BMC Bioinformatics

RESEARCH Open Access

A mapping-free natural language processing-based technique for sequence search in nanopore long-reads

Tomasz Strzoda[1], Lourdes Cruz-Garcia[2], Mustafa Najim[2], Christophe Badie[2] and Joanna Polanska[1*]

*Correspondence:
joanna.polanska@polsl.pl

[1] Department of Data Science and Engineering, Silesian University of Technology, Gliwice, Poland
[2] Cancer Mechanisms and Biomarkers Group, Centre for Radiation, Chemical and Environmental Hazards, UK Health Security Agency, Oxfordshire OX11 0RQ, United Kingdom

Abstract

Background: In unforeseen situations, such as nuclear power plant's or civilian radiation accidents, there is a need for effective and computationally inexpensive methods to determine the expression level of a selected gene panel, allowing for rough dose estimates in thousands of donors. The new generation in-situ mapper, fast and of low energy consumption, working at the level of single nanopore output, is in demand. We aim to create a sequence identification tool that utilizes natural language processing techniques and ensures a high level of negative predictive value (NPV) compared to the classical approach.

Results: The training dataset consisted of RNA sequencing data from 6 samples. Multiple natural language processing models were examined, differing in the type of dictionary components (word length, step, context) as well as the encoding length and number of sequences required for algorithm training. The best configuration analyses the entire sequence and uses a word length of 3 base pairs with one-word neighbor on each side. For the considered FDXR gene, the achieved mean balanced accuracy (BACC) was 98.29% and NPV was 99.25%, compared to *minimap2's* performance in a cross-validation scenario. The next stage focused on exploring the dictionary components and attempting to optimize it, employing statistical techniques as well as those relying on the explainability of the decisions made. Reducing the dictionary from 1024 to 145 changed BACC to 96.49% and the NPV to 98.15%. Obtained model, validated on an external independent genome sequencing dataset, gave NPV of 99.64% for complete and 95.87% for reduced dictionary. The salmon-estimated read counts differed from the classical approach on average by 3.48% for the complete dictionary and by 5.82% for the reduced one.

Conclusions: We conclude that for long Oxford nanopore reads, a natural language processing-based approach can reliably replace classical mapping when there is a need for fast, reliable and energy and computationally efficient targeted mapping of a pre-defined subset of transcripts. The developed model can be easily retrained to identify selected transcripts and/or work with various long-read sequencing techniques. Our results of the study clearly demonstrate the potential of applying techniques known from classical text processing to nucleotide sequences.

Keywords: Natural language processing, Machine learning, Sequencing, Alignment, Transcriptomics

Introduction

Understanding the DNA code and searching for specific sequences within them has been a subject of research for years [1]. It has led to a number of discoveries and innovations, bringing different ways of sequencing the obtained reads [2], to which one then tries to assign an origin. The first ways of reading nucleotides, such as the Sanger method [3] and the Maxam–Gilbert method [4], created a good starting point for development, later extended by the Illumina technology [5]. It is part of the so-called second-generation NGS (Next Generation Sequencing), significantly speeding up the sequencing process. Despite its wide popularity, the limitation of short reads has led to the emergence of TGS (Third Generation Sequencing) methods, one representative of which is the company Oxford Nanopore Technologies (ONT) with its sequencing approach [6]. It is characterized by the ability to read much longer reads, whose average length is measured in thousands of bp (base pairs), compared to hundreds of bp for NGS [7]. The technique proposed by ONT involves passing nucleic acids through nanopores (protein channels), thereby causing changes in the measured electrical signal used for sequence identification. This results in a short sequencing time, while preserving the native form of DNA/RNA [8].

Regardless of the technique used, identifying the occurrence of a given genome fragment is an essential task. This information allows, among other things, the discovery of new treatments and therapies. The current approach is based on the use of a mapping process, which involves comparing a read to a certain reference sequence. Appropriate software, called an aligner, analyses the match between the two nucleotide sequences and determines the most likely location of the read on the reference. The main alignment algorithms include Needleman-Wunsch [9] and Smith-Waterman [10], representing dynamic programming, and one of the available and ready-to-use tools is minimap2 [11, 12], which supports ONT long-reads.

The described way of aligners works has some limitations due to its generality. Performing a full analysis provides the sequence match location, which is not useful for tasks such as classification. In addition, the mapping time itself is related to the length of the sequence, the number of reads, the aligner used and the available computing resources. However, in some situations, the most important information is the occurrence (or not) of the sought-after sequence in a long-read, disregarding the exact location or differences in matching. This paper considers the exemplary problem of ionizing radiation, being a permanent element of the environment in today's life, without which the surrounding world is difficult to imagine. Despite the perception as a harmful factor, it occurs in basic medical procedures such as lung X-rays and CT scans. Moreover, radiation is an integral part of nuclear power stations, which have historically experienced various types of accidents (Chernobyl, Fukushima). Unfortunately, it becomes a threat difficult to detect due to invisibility and insensibility. Therefore, research is still ongoing to find potential markers to help in the task. Based on the available literature [13], there are a number of genes that appear to be suitable for biological dosimetry. Our main goal was to propose a mapping-free solution

to find the sought-after sequence in a set of long-reads and provide an alternative to classical bioinformatics methods. The idea we presented, called 'noMapping mapping', consisted of replacing the aligner (and its sequence matching algorithm) with a machine learning model using techniques known from natural language processing (NLP). The classifier system created, hereafter referred to as noMapper, was designed to find ONT long-reads that potentially contain the sought-after sequence. Such filtering allows undesirable reads to be discarded at an early stage and permits further (more detailed) analysis, for example, determining absorbed radiation dose in order to serve as a screening test.

The main NLP technique was the 'bag of words' method, allowing the text to be converted into a vector of numbers that makes possible further calculations. A similar approach was proposed in a paper [14], which focused on finding the viral genome. It used NGS data as the study material and employed de novo genome assembly, distinguishing it from our solution. Another paper [15] used the 'TF-IDF' technique, being an NLP alternative to 'bag of words'. The authors therein focused on detecting regions of lateral origin, relying on the frequency distributions of k-mers in the sequences. In addition, our previous studies [16–18] have shown the potential to explore the solution described in this paper in more depth.

As far as we are aware, it is the first work analysing ONT long-reads to identify the sought-after gene, using such NLP encoding and not requiring other time-consuming preprocessing operations. The outcomes shown in this paper focus on the analysis of the FDXR gene, but the approach used can easily be applied to other sought-after sequences as well (confirmed for the NACA gene and described later). Examining the expression of the FDXR gene, which plays an important role in the cellular response to ionising radiation, can be one tool to assess radiation dose [19]. This gene is a reliable biomarker of radiation exposure, as its expression levels are closely related to the amount of exposure. It is also rapidly induced after radiation exposure, making it a useful tool for detecting exposure in real-time or near real-time. In addition, FDXR is upregulated in many tissue types, increasing its value as a biomarker of radiation exposure in different biological systems. Combining FDXR with other genes involved in radiation response may increase the accuracy and comprehensiveness of assessing cellular response to radiation. Researchers are creating gene expression panels that include FDXR to improve the predictive power of gene expression analyses in different radiation exposure scenarios.

The noMapper repository is available at https://github.com/ZAEDPolSl/noMapper. In addition to the source code, the repository provides detailed instructions on how to use the tool, along with a pre-built model and encoder specifically designed for detecting irradiation marker. If you wish to create a custom version of noMapper tailored to solving different tasks and identifying other genes/markers, the repository includes a comprehensive guide that walks you through the necessary steps. One notable feature is the ability to use your own dataset, consisting of raw sequences directly from sequencing. Users aiming to prepare their version of noMapper do not need to perform any preprocessing; they simply need to run the implemented pipeline, which will generate the model and system encoder. Once the custom version is set up, its operation and usage will be similar to the provided noMapper version focused on irradiation marker detection.

