Expert Systems With Applications 258 (2024) 125126

Contents lists available at ScienceDirect

Expert Systems With Applications

journal homepage: www.elsevier.com/locate/eswa

AI-driven transcriptomic encoders: From explainable models to accurate, sample-independent cancer diagnostics

Danilo Croce [a,*,1], Artem Smirnov [b,c,1], Luigi Tiburzi [a,1], Serena Travaglini [b,1], Roberta Costa [a,1], Armando Calabrese [a,1], Roberto Basili [a,1], Nathan Levialdi Ghiron [a,1], Gerry Melino [b,1]

[a] Department of Enterprise Engineering, University of Rome "Tor Vergata", via del Politecnico 1, 00133 Rome, Italy
[b] Department of Experimental Medicine, University of Rome "Tor Vergata", via Montpellier 1, 00133 Rome, Italy
[c] Biochemistry Laboratory, Istituto Dermopatico Immacolata (IDI-IRCCS), 00166 Rome, Italy

ARTICLE INFO

Keywords:
Transcriptomic AI encoders
Explainable cancer prediction
Sample-independent diagnostic models

ABSTRACT

In the rapidly evolving domain of medical technology, the utilization of sophisticated algorithms for deciphering transcriptional data has emerged as a critical aspect, especially in the oncology sector. These algorithms, drawing upon methodologies from fields such as natural language processing and advanced image analysis, can significantly enhance the accuracy in predicting cancer-related molecular states. Notably, Transformer models, renowned for their proficiency in handling extensive datasets, are now being adapted for breakthroughs in medical diagnostics or in stratifying patients according to prognostic levels. Our study contributes to the field of precision medicine by integrating Transformer-based learning, exemplified by the Geneformer model, with explainable AI techniques. These techniques are employed to find out the input variables (genes resulting from genomic transcription) most correlated with the decisions of neural network systems. This insight, a key goal in genomic research, aims to select the most relevant gene subset for each specific task in which a neural network is employed. This selection approach has proven to be effective in two classification tasks: cell type classification and breast cancer type classification. Such effectiveness has been demonstrated even across various cohorts of patients. When applying Geneformer-like architecture analyses solely to the selected gene subsets, the outcomes either maintain their accuracy or significantly improve. This approach, aims not only to contribute to the identification of vital genetic markers in cancer genomics, but also to exemplify the adaptability of AI models to different datasets, marking a significant step towards the development of accurate and universally applicable diagnostic tools for precision medicine.

1. Introduction

The journey to unravel the mysteries of genetic information, tracing back to Watson and Crick's discovery of the structure of DNA (Watson & Crick, 1953), has propelled significant advancements in genomic research. Initiatives like the Human Genome Project (Lander et al., 2001) and subsequent endeavors such as FANTOM (Kawai et al., 2001), ENCODE (Consortium et al., 2012) and Roadmap Epigenomics (Roadmap Epigenomics Consortium et al., 2015) have vastly expanded our understanding of the human genome, epigenome, and transcriptome, as in Vitale et al. (2023). This expansion fueled the transition from genetics, which focuses on individual genes, to genomics, a discipline encompassing the study of all genes of an organism and their interactions with the environment. Multi-omics research, now a data-intensive field, leverages high-throughput sequencing technologies, making it possible to analyze complete DNA sequences rapidly. These technologies, globally accessible, have led to a substantial increase in genomic and transcriptomic data, available through databases such as GDC, dbGaP, and GEO, discussed in, just to cite a few, Amelio, et al. (2020, 2020), Ganini et al. (2021) and Stephens et al. (2015). This abundance of data, combined with advanced statistical methods, has opened new avenues for exploring genomic and gene expression information, paving the way for innovative research in identifying and understanding genetic elements and their functions.

In the rapidly evolving development of artificial intelligence (AI) methods in medicine, the exploitation of techniques for the analysis

[*] Corresponding author.
E-mail addresses: croce@info.uniroma2.it (D. Croce), artem.smirnov@uniroma2.it (A. Smirnov), luigi.tiburzi@uniroma2.it (L. Tiburzi), serena.travaglini@uniroma2.it (S. Travaglini), roberta.costa@uniroma2.it (R. Costa), calabrese@dii.uniroma2.it (A. Calabrese), basili@info.uniroma2.it (R. Basili), levialdi@dii.uniroma2.it (N. Levialdi Ghiron), melino@uniroma2.it (G. Melino).
[1] All authors contributed equally to the publication.

