences, but rather entire sets of samples generated from HMM.

Chapter 3

Sampling algorithm design and implementation

Here we should describe the though process that led to the conception of our sampling algorithm for HMM.

Chapter 4

Seeding algorithm design and implementation

Following paragraphs offer a brief overview of the process that gave birth to the seeding stragedy and algorithm.

Chapter 5

Experimental comparison with existing methods

Here we shall compare our algorithms with already existing ones.

Chapter 6

Conclusion

Final chapter that concludes the results of our work.

Bibliography

[1] Simon L Cawley and Lior Pachter. HMM sampling and applications to gene finding and alternative splicing. *Bioinformatics*, 19(suppl 2):ii36–ii41, 2003.

[2] Richard Durbin, Sean R. Eddy, Anders Krogh, and Graeme Mitchison. *Biological Sequence Analysis: Probabilistic Models of Proteins and Nucleic Acids*. Cambridge University Press, 1998.

[3] Rastislav Rabatin. Alignment of nanopore sequencing reads, 2016. bachelor thesis at Comenius University.