Methods

Data

Two datasets, marked as I and II, were used in this study. The first one, RNA sequencing dataset, was utilized for all the numerical experiments performed, both for training and pre-testing the machine learning models (subsection 'Experimental design'). The second was a genome sequencing dataset, with different properties, which was used solely as an independent validation set, thus verifying the final effectiveness of the proposed solution (subsection 'Testing on an external dataset').

Dataset I

Dataset I contained Oxford Nanopore long-reads. Full-length sequencing was performed on a GridION sequencer with libraries prepared using the direct RNA sequencing kit (SQK-RNA002). All available data were generated from three repetitions of the biological experiment: A, B, C. They were prepared using the HT1080 cell line, which was purchased from the American Type Culture Collection (ATCC). In each repetition, the cells in T-25 flasks were exposed to a 10 Gy X-rays dose. Once irradiated, the samples were incubated at 37 °C with 5% CO2 for 24 h. The cell line was maintained in Minimum Essential Medium (MEM) containing 10% FBS (fetal bovine serum) and 1% penicillin/streptomycin.

In the subsequent stages, RNA extraction was carried out using the RNeasy Mini kit following the manufacturer's instructions. Quantity of isolated RNA was determined by spectrophotometry with a ND-1000 NanoDrop and quality was assessed using a Tapestation 2200. The resulting dataset consisted of six samples, three of which were non-irradiated (A1, B1, C1) and a further three samples 24 h after exposure (A2, B2, C2). Eventually, ONT sequencing yielded 8.5 million RNA long-reads.

Dataset II

Dataset II was an excerpt from ONT's GM24385 open dataset (SRE version after base-calling with Guppy 5.0.6). It contained high molecular weight DNA from lymphoblastoid cells, representing the human genome. However, the present work did not use all the available sequences, but only two chromosomes: one on which the FDXR gene is located (no. 17) and the other randomly selected containing a relatively similar number of sequences (no. 14). More information on the entire shared GM24385 dataset can be found [20].

Data preparation

The sequencing data were subjected to several procedures to prepare them for further analysis. The whole process started with finding and removing possible adapters using Porechop [21]. Filtering was then carried out, thereby removing short reads. This step was performed under the assumption that they could be the result of various errors, which would not have a positive impact on the functioning the entire proposed solution. In order to determine the threshold value, an Empirical Cumulative Distribution Function (ECDF) was drawn for each sample from dataset I. One of these is shown in supplementary Fig. S1. A value was chosen as the cut-off threshold for which the

beginning of the graph line to its right had a steep slope to the horizontal axis, while not excluding too many reads from further analysis. After detailed comparative analysis across all samples, the cut-off value was set to 500 bp. Therefore, all reads whose length was less than 500 bp were discarded. Finally, almost 7.2 million reads remained in the dataset I for further use. For each sample, descriptive statistics of location were calculated, in the form of string length quartiles (Q1, median, Q3). The overall summary is presented in Table 1.

Making a selection of ONT long-reads, still required assigning them to one of two categories: 'gene/transcript' and 'no-gene/transcript'. The first referred to such reads that could potentially belong to a particular genome fragment. The second referred to the opposite case. In order to accomplish the task posed in this way it was decided to use an alignment software, minimap2 [11, 12], which is responsible for aligning the given sequences to a given reference sequence. Using it, finding the likely location of each read becomes possible. Since the need to divide into two groups, the sought-after gene was chosen as the reference sequence. Ultimately, the majority of long-reads (7,128,352; 99.48%) were not mapped. All the rest of reads (36,914), according to the aligner, could potentially come from the gene being searched for.

Classifier structure

The structure of the system intended for classification, consisted of two main components. The first was responsible for appropriately encoding the input sequence and the second for performing the output prediction.

The proposed concept employs a well-known and widely used technique from natural language processing, the 'bag of words' [22]. It is based on counting the occurring keywords, belonging to a finite set (dictionary), which allows the input information to be represented as a vector of natural numbers. The algorithm works successfully with 'classic' text containing words, i.e. letter combinations separated by spaces. However, in the DNA/RNA data, there is a challenge in defining the 'word' within strings of nucleotides. To address this, the sequence is split into k-mers. Such words have a fixed length and are formed with a predetermined step, allowing long strings of nucleotides to be interpreted as an NLP problem. The encoded sequences as vectors were subjected to prediction, determining the probability of origin from a chosen gene.

Table 1 A number of reads and read length quartiles for a dataset I

Sample	Not filtered	Filtered (> 500 bp)			
	Long-reads	Long-reads	Q1	Median	Q3
A1	1,232,364	1,040,293	784	1231	1692
B1	1,665,661	1,410,171	780	1204	1702
C1	1,254,760	1,069,828	802	1263	1720
A2	1,670,162	1,356,675	745	1146	1556
B2	1,859,656	1,570,158	786	1229	1721
C2	862,354	718,141	772	1209	1648
Total	8,544,957	7,165,266			

To perform this classification, we employed a neural network, which is a computational model inspired by the structure and function of the human brain. Neural networks consist of layers of interconnected 'neurons' that process input data and adjust their connections based on the information they learn, allowing them to recognize patterns and make predictions. The specific neural network architecture used in this study was a fully connected model. The input layer received the encoded sequence vectors, with the size depending on the dictionary. The subsequent two hidden layers each consisted of 50 neurons, utilizing the ReLU activation function to introduce non-linearity and capture complex patterns within the data. Finally, the output layer, consisting of a single neuron with a sigmoid activation function, was responsible for determining the class membership, indicating whether the sequence belonged to one of the two predefined classes.

While we chose a neural network for its ability to capture complex relationships in the data, it is important to note that other classifiers could also be applied to this problem. The choice can range from simpler algorithms such as random forests to more complex architectures such as recurrent networks and transformers, depending on specific requirements and constraints. Similarly, the encoding component can be implemented in various ways. In this work, we also tested the embedding layer of a neural network, instead of a 'bag of words'. However, the concept of word construction has not changed.

Classifier's degrees of freedom

The implementation of the presented encoding process requires the setting of certain parameters, which impacts the final classification quality. One example of a value potentially influencing its efficiency and, at the same time, a necessary parameter from the point of view in generating words (k-mers) is their length (κ). Additionally, having a long sequence of nucleotides, subsequent vocables can be generated with a certain step ϕ, meaning an offset from the beginning of the previous one. Apart from that, the finite dictionary needed for encoding may contain components representing single words or also their neighborhood/context (λ). It is a case of a combination of several consecutive words representing a context, which would, in such a case, be a single dictionary component. Another parameter needed to consider is the impact of the encoded long-read length τ. Perhaps for some cases, analyzing only the initial τ nucleotides is sufficient, making the calculation certainly faster.

The last aspect is the size and structure of a training dataset (Ω). In this paper, all such parameters are called degrees of freedom, and the significance of which has been analyzed in depth. Figure 1 presents a visualization of the analyzed degrees of freedom.

Experimental design

The main aim of the study is to construct an NLP-based classification system that allows the detection of sequences that could potentially originate from the gene being searched for. Several configurations of parameters were analyzed in the leave-one-sample-out-cross-validation (LOSOCV) schema. Firstly, the randomly chosen sequences from both categories (gene and no-gene transcripts, with a 1:3 ratio to emphasize the rarity of the first group) from each sample constituted six fixed testing datasets, one for each LOSOCV experiment.