https://doi.org/10.1016/j.eswa.2024.125126
Received 14 February 2024; Received in revised form 24 July 2024; Accepted 14 August 2024
Available online 22 August 2024

of such data represents the forefront of innovation. This is particularly evident in oncology, where AI methodologies are increasingly employed to predict cellular and tissue states linked to cancer (Kumar, Gupta, Singla, & Hu, 2022). The integration of advanced neural learning techniques, particularly those derived from fields such as natural language processing (NLP) and computer vision, has marked a significant advancement in these predictive processes. Transformer-based models (Lin, Wang, Liu, & Qiu, 2022; Vaswani et al., 2017), renowned for their efficiency in handling large volumes of data, are now being adapted for the complex and nuanced field of medical diagnostics (Yue et al., 2023). Their capability to exploit extensive transcriptomic datasets for pre-training neural models has proved invaluable, particularly in applications characterized by few annotated data. In particular, a novel Transformer-based architecture specifically developed for the analysis of transcriptomic data within medical contexts, Theodoris, Xiao, Chopra, Chaffin, Sayed, Hill, Mantineo, Brydon, Zeng, Liu, and Ellinor (2023), is the subject of this study. Starting from the most recent advancements in deep learning methodologies (Yue et al., 2023), this research investigates how Geneformer, together with explainable AI techniques (XAI), can contribute to genomic research.

In fact, despite substantial advancements in genomic research, significant challenges persist in efficiently and accurately decoding the vast quantities of genetic data. Studies, such as Stephens et al. (2015), have highlighted limitations in existing methodologies, particularly in their ability to handle the complexity and heterogeneity of transcriptomic data. The resulting models are difficult to interpret and seem quite ineffective against the datasets on which they are trained. This backdrop underscores the need for innovative approaches that not only address the breadth and complexity of genetic data, but also provide deeper, easier to interpret, insights into the results obtained.

Specifically, the study focuses on the role played by AI in representing gene information within medical or biological prediction tasks. As a matter of fact, a complex interplay of genes plays a fundamental role in biological phenomena. As a result, identifying those most informative genes for learning a model of such phenomena, such as diagnosing distinct cancer classes, is crucial for understanding the whole genetic dynamics involved, e.g., how cancer development and progression can be justified and signaled at genetic level. Models induced via Machine Learning (ML) are trained on known phenomena and tend to select the meaningful signals (also called *features*) that correlate with the target phenomena. The models bet on the correspondence between correlation and causal relations. In such a framework, important features for a highly accurate induced model should have a higher probability of showing biological and clinical correlations or causal relations.

Due to the above reasoning, it is clear why a pivotal element of this study is the application of XAI methods to high-performance computational models (Došilović, Brčić, & Hlupić, 2018). This approach is grounded in the rationale that if a model's decision is accurate, then the genes selected for that decision are likely to be those associated with the analyzed phenomenon. In other words, if a system can accurately classify cells into specific cancer types, it is fundamental to learn the genes most affecting the outcome as they are likely to be somehow involved in the process. This challenge is a cornerstone in transcriptomics, the aim being to devise neural methods that, while capable of observing roughly 20,000 genes, can concentrate on a potentially small, yet significant subsets of them. Identifying these subsets could be immensely beneficial, for instance, in defining "customized" markers for types of inferences and diseases. This is particularly effective given that current Transformer-based architectures like Geneformer can handle gene sequences up to 2048 symbols, an order of magnitude less than the existing 20,000.

In this work, we suggest that explainable AI methods are used to pre-analyze gene sequences, thereby restricting Geneformer-like analysis to a focused and significant subset of relevant genes. This strategy is advantageous as it exploits a model that, though simpler, can effectively identify which input variables (gene transcripts, hereafter "genes") have most contributed to the model's final decision. The challenge lies in verifying the accuracy of the selected gene subset. Our hypothesis is that if a Geneformer-like architecture is applied to a truly relevant subset of genes, its performance should either remain consistent or improve.

In detail, we have employed the proposed methodology to select the most pertinent genes in two distinct tasks. The first is *cell type recognition*, which we have approached as a classification of genes into 9 different organs via Transformer-based neural architectures. The results of this task are exceptionally promising. We observed that by focusing the analysis on a few hundred informative genes, rather than the over 20,000 known functional genes, the predictive capability of the model is either maintained or even enhanced. The second and more intriguing task focused on *breast cancer type classification*. The implementation of our proposed techniques here significantly improves the model's quality. More importantly, it allows for the application of the model across diverse patient cohorts, substantially amplifying the method's applicability. This research not only aims to contribute to the identification of vital genetic markers in cancer genomics but also to exemplify the adaptability of AI models to different datasets to mark a significant step in the development of accurate and universally applicable diagnostic tools.