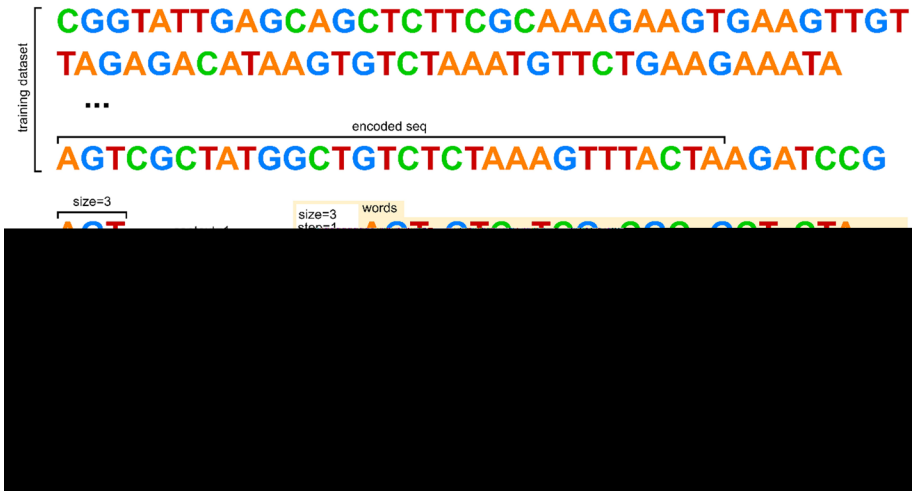

Fig. 1 Considered degrees of freedom

A preliminary configuration setting experiment focused on the subset of parameters. Reflecting the genetic code, the word length was initially assumed to mimic one codon and to be equal to 3 bp ($\kappa=3$). Next, to see which approach would perform better for the classification task, a comparison was made between a dictionary containing single words ($\lambda=1$) and a dictionary built from words with context, built with one 3 bp neighbor on each side ($\lambda=3$; 3 bp vs. 3 bp|3 bp|3 bp). For this experiment, step $\phi=1$ was used and all nucleotides of each sequence were encoded ($\tau=$whole seq). The first parameter investigated for its impact on the final prediction performance was the training set size Ω. Since the task of finding a sequence that would potentially be derived from a specific gene, it was decided to use an unbalanced dataset at the training stage. The criterion was based on biological reality, as the sequence sought represents only a small fraction of the entire genome. The assumed ratio of representatives of the two classes was 1:3, where the second referred to sequences other than the gene searched for (just like the test set). Based on this assumption, the following datasets were considered: $\boldsymbol{\Omega}=\{1{,}000+3{,}000;\ 3{,}000+9{,}000;\ 10{,}000+30{,}000;\sim 30{,}000+90{,}000;\sim 30{,}000+300{,}000$ and $\sim 30{,}000+900{,}000\}$. To clarify, $\sim 30{,}000$ should be understood as all available sequences derived from the gene, the exact number of which varies according to the cross-validation experiment. Therefore, the imbalance ratio in subsequent training sets increases. Additionally, in order to shorten the notation, '$\sim$' was omitted and the numeral k was used to denote a thousand.

Having the results from the preliminary experiment, it was decided to select such a training dataset Ω and neighborhood λ for which the classifier achieved the best performance, and then to carry out the main experiment, allowing the remaining degrees of freedom to be analyzed. The number of nucleotides undergoing encoding τ was taken as the whole sequence and the values representing measures of position—string length quartiles: Q1, median, Q3. Then, observing a constant difference between these values, it was decided to augment this set with a number even lower than Q1. In the end, five different values τ were obtained. Deciding on a two-element set of steps $\boldsymbol{\phi}\in\{1,2\}$ was supported by the occurrence of mutations and, in general, the appearance of read errors,

which is particularly characteristic of ONT long-reads. However, it is worth mentioning that as the parameter ϕ increases for $\lambda > 1$, the size of the final dictionary for the bag-of-words method rises. This is due to the higher number of permutations, which translates into the memory complexity of the hardware used to train the model. Therefore, among other reasons, it was decided to consider maximally two values ϕ. As the last degree of freedom determining the length of the k-mer, $\kappa \in \{3,6\}$was chosen. Again, the first value is related to the amino acid encoding and the second value is its doubling. Similar to the ϕ parameter, the size of κ directly impacts the cardinality of the final dictionary. By selecting one value λ and Ω, the final sets $\boldsymbol{\tau}$, $\boldsymbol{\phi}$, $\boldsymbol{\kappa}$, it was possible to conduct the main experiment and analyze the effectiveness of all classification models. Finally, 20 classifiers were compared with each other.

Testing on an external dataset

Based on the conclusions from the experiments, the optimal configuration of parameters for the classifier was selected. The goal was to identify values that would create an efficient classifier with the smallest possible dictionary size. After training the model on dataset I, it was evaluated on an external dataset II. Sequences from second dataset were taken, and it was checked whether the predictions made by the previously trained model matched the correct class assignments.

Dataset II differs significantly from the first; it involves genome sequencing rather than transcriptome sequencing, comes from a different source, and has distinct features. This variation provided a rigorous test for the generalizability of the classifier. Evaluating the model on such a diverse input helps determine its performance on new, unseen data, which is essential for assessing the effectiveness and robustness of the proposed solution.

Dictionary optimization

The size of the dictionary depends on the configuration parameters, and its range can include significant values depending on the choices made. Consequently, a key aspect becomes the optimization problem, which involves selecting the most relevant words in order to effectively manage the dictionary size. In the context of the case presented, certain nucleotide sequences (containing certain words) occur significantly more frequently in the analyzed gene compared to the rest of the genome. Moreover, they can be unique and only occur in one location, which is crucial in the classification task.

Taking these considerations into account, it was decided to examine the importance level (weight) of each dictionary component. The features were ranked in ascending order, from the least important to the most important ones – a component with the least weight was assigned the number one, while the most relevant dictionary component was placed at the end of the list with the highest possible rank. The weights (feature importance factors) were assigned based on three different approaches: odds ratio (OR) based, effect size based and using explainable artificial intelligence (XAI) tools. For the first, OR-based method, a 1/OR transformation was applied for OR values less than 1, which makes feature rank independent from its type of impact (risk or protection). The effect size-based method used Cramer's V [23]. In the third method, the SHAP tool [24] was used. Finally, three independent rankings of features were obtained, which were later collated together and their selection consistency compared.

The optimization task was completed by investigating the impact of reducing the dictionary size on the efficiency level of the models. In a first step, dominant words were selected, meaning those with the highest weights, and a reduced feature space was used to train the model. The selection of dominant words can be approached in several ways. The simplest one is based on choosing a fixed d, which determines the number of words with the highest weight. This approach is quite complex, namely the question arises what value of d should be chosen. To solve the problem, two different data-driven approaches, known from the analysis of Receiver Operating Characteristic (ROC) in machine learning optimization task, were used. The previously calculated word importance values were sorted and plotted to constitute the importance curve. In the first method, called 'max distance', the distance of each importance curve data point from a straight line connecting the first and last points was calculated. The data point (representing a particular component of the dictionary) with the maximum distance defined the cut-off. All words with an importance score higher than that were included in the reduced dictionary. The second approach uses piecewise linear regression to model the importance curve. The dividing points were determined by minimizing the residual sum of squares for the entire dataset. The cut-off point was the data point separating the first and second regression lines (see supplementary Figure S2).

Evaluation metrics

The developed approach requires tuning several parameters, which was done step by step. Firstly, we selected the values of λ and Ω (preliminary experiment) and then the remaining configuration parameters that determine the most effective classifier. Several indicators can evaluate a classifier, each focusing on a different property to determine the model's effectiveness. As the focus is on the sequence filtering task, the main objective is to separate all potentially possible gene sequences from the rest. Omitting a read that potentially contains the specific gene is expected to occur rarely. Therefore, the negative predictive value (NPV), defined as the fraction of true negatives among all negatives obtained, is chosen to be maximized. To determine the confidence that the reads classified into the group of the searched sequence actually contained it, we used positive predictive value (PPV, also known as precision), which represents the fraction of true positives among all positives. Additionally, balanced accuracy (BACC) was employed to evaluate the overall effectiveness of the proposed solution, particularly considering the imbalance between classes.

To provide a more comprehensive evaluation of the classifier's performance, we also included additional metrics such as sensitivity, specificity, and the F1 score, offering insights from various perspectives.

Results and discussion

As mentioned earlier, the whole study included two experiments: a preliminary and a main experiment. The first one allowed to select the training set size Ω and the neighborhood value λ, while the second one allowed to compare different NLP encoding configurations: τ, ϕ, κ.

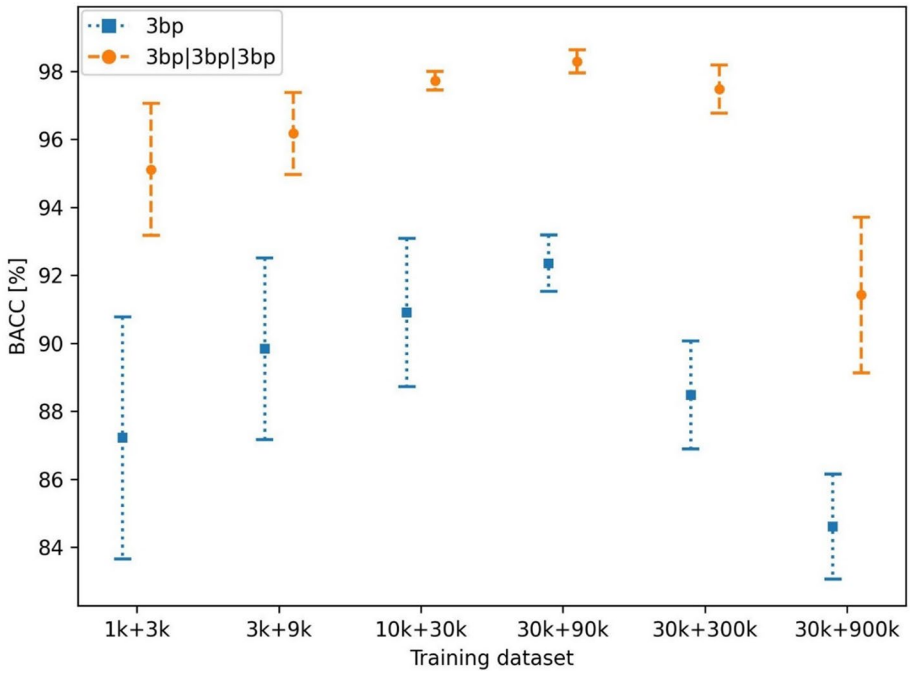

Fig. 2 Balanced accuracy (BACC) depending on training set size for 3 bp and 3 bp|3 bp|3 bp. Results are presented as mean values and its 95% confidence intervals. 3 bp is $\lambda = 1$ and 3 bp|3 bp|3 bp is $\lambda = 3$

Preliminary experiment

Figure 2 shows the balanced accuracy of the classifiers for which the training set size was altered.

The approach with single words ($\lambda = 1$) in the dictionary and one with two-sided neighbors ($\lambda = 3$) were compared to each other. In turn, the exact NPV and PPV values, together with the number of representatives in the final dictionary for the same sets, are summarized in Table 2. Assumed values for the remaining parameters are: $\tau = whole\ seq$, $\phi = 1$, $\kappa = 3$. The results clearly illustrate the effectiveness of the applied method for the classification task. Even for a relatively small dataset, the model is able to classify sequences with an average of 87.22% ($\lambda = 1$) and 95.11% ($\lambda = 3$) balanced accuracy. The most effective model expressed by this indicator was achieved for $\Omega = 30$ k + 90 k (all reads potentially derived from the sought-after gene), which had a very high value of 98.29% for words with the neighborhood, PPV = 96.58%, NPV = 99.25% and F1 score = 97.15%. The one on the same set $\boldsymbol{\Omega}$, also proved to train the classifier most effectively considering models with $\lambda = 1$, where BACC = 92.35%, PPV = 92.47%, NPV = 95.78% and F1 score = 89.66%. By directly comparing the models against the considered parameter λ, it can be concluded that the use of neighborhood words improves the effectiveness of the classifier by about 5–9% for balanced accuracy, 3–5% for NPV, 11–18% for sensitivity and 7–12% for F1 score. However, the nature of the changes depending on Ω is similar. Initially, there is a dynamic improvement in efficiency up to a saturation level, and then worse model evaluation rates are achieved. However, the advantage of the classifiers based on $\lambda = 1$ was the very small size of the final dictionary. Due to the four-nucleotide alphabet, the number of possible permutations of three-letter

Table 2 Mean values of evaluation metrics with their 95% confidence intervals (in brackets)

Dataset size	Indicator	3 bp [%]	3 bp|3 bp|3 bp [%]
1 k + 3 k	NPV	93.84 (90.11, 97.56)	97.44 (95.89, 98.99)
	PPV	84.80 (71.22, 98.38)	94.31 (90.53, 98.10)
	BACC	87.22 (83.66, 90.78)	95.11 (93.17, 97.05)
	Sensitivity	80.55 (67.55, 93.55)	92.16 (87.16, 97.16)
	Specificity	93.89 (87.26, 100.00)	98.06 (96.61, 99.51)
	F1 Score	81.03 (78.40, 83.65)	93.08 (91.50, 94.66)
3 k + 9 k	NPV	94.32 (92.51, 96.13)	97.98 (96.92, 99.05)
	PPV	91.46 (88.67, 94.25)	95.54 (92.24, 98.85)
	BACC	89.83 (87.16, 92.50)	96.17 (94.96, 97.38)
	Sensitivity	82.28 (76.08, 88.49)	93.86 (90.54, 97.19)
	Specificity	97.38 (96.32, 98.44)	98.48 (97.21, 99.76)
	F1 Score	86.46 (83.79, 89.13)	94.61 (93.50, 95.72)
10 k + 30 k	NPV	95.02 (93.29, 96.76)	98.95 (98.61, 99.30)
	PPV	91.63 (86.12, 97.14)	95.82 (94.05, 97.59)
	BACC	90.91 (88.72, 93.09)	97.72 (97.45, 97.99)
	Sensitivity	84.56 (78.55, 90.56)	96.86 (95.80, 97.91)
	Specificity	97.25 (95.23, 99.27)	98.58 (97.93, 99.23)
	F1 Score	87.68 (85.85, 89.50)	96.32 (95.84, 96.81)
30 k + 90 k	NPV	95.78 (95.10, 96.47)	99.25 (98.94, 99.55)
	PPV	92.47 (90.29, 94.66)	96.58 (95.72, 97.44)
	BACC	92.35 (91.53, 93.18)	98.29 (97.95, 98.63)
	Sensitivity	87.09 (84.80, 89.38)	97.74 (96.82, 98.67)
	Specificity	97.61 (96.84, 98.38)	98.84 (98.53, 99.15)
	F1 Score	89.66 (89.06, 90.27)	97.15 (96.88, 97.42)
30 k + 300 k	NPV	92.99 (92.07, 93.91)	98.62 (98.12, 99.11)
	PPV	97.96 (97.56, 98.36)	97.39 (96.32, 98.47)
	BACC	88.48 (86.89, 90.06)	97.48 (96.77, 98.18)
	Sensitivity	77.49 (74.30, 80.67)	95.82 (94.29, 97.34)
	Specificity	99.46 (99.35, 99.57)	99.14 (98.77, 99.51)
	F1 Score	86.50 (84.50, 88.51)	96.59 (95.84, 97.34)
30 k + 900 k	NPV	90.73 (89.88, 91.58)	94.65 (93.26, 96.04)
	PPV	99.39 (98.83, 99.96)	99.41 (99.12, 99.69)
	BACC	84.61 (83.06, 86.16)	91.42 (89.13, 93.71)
	Sensitivity	69.36 (66.25, 72.47)	83.01 (78.36, 87.65)
	Specificity	99.86 (99.72, 99.99)	99.83 (99.75, 99.92)
	F1 Score	81.67 (79.51, 83.84)	90.41 (87.75, 93.07)
		Dictionary size = 64	*Dictionary size = 1,024*

words was only 64. This was as much as sixteen times less than classifiers with a neighborhood of $\lambda = 3$.