We focus on two key areas that contribute to the field of medical AI in genetic analysis for cancer diagnostics:

1. We explore how explainable AI can effectively identify the most relevant genes in an inductive process. This targeted approach, when combined with a recent transformer-based method, has shown to be effective. Our research highlights the potential of using explainable AI to refine gene expression analysis, making it more precise and relevant for specific diagnostic tasks.
2. We introduce a new method for input representation that reduces the reliance on the specific techniques used to gather and encode data. This involves a rank-based approach to represent gene expressiveness, focusing on relative expression levels. This methodology has enabled models to achieve improved accuracy levels by exploiting far fewer genes than previous methods in two tasks: cell type classification and cancer type discrimination. It is worth noticing that the model also shows good generalization abilities, as, when trained on a Caucasian patient cohort, it is also effective over Korean patients: this is a good indication of its good generalization and broader applicability. Our approach aims to make AI models in genomics more adaptable and applicable across different populations.

In the remainder of the paper, Section 2 discusses the related work. Section 3 details the proposed methodology, Section 4 outlines the experimental evaluation, and Section 5 provides the conclusions.

2. Related works

In the past decade, the contribution of Machine Learning and Artificial Intelligence to the medical field has been growing in intensity and significance (Van der Laak, Litjens, & Ciompi, 2021; Momeni, Hassanzadeh, Saniee Abadeh, & Bellazzi, 2020; Sung et al., 2021). Many new works have been published experimenting with different strands of both fields (Osama, Shaban, & Ali, 2023). Particularly relevant, and successful, have been the applications of Machine Learning (ML) and Artificial Intelligence (AI) to tasks of visual or genetic data classification for diagnostic purposes (Akhavan & Hasheminejad, 2023; Miguel, Neves, Martins, do Nascimento, & Tosta, 2023; Osama et al., 2023; Zhou, Chen, Yu, Pang, Cong, & Cong, 2024). The latter is the focus of this study.

Regarding genetic and transcriptomic data for diagnostic purposes, a traditional ML pipeline – i.e., data acquisition, data exploration, data preprocessing, dimensionality reduction, machine learning algorithm, evaluation – is used in Osama et al. (2023) to predict an outcome based on several features. The authors have reviewed different algorithms to effectively select the smallest subset of genes for an accurate classification and have developed a taxonomy including five classes of feature selection algorithms and two of feature extraction. Although the logic from data acquisition to evaluation is the same across the classes, they differ in their approach to the selection of the relevant features and the choice of the classification routine. All these techniques overcome the curse of dimensionality and the curse of data sparsity (Osama et al., 2023). Moreover, they are generally easier to interpret than more sophisticated approaches (e.g., deep learning techniques). Major limitations of these approaches include that data reduction is a property of the data set, the inability to take advantage of the links among the genes, and a tendency towards redundancy and overfitting (Khan & Lee, 2023; Osama et al., 2023).

Deep learning techniques, including artificial neural networks, have also been applied in several other works with a similar approach (Aziz, Verma, & Srivastava, 2017, 2018; Vanitha, Devaraj, & Venkatesulu, 2015). A number of features in input, such as genetic data, have been used to predict some outcomes, as for example cancer subtypes (Hassan Zadeh, Alsabi, Ramirez-Vick, & Nosoudi, 2020). Even this conventional approach has shown some limitations, notably a bias towards overfitting (Khan & Lee, 2023). In this case too, feature selection has to be performed before the deep network can be employed because the number of symbols (genes) that can be handled by a neural network is limited (Khan & Lee, 2023).

Approaches based on transformers, namely dynamic representation of features from self-attention-based architectures, have been recently suggested to overcome such limitations. Given their success in various natural language processing (NLP) tasks (Dosovitskiy et al., 2020), some works have tried to combine neural networks and self-attention mechanisms to other domains (Bello, Zoph, Vaswani, Shlens, & Le, 2019; Carion et al., 2020). In the medical field, transformers have been used mainly with visual data (Esteva et al., 2021). For example, Zhou et al. (2024) developed an end-to-end weakly supervised framework for classifying cancer subtypes based on histopathological slides. In their work, the authors adopt the self-attention mechanism to produce slide-level features for slide-level supervision, aiming to increase the number of usable patch-level labels from experts. Xin et al. (2022) developed a vision transformer, *SkinTrans*, to classify images of skin cancer. Their framework results in an accuracy across multiple databases of over 94%.