Main experiment

After a preliminary experiment, the parameters $\Omega = 30\text{ k} + 90\text{ k}$, $\lambda = 3$ were chosen, and the next experiment focused on the impact of word length and step. The number of the first base pairs from the analyzed sequence was also parametrized. The obtained estimates of balanced accuracy are shown in Fig. 3.

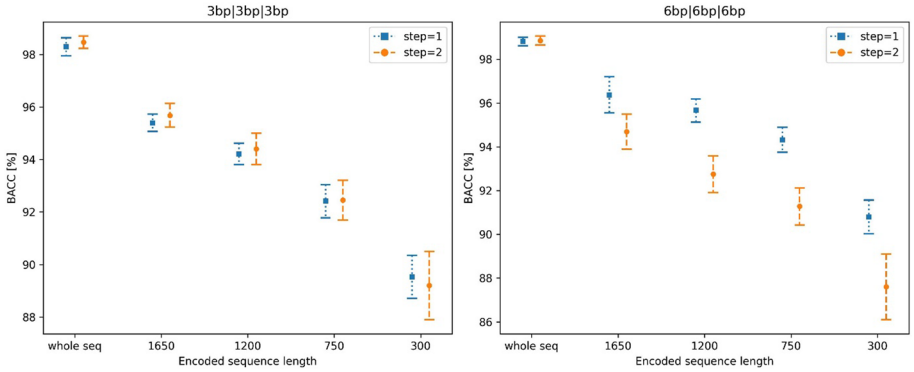

Fig. 3 Model balanced accuracy (BACC) depending on the encoded sequence length. Steps $\varphi \in \{1; 2\}$ and $\kappa = 3$ (left panel) and $\kappa = 6$ (right panel) were used. Results are presented as mean values and their 95% confidence interval

They compare models varying in the number of encoded nucleotides τ and the word generation step ϕ. In the left panel, the results for $\kappa = 3$ (3 bp|3 bp|3 bp) are accumulated, and in the right one for $\kappa = 6$ (6 bp|6 bp|6 bp). The NPVs, PPVs and BACCs are summarised in supplementary Table S1 and Table S2. Again, mean values and their 95% confidence interval based on all LOSOCV replicates are used.

As expected, the best performance was achieved by classifiers that encoded all nucleotides of long-reads. Regardless of ϕ and κ, their BACC was over 98%, PPV over 96%, NPV over 99% and specificity over 98%. Analyzing Fig. 3 (left panel) and Table S1, it can be deduced that for the 3 bp|3 bp|3 bp encoding there was no significant improvement in model efficiency depending on the step ϕ. However, there was a significant increase in the size of the final dictionary. A 16-fold larger list, translated into significantly higher memory complexity, but did not result in noticeable increases in classification. The situation was different for the 6 bp|6 bp|6 bp encoding (Fig. 3, right panel), where, comparing the models against the parameter ϕ, those with a lower step were characterized by a better balanced accuracy score for all tested values of τ other than whole seq. The same situation is observed for the NPV, sensitivity and F1 score indicators, shown in Table S2. It is also worth noting the significant size of the final dictionary for $\phi = 2$. The number of permutations was so large that only for such parameters configuration, not all possible combinations of 6 bp|6 bp|6 bp were included in the dictionary. Nevertheless, encoding all the nucleotides of long-reads in this way achieved the best classifier considering BACC, NPV, sensitivity and F1 score indicators. The worst among the $\tau =$ whole seq models was the one with 3 bp|3 bp|3 bp encoding for $\phi = 1$. However, the difference in the quality measures analyzed is marginal, especially bearing in mind the approximately 1017% smaller size of the final dictionary compared to the most effective of them.

Embedding neural layer

As mentioned earlier, the structure of the system consists of two main components: for encoding and prediction. Once a long sequence has been split into words, different approaches can be applied. Classification results using the embedded neural layer instead of a 'bag of words' method are shown in Table S3. It summarizes the selected encoding configurations: 3 bp, 3 bp|3 bp|3 bp and 6 bp|6 bp|6 bp.

Comparing the obtained results with those presented in Table 2 ($\Omega = 30$ k + 90 k) and Table S2 ($\tau =$ whole seq, $\phi = 1$), it can be observed that the 'bag of words' method generally provided better performance. Each evaluation metric showed higher (better) value, except for PPV, sensitivity, and F1 score in the 6 bp|6 bp|6 bp scenario. Nevertheless, as the dictionary size increased, there was a noticeable improvement in all evaluated metrics.

Testing on an external dataset

As mentioned before, performing the preliminary and main experiments made it possible to select the optimal values for the configuration parameters. Based on the results presented, it was decided that the classifier tested on the external dataset should have the following properties: neighborhood $\lambda = 3$, word length $\kappa = 3$ and step $\phi = 1$. $\Omega = 30$ k + 90 k was chosen as the training set and $\tau =$ *whole seq* was encoded. This configuration provided a 1024 components dictionary, with high model efficiency. Running a verification test on an external dataset II provided feedback for which the BACC = 75.28%, PPV = 66.96%, NPV = 99.64%, sensitivity = 99.82%, specificity = 50.74% and F1 score = 80.15%. These results were based on 58,574 long sequences, with 29,287 representatives in each class.

It can be seen that, despite a completely different dataset, characterized by dissimilar properties from dataset I, the results confirmed the effectiveness of the proposed solution. Noteworthy is the very high value of the NPV metric, which is especially important from the point of view of the considered task and further analysis of sequences potentially derived from the searched gene. Thus, it seems that the 3 bp|3 bp|3 bp encoding, with dictionary size = 1024, fulfils the task of filtering out long-reads that with high confidence do not originate from the sought-after gene.

Dictionary optimization

Using exactly the same configuration parameters, a dictionary containing 1024 components was analyzed. It began by creating three independent rankings, which were then compared. The rank value represented the feature importance – the higher ranking value the more important component ('word') is.

First, each component was assigned a position in all rankings, and next the average position was calculated, which was used to order the dictionaries. In this way, supplementary Figure S3 and Figure S4 were created, focusing on the components with the highest ranks. The green line indicates the perfect match of all three methods.

Table 3 Dictionary size after reducing its components to dominant ones for each LOSOCV repetition

Selection strategy	Max distance						Regression-based					
	A1	B1	C1	A2	B2	C2	A1	B1	C1	A2	B2	C2
Odds ratio	149	156	145	170	173	169	143	139	143	140	135	142
Effect size	117	118	118	110	114	120	88	93	88	93	95	87
XAI	74	53	77	73	52	91	44	44	52	55	44	41

The next step was to select the most dominant features treated as gene (transcript) fingerprints. Using the max distance and regression approaches, cut-off points were determined, i.e. the number of dictionary components with the highest significance level. The sizes of the reduced dictionaries are presented in Table 3. Subsequently, the consistency in selection of such dictionaries was compared. First against all ranking methods per each LOSOCV (supplementary Figure S5 for exemplary LOSOCV) and then across all LOSOCV repetitions (Table 4).

The analysis was concluded by comparing how the dictionary restriction affects the final performance of the classifiers. During this step, the focus was on comparing the 95% confidence intervals for NPV, PPV, BACC. The obtained results are presented in Fig. 4.

Based on the results achieved, there is a consistency in the selection of the most significant dictionary components. Regardless of the chosen ranking method, such words needed for encoding are outstanding and turn out to be of clear relevance during model prediction. This is shown in supplementary Figure S3 and Figure S4. At first, for the initial dictionary components, there is quite a spread of differences in ranking positions, but as one approaches the words with the highest importance, the spread decreases and the points tend towards the line of perfect match.

In the case of methods for selecting dominant features, it can be seen (Table 3) that the regression approach generally selects fewer dictionary components, and the difference from max distance depends on the nature of the plot (related to the way the weights were calculated). The least noticeable difference is for the OR-based method. However, it is important to say that the components in the reduced dictionaries are repeated. The phenomenon occurs across the LOSOCV repetition (supplementary Figure S5) and in the ranking methods (Table 4). The XAI-based approach achieves the lowest values of the Dice similarity coefficient, but it is worth noting that the least numerous dictionaries of dominant features are also observed. The other two approaches observe high coverage of selected components.

Importantly, the NPV metric decreases relatively slightly (Fig. 4), despite reducing the dictionary components by up to 20 times (XAI & regression). For the OR-based method, the average NPV dropped from 99.25% (all components) to 98.15% in the max-distance-based optimized dictionary and 97.89% in the regression-based. For effect size, averages of 97.24% (max distance) and 96.70% (regression) were achieved. Similarly, NPV fell to

Table 4 Consistency in selecting dominant dictionary components across LOSOCV repetitions in relation to the first repetition

LOSOCV	Max distance			Regression-based		
	OR	Effect size	XAI	OR	Effect size	XAI
A1–reference	1.0000 (149)	1.0000 (117)	1.0000 (74)	1.0000 (143)	1.0000 (88)	1.0000 (44)
B1	0.9574 (146)	0.9702 (114)	0.6457 (41)	0.9574 (135)	0.9724 (88)	0.6591 (29)
C1	0.9864 (145)	0.9872 (116)	0.6623 (50)	0.9860 (141)	0.9886 (87)	0.7292 (35)
A2	0.9342 (149)	0.9604 (109)	0.6667 (49)	0.9894 (140)	0.9613 (87)	0.6263 (31)
B2	0.9255 (149)	0.9524 (110)	0.6349 (40)	0.9712 (135)	0.9508 (87)	0.6136 (27)
C2	0.9371 (149)	0.9873 (117)	0.6424 (53)	0.9895 (141)	0.9943 (87)	0.7059 (30)

The Dice similarity coefficient is shown, along with the number of common dictionary components (in brackets)

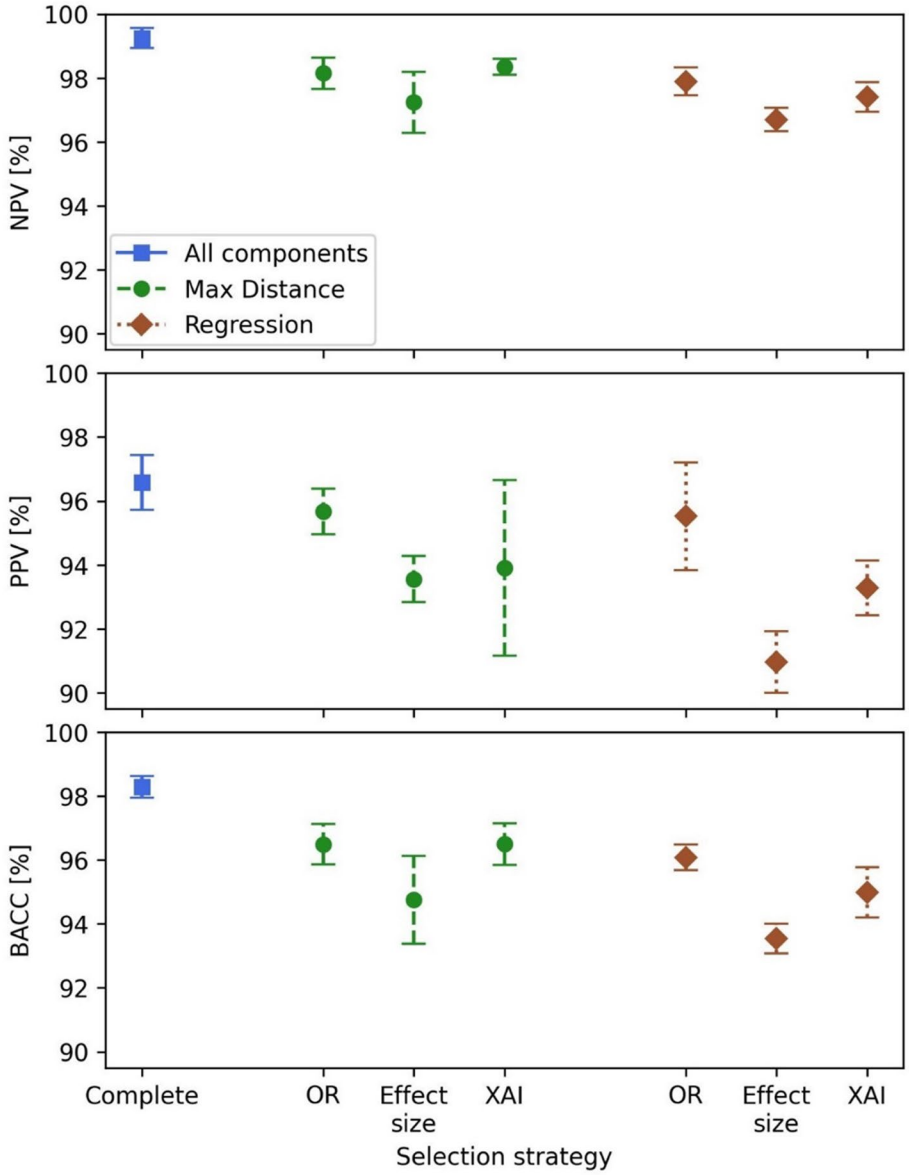

Fig. 4 The summarized classifier's performance in LOSOCV obtained for different dictionary optimization strategies. Results are presented as mean value and its 95% confidence interval

98.35% and 97.41% for the XAI-based method. Such a property allows increased efficiency during the encoding stage and reduced model prediction time. The worst performance was achieved by classifiers with dominant features selected using effect size. The OR-based technique performed best, but it should be noted the classifiers were trained on the largest feature space (reduced dictionary size: 135–173). PPV values generally reached lower levels compared to NPV, but a similar nature of change can be observed. The highest PPV = 96.58% reached the complete dictionary. The max distance method resulted in decreases to 95.67% (OR), 93.55% (effect size) and 93.91% (XAI), while the regression technique led to 95.52% (OR), 90.96% (effect size) and 93.28% (XAI). It is noteworthy that in all cases, except the effect size & max distance combination, a wider

confidence interval is observed in comparison to NPV. In the last indicator also the greatest BACC = 98.29% was observed for the complete dictionary. Methods based on OR, effect size and XAI achieved: 96.49%, 94.76%, 96.50% for max distance and 96.08%, 93.54%, 94.99% for regression.

The final evaluation of the developed no-mapping sequence aligner used the gene expression values as indicator. All sequences classified as gene/transcript related were subjected to transcript counts per million estimation by *salmon* software [25]. NCBI database was chosen for gene transcript references. The results are presented in Table 5. Additionally, to verify the effectiveness of the proposed solution, the same approach was performed for the NACA gene, characterised by higher expression. The results are included in Table S4.