Some researchers have recently been attracted to the combination of transformers and explainable AI to genetic data, which is the application presented in this work. Due to the novelty of the topic, contributions are still few (Khan & Lee, 2023). Among the most relevant, at least for this research, are Theodoris et al. (2023), Khan and Lee (2023), and Rajpal et al. (2023). Theodoris et al. (2023) exploits a set of pre-trained embeddings on a large sample of genes (almost 30M) that can be fine-tuned to the solution genetic classification tasks. Khan and Lee (2023) and Rajpal et al. (2023), on the other hand, are two novel frameworks based on self-attention mechanisms to identify relevant biomarkers in various cancer subtypes.

Compared to previous attempts, this work is innovative for several reasons. First, it does not use conventional ML techniques in the classification step as these have been shown to be inefficient and potentially biased towards overfitting (Rajpal et al., 2023). In particular, this work develops a two-stage metaheuristic algorithm combining explainable Artificial Intelligence (XAI) and Deep Learning (DL) techniques, as formalized in Yaqoob, Verma, and Aziz (2024), Yaqoob, Verma, Aziz, and Saxena (2024). In fact, as discussed in Yaqoob, Verma, Aziz, and Saxena (2024), a common hindrance in extracting relevant features from genetic data is the presence, in our genome, of a large amount of redundant information. Hence, the importance of the filtering step prior to any classification attempt. Yet, rather than developing algorithms anew (Akhavan & Hasheminejad, 2023; Saxena, Chouhan, Aziz, & Agarwal, 2024; Yaqoob, Verma, Aziz, & Saxena, 2024), our methodology makes use of already available building blocks, namely linear Support Vector Machine (SVM) and the Geneformer model from Theodoris et al. (2023). Moreover, and this is the second novel element in this work, compared to Theodoris et al. (2023), in our case a simple XAI technique – SVM – is used in the feature selection step in the pipeline leading to the final prediction. Other metaheuristics, such as Akhavan and Hasheminejad (2023), Saxena et al. (2024) and Yaqoob, Verma, Aziz, and Saxena (2024), have instead combined sophisticated learning algorithms to enhance the accuracy of their predictions at the expense of explainability. In our case, the complex interactions between genes have been taken into account thanks to the Theodoris et al. (2023)'s embeddings, while the task-specific feature selection has been performed with an explainable technique. This choice has the advantage of making it easier to identify important features behind a classification problem, which we believe is highly relevant from a transcriptomic research perspective.

In summary, given the limitation of current attempts (Akhavan & Hasheminejad, 2023; Khan & Lee, 2023; Rajpal et al., 2023; Saxena et al., 2024; Yaqoob, Verma, Aziz, & Saxena, 2024), we rely on a transformer-based approach that learns optimal embeddings in the first step (Theodoris et al., 2023). These are used for representing genes as well as their context (i.e., interactions). Then, an explainable AI technique, SVM, is used to reduce the number of features (i.e., genes) exploited for representing instances, so that the final inference is improved. More details are given in the next section. As suggested by the experimental evaluation reported in this work, the proposed approach has a significant impact in terms of performance and range of applicability. In fact, combining a linear classification technique, SVM, with *Geneformer* improves its overall performance. On the other hand, using general, i.e. organ or cell or task-independent, embeddings to perform the classification allows the application of the algorithm to heterogeneous cohorts without any significant loss of accuracy.

3. Enhancing geneformer with salient genes

In this section, we detail our approach to enhancing the Geneformer model by integrating explainable AI techniques for selecting the most informative genes. Our methodology aimed to improve both the interpretability and the performance of the Geneformer model by focusing on a refined subset of genes, which emerge as crucial for specific classification tasks. The process involves several key steps, from the initial input of gene sequences to the training of the Geneformer model on filtered sequences. The diagram in Fig. 1 summarizes the overall workflow.

The process starts with the input dataset containing gene sequences. These sequences are normalized and ranked based on their expression levels to ensure that the data is in a suitable format for subsequent analysis. A linear Support Vector Machine (SVM) is then applied to the normalized data to identify the most informative genes, which are crucial for the classification tasks. From these identified genes, the top-k genes for each class are selected. These top-k genes are then used to construct filtered gene sequences, which serve as input for training the Geneformer model. The final output of this process is the trained Geneformer model along with the set of most relevant genes, for the task at hand.

By integrating these steps, our methodology enhances the Geneformer model's ability to make accurate and interpretable predictions. This approach not only improves the efficiency of gene sequence classification but also aids in uncovering causal connections between the system's inferences and underlying genetic factors. This is particularly significant in the field of genomics, where understanding the relationships between genetic components and observed phenotypic traits or diseases is crucial.

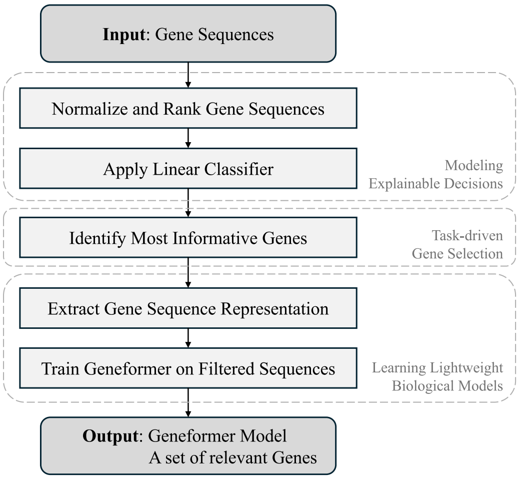

Input: Gene Sequences
Normalize and Rank Gene Sequences
Apply Linear Classifier
Modeling Explainable Decisions
Identify Most Informative Genes
Task-driven Gene Selection
Extract Gene Sequence Representation
Train Geneformer on Filtered Sequences
Learning Lightweight Biological Models
Output: Geneformer Model
A set of relevant Genes

Fig. 1. Process flow diagram of our proposed methodology.

In the rest of this section, Section 3.1 discusses BERT (Bidirectional Encoder Representations from Transformers), emphasizing its impact on NLP tasks through pre-training and fine-tuning. Section 3.2 examines Geneformer, an adaptation of BERT for genomics, highlighting the processing of genetic sequences. Lastly, Section 3.3 introduces a task-specific gene-selection method using linear classifiers, particularly Support Vector Machines (SVMs), to identify the most informative genes for classification tasks.

3.1. BERT: Bidirectional encoder representations from transformers

In the field of computer vision, researchers have repeatedly shown the beneficial contribution of transfer learning. Transfer learning is the ability to use data available for a given task T1 as a useful source of information to solve a different task T2. The common procedure is to *pre-train* the neural network on data available for T1, and then over the resulting pre-initialized model on T1 start training the target system, as a *fine-tuning* stage, on data available for T2. For example, the pre-training of convolutional neural networks on the ImageNet dataset is commonly used to support the later fine-tuning stage of the resulting pre-trained network to obtain a new optimized object detection task-specific model, e.g., Girshick, Donahue, Darrell, and Malik (2013).

The approach proposed in Devlin, Chang, Lee, and Toutanova (2019), namely, Bidirectional Encoder Representations from Transformers (BERT) provides a very effective model to pre-train a deep and complex neural network over a very large-scale corpus of unannotated texts. In this approach, after the network has been pre-trained, it can be applied to a large variety of NLP task by simply fine-tuning the entire architecture to each new problem. The building block of BERT is the *Transformer* element, an attention-based mechanism that learns contextual relations between words (or sub-words, i.e. word pieces, Schuster & Nakajima, 2012) in a text. In its original form, proposed in Vaswani et al. (2017), the Transformer includes two separate mechanisms: an encoder that reads the text input and a decoder that produces a prediction for targeted Machine Translation tasks.

In addition to the Transformer architecture's core capabilities, a crucial aspect of BERT's input handling is the incorporation of positional embeddings. When BERT processes a sequence of symbols, such as words or word-pieces, each symbol is coupled with a positional embedding. These embeddings are vectors derived through sinusoidal functions, designed to encode the position of each word within a sentence. For example, consider the sentence "*the dog is running in the garden.*" In this sentence, the positional embedding ensures that the representation of the first occurrence of the word "*the*" is different from its representation when it appears in the sixth position. This differentiation is achieved through the unique positional embeddings that are assigned to each word based on their location in the sequence. The significance of these positional embeddings lies in their ability to capture and represent the contextual and syntactic information that is inherently tied to the order of words in a sentence. By encoding the position of each word, BERT can understand the role and relationship of each one in the context of the entire sentence sequence.