One can notice that the estimation error, as related to the classical approach, varies from 1.30% to 5.97% (mean 3.48%, standard deviation 1.56%) for FDXR gene expression for complete dictionary NLP model and from 2.25% to 9.86% (mean 5.82%, standard

Table 5 Standardized FDXR transcript/gene count estimates (per million) for different data processing models

Transcript	A1	A2	B1	B2	C1	C2
Classical approach: minimap2 + salmon						
NR_047576.3	0	24.31	15.29	0	0	0
NM_001258014.4	13.79	0	0	0	0	0
NM_024417.5	13.79	156.46	15.47	110.16	28.29	59.03
NM_004110.6	13.79	0	15.47	91.08	28.29	92.59
NM_001258015.3	0	0	0	0	0	0
NM_001258012.4	0	4.24	0	0	0	21.50
NM_001258013.4	0	0	0	24.07	0	33.28
NM_001258016.3	0	0	0	0	0	0
FDXR total	**41.38**	**185.01**	**46.23**	**225.31**	**56.58**	**206.41**
noMapper (complete dictionary) + salmon						
NR_047576.3	0	25.87	15.10	0	0	0
NM_001258014.4	13.25	0	0	0	0	0
NM_024417.5	13.25	149.63	15.26	103.66	27.50	56.52
NM_004110.6	13.25	0	15.26	84.88	27.50	89.19
NM_001258015.3	0	0	0	0	0	0
NM_001258012.4	0	4.12	0	0	0	20.60
NM_001258013.4	0	0	0	23.33	0	31.98
NM_001258016.3	0	0	0	0	0	0
FDXR total	**39.76**	**179.62**	**45.63**	**211.87**	**54.99**	**198.29**
noMapper (reduced dictionary) + salmon						
NR_047576.3	0	24.75	14.71	0	0	0
NM_001258014.4	13.25	0	0	0	0	0
NM_024417.5	13.25	143.14	14.86	101.81	25.50	57.60
NM_004110.6	13.25	0	14.86	82.90	25.50	90.65
NM_001258015.3	0	0	0	0	0	0
NM_001258012.4	0	3.95	0	0	0	20.98558
NM_001258013.4	0	0	0	22.85	0	32.54
NM_001258016.3	0	0	0	0	0	0
FDXR total	**39.76**	**171.84**	**44.43**	**207.57**	**51.01**	**201.77**

deviation 2.91%) for the model using the reduced dictionary. In the case of the NACA gene, the estimation error is 0.55–2.02% (complete dictionary, mean 1.28%, standard deviation 0.59%). Keeping in mind that the targeted approach is the radiation accident victim triage, the obtained accuracy fulfills the system requirements.

Hardware performance

To evaluate the computational efficiency of noMapper, performance tests were conducted on a Raspberry Pi 5. The single-board computer is equipped with a 2.4 GHz quad-core ARM Cortex-A76 CPU, 8 GB RAM, making it a suitable platform for assessing the tool's practicality in resource-constrained environments.

The encoding stage is identified as the most time-consuming part of the process. The final computation time depends on the dictionary size, so the optimization described plays an important role. Our noMapper processed 8,000 Nanopore long-reads in about 15 s for dictionary containing 1024 components.

Limitations

Despite the promising results achieved with our tool, limitations must be acknowledged to provide readers with a comprehensive understanding of its current form. The necessity for target-specific system preparation must be enlisted firstly. To effectively utilize the tool for a particular marker/sequence/gene, users must first train the noMapper model using machine learning algorithms tailored to the intended target. This process requires leveraging sequence alignment software to generate the necessary training data. Although our repository provides a detailed guide and a complete pipeline for conducting this preparation, simplifying the process, it does require some additional effort from the user. Once the training is complete, the noMapper no longer depends on alignment software for operation. It can be then deployed on the target hardware, such as a Raspberry Pi, but this initial customisation step is essential for effective tool application.

Although initial training data generated from an alignment tool is required for noMapper, it is crucial to emphasise that in certain tasks, a portable, reliable, energy-efficient and computationally cost-effective solution for estimating the selected transcript expression is of paramount importance. The aforementioned example pertains to the selection of individuals who may have been exposed to irradiation, as may occur in the event of a nuclear power plant incident. In such cases, a number of genes are responsible for the formation of what is known as the irradiation signature. In the context of the ongoing miniaturisation of sequencing devices, such as the Oxford Nanopore MinION sequencer, which enables rapid, real-time long read sequencing of nucleic acids, the development of a bioinformatics tool capable of simultaneous targeted mapping of reads is of uppermost importance. NoMapper is designed to meet this need. Once the system has undergone the requisite training, it can be integrated with the sequencer and dose predicitive model and then used as a fast portable patient triage supporting device. Moreover, irradiation dose estimation is not the only area of noMapper usage. The system can also be used to identify diseases/pathogens in the event of an outbreak as, after initial training, it allows rapid analysis of the transcriptome to assess viral/bacterial response. Similarly, in the case of bioterrorism, we can cite as an example the estimation of the expression of transcriptomic biomarkers of the presence of botulinum toxins or assessment of food

biocontamination. To mitigate this alignment-based training limitation, we propose the creation of a repository of pre-trained noMapper models specialized for various targets. By developing a web-based platform to host these models, users could easily access and utilize noMapper instances that are ready for immediate application, reducing the time and effort required for customization.

In addition, when discussing the limitations of noMapper, we will emphasize that the encoding step affects the overall prediction time, with the duration of this process primarily depending on the dictionary size used. In our work, we have explored the optimisation of dictionary size and its impact on system efficiency. However, it is important to note that larger dictionaries tend to lengthen the encoding process. This trade-off between speed and performance is an inherent limitation that users must consider, especially when using the tool in time-sensitive scenarios.

Conclusions

This paper presents a method for classifying long-reads in search of a specific transcript sequence(s). The proposed solution consisted of two main components: the first was responsible for encoding the sequence using NLP methods, and the second was a neural network for performing the sequence classification. Various parameters were analyzed to encode the set of long-reads, as well as construct the training dataset. A total of 31 machine-learning models were considered. The best-performing classifier used the training dataset with 1:3 ratio between possible gene and no-gene categories. The prepared comparison unequivocally showed the advantage of encoding taking into account neighbors over a dictionary containing only single words with a length (κ) of 3 bp. Based on the results obtained, there is a marginal to small effect of the step ϕ parameter on NPV, PPV and BACC. As the parameters κ and ϕ rise, the size of the final dictionary increases significantly. During the entire work carried out, the classifier systems ranged from 64 to approximately 1,041,599 encoding elements. This translates into memory and time complexity. Speeding up the encoding process can be done by choosing an suitable value for the parameter τ. As the obtained results showed, when the first 1650 initial nucleotides (equivalent to Q3) were encoded, the decrease in quality of the BACC metric was 2–4% and the NPV 2–3% relative to τ = whole seq. For 750 initial nucleotides (Q1 equivalent), the quality drops were 4–7% and 3–5%, respectively.

The finally selected classifier system with the configuration of the parameters: $\Omega = 30\text{ k} + 90\text{ k}$, $\lambda = 3$, $\kappa = 3$, $\phi = 1$ and τ = whole seq, was tested on an external genome sequencing dataset and the obtained results confirmed the effectiveness of the proposed solution.

Further investigations of classifiers with a dictionary equal to 1024 showed the potential for optimization. The results obtained clearly indicate that some features have a greater or lesser influence on the final prediction of the model. Regardless of the method used to calculate the weights, it is possible to distinguish the components which rank the most influential positions in the rankings. By reducing the dictionaries to only the key ranking places, the effectiveness of the classifiers decreases by 1–3% for NPV, gaining 6–25 × smaller dictionary size, depending on the approach used to calculate the weights and locate the cut-off point.

The analysis results presented in this paper show the potential of applying techniques known from NLP to the field of bioinformatics. Appropriate processing of long strings of nucleotides, allows the reads to be treated as 'classic' text, consisting of single words. The demonstrated solution thus provides an alternative to the classical alignment tool. By narrowing down the task to the search for a specific sequence, we can bypass the mapping process and at the same time apply the shown machine learning based method. The proposed noMapper can be easily used to identify sequences of interest. High efficiency results have proven the point of transforming DNA/RNA data into a form friendly to NLP techniques and make advancements in this branch of science.

Abbreviations

ATCC	American type culture collection
BACC	Balanced accuracy
bp	Base pairs
ECDF	Empirical cumulative distribution function
FBS	Fetal bovine serum
LOSOCV	Leave-one-sample-out-cross-validation
MEM	Minimum essential medium
NGS	Next generation sequencing
NLP	Natural language processing
NPV	Negative predictive value
ONT	Oxford nanopore technologies
OR	Odds ratio
PPV	Positive predictive value
ROC	Receiver operating characteristic
TGS	Third generation sequencing
XAI	Explainable artificial intelligence

Supplementary Information

The online version contains supplementary material available at https://doi.org/10.1186/s12859-024-05980-7.

Additional file1 (DOCX 66 KB)

Additional file2 (DOCX 159 KB)

Additional file3 (DOCX 739 KB)

Additional file4 (DOCX 402 KB)

Additional file5 (DOCX 288 KB)

Additional file6 (DOCX 18 KB)

Additional file7 (DOCX 18 KB)

Additional file8 (DOCX 17 KB)

Additional file9 (DOCX 19 KB)

Author contributions

TS and JP designed the algorithm concept. TS implemented the scripts, performed the tests and all computational analysis. TS and JP wrote the manuscript. JP supervised research and critically revised the work. LCG, MN and CB were involved in the data acquisition, carried out the experiments in the laboratory, performed the sequencing. LCG and MN conducted the first interpretation of the data. CB arranged funding, planned the project and participated in research design. All authors contributed to the discussion, participated in interpretation of the results, read and approved the final manuscript.

Funding

Partially financially supported by the United Kingdom Health Security Agency, CRCE Capital Programme, grant no CRCE/CAP/20–21/406, IDEA: bIological DosimEtry Assessment device. TS was partially financially supported by the Silesian University of Technology grant for Support and Development of Research Potential (02/070/BKM23/0050).

Availability of data and materials

The source code is available at https://github.com/ZAEDPolSl/noMapper. The dataset I used and analyzed during the current study is accessed at the Sequence Read Archive with accession number PRJNA1131758 https://www.ncbi.nlm.nih.gov/bioproject/PRJNA1131758. The original dataset II is publicly available at https://labs.epi2me.io/gm24385_2020.11/.

Declarations

Competing interests
The authors declare that they have no competing interests.

Ethical approval and consent to participate
Not applicable.

Consent for publication
Not applicable.

Received: 21 June 2024 Accepted: 6 November 2024
Published online: 13 November 2024

References

1. Collins FS. The human genome project: lessons from large-scale biology. Science. 2003;300(5617):286–90.
2. Heather JM, Chain B. The Sequence of sequencers: the History of Sequencing DNA. Genomics. 2016;107(1):1–8.
3. Sanger F, Nicklen S, Coulson AR. DNA sequencing with chain-terminating inhibitors. Proc Natl Acad Sci. 1977;74(12):5463–7.
4. Maxam AM, Gilbert W. A new method for sequencing DNA. Proc Natl Acad Sci. 1977;74(2):560–4.
5. Bentley DR, Balasubramanian S, Swerdlow HP, Smith GP, Milton J, Brown CG, et al. Accurate whole human genome sequencing using reversible terminator chemistry. Nature. 2008;456(7218):53–9.
6. Branton D, Deamer DW, Marziali A, Bayley H, Benner SA, Butler T, et al. The potential and challenges of nanopore sequencing. Nat Biotechnol. 2008;26(10):1146–53.
7. Kchouk M, Gibrat JF, Elloumi M. Generations of Sequencing Technologies: From First to Next Generation. Biology and Medicine. 2017;09(03).
8. Wang Y, Zhao Y, Bollas A, Wang Y, Au KF. Nanopore sequencing technology, bioinformatics and applications. Nat Biotechnol. 2021;39(11):1348–65.
9. Needleman SB, Wunsch CD. A general method applicable to the search for similarities in the amino acid sequence of two proteins. J Mol Biol. 1970;48(3):443–53.
10. Smith TF, Waterman MS. Identification of common molecular subsequences. J Mol Biol. 1981;147(1):195–7.
11. Li H. New strategies to improve minimap2 alignment accuracy. Bioinformatics. 2021;37(23):4572–4.
12. Li H. Minimap2: pairwise alignment for nucleotide sequences. Bioinformatics. 2018;34(18):3094–100.
13. Kabacik S, Mackay A, Tamber N, Manning G, Finnon P, Paillier F, et al. Gene expression following ionising radiation: identification of biomarkers for dose estimation and prediction of individual response. Int J Radiat Biol. 2010;87(2):115–29.
14. Alshayeji MH, Sindhu SC, Abed S. Viral genome prediction from raw human DNA sequence samples by combining natural language processing and machine learning techniques. Expert Syst Appl. 2023;218: 119641.
15. Cong Y, Chan Yb, Ragan MA. A novel alignment-free method for detection of lateral genetic transfer based on TF-IDF. Scientif Report. 2016. https://doi.org/10.1038/srep30308.
16. Strzoda T, Cruz-Garcia L, Najim M, Badie C, Polańska J. (2023) No-mapping mapping of Oxford Nanopore long reads. In: Recent advances in computational oncology and personalized medicine Vol 3, Crossing borders, connecting science. Politechnika Śląska,. pp. 60–70
17. Strzoda T, Cruz-Garcia L, Najim M, Badie C, Polańska J. How to encode in no-mapping mapping of Oxford Nanopore long reads? In: PTBI Symposium 2023 Step into the world of science, September 13–15, Gliwice, Poland
18. Strzoda T, Cruz-Garcia L, Badie C, Polańska J. How far can we go with sequence shortening in NLP-based mapping of transcriptome data derived from Oxford Nanopore sequencing technology? In: Computational oncology and personalized medicine—Crossing borders, connecting science COPM2023, Gliwice, April 26th, 2023
19. Cruz-Garcia L, O'Brien G, Sipos B, Mayes S, Tichý A, Sirák I, et al. In vivo validation of alternative FDXR transcripts in human blood in response to ionizing radiation. Int J Mol Sci. 2020;21(21):7851.
20. November 2020 GM24385 Dataset Release [Internet]. EPI2ME Labs. 2020 [cited 2024 Jun 6]. Available from: https://labs.epi2me.io/gm24385_2020.11/
21. Wick R. rrwick/Porechop [Internet]. GitHub. 2024 [cited 2024 Apr 7]. Available from: https://github.com/rrwick/Porechop/
22. Qader W, Ameen M, Ahmed B. (2019) An overview of bag of words;importance, implementation, applications, and challenges
23. Cramér H. Mathematical methods of statistics. Princeton: Princeton University Press; 1999.
24. Lundberg SM, Lee SI. A Unified approach to interpreting model predictions. Vol. 30, Neural information processing systems. curran associates, Inc.; 2017.
25. Patro R, Duggal G, Love MI, Irizarry RA, Kingsford C. Salmon provides fast and bias-aware quantification of transcript expression. Nat Methods. 2017;14(4):417–9.

Publisher's Note