A neural architecture such as BERT is inherently complex, comprising over 110 million parameters. The key to the success of such an architecture lies in the concept of pre-training. This involves initializing the network's weights through pre-training tasks that, while potentially unrelated to the network's eventual primary task, are linguistic and therefore help the model to generalize its understanding of language use.[2] These tasks are thus carried out by applying the network to extensive document collections, often consisting of billions of tokens. Such large-scale exposure to diverse linguistic patterns enables BERT to develop a deep and nuanced understanding of language. It is akin to how humans learn language: by being exposed to a wide range of sentences, words, and their corresponding contexts.

As shown in Fig. 2 (on the left), *during pre-training* the Transformer encoder reads the entire sequence of words at once and acquires a language model by learning to reconstruct the original sentence applying an MLM (*masked language model*) pre-training objective: the MLM randomly masks some of the tokens from the input, and the objective is to predict the original masked word based only on its context. In addition to the masked language model, BERT also uses a *next sentence prediction* task that jointly pre-trains text-pair representations. This last objective is crucial to improve the network capability of modeling relational information between text pairs, which is particularly important in tasks such as Dialogue Modeling or Question Answering (Devlin et al., 2019) to relate an answer to a question.

After the language model has been trained over a generic document collection, the BERT architecture allows encoding (i) specific target words belonging to a sentence, (ii) the entire target sentence, or (iii) sentence pairs with dedicated embeddings. These can be used as input for further deep architectures to solve sentence classification, sequence labeling or relational learning tasks by simply adding layers and fine-tuning the entire architecture (Bouraoui, Camacho-Collados, & Schockaert, 2020). In detail, on top of the pre-trained embeddings, *fine-tuning* is applied by adding task-specific layers. In a nutshell, these layers introduce a minimal number of additional task-specific parameters that are used to train the extended network on the targeted tasks. This additional training is a simple fine-tuning of all pre-trained parameters to optimize the performance of the network on the problem at hand.

BERT's encoding for classification in genomics. Consider BERT as a model $h(s) = M_{BERT}(s)$ that takes an input sequence s and generates a vector representation. In the context of this work, our primary interest is in classification tasks. For a given input sequence s, BERT can be viewed as generating a vector from the first symbol of the sequence. In BERT's architecture, this first symbol is the artificial token `[CLS]`, and $h(s)$ is a dense vector of d dimensions (e.g., $d = 768$). To utilize BERT for classification tasks, the output $h(s)$ of BERT, a 768-dimensional vector, is processed through a classifier that maps this high-dimensional vector to a space of c dimensions, where c is the number of classes. The output of this classifier, y, is a one-hot vector

[2] According to Wittgenstein (1953), language meaning arises as a side-effect of its use by native speakers. Language use is thus the crucial source of information about syntactic and lexical semantics phenomena in natural language. Pre-training in transformers aims at capturing exactly such universal properties of natural languages *before* attempting the training aimed at specific linguistic inferences (e.g., machine translation or question answering).

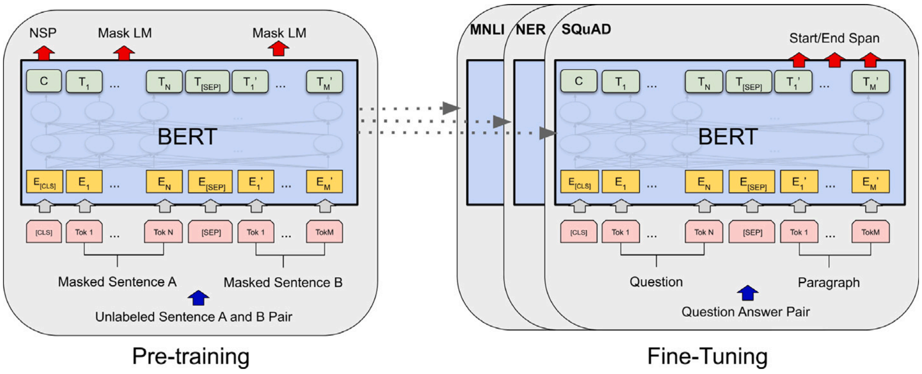


Fig. 2. Overall pre-training and fine-tuning procedures for BERT. Apart from output layers, the same architectures are used in both pre-training and fine-tuning. The same pre-trained model parameters are used as initial model for different downstream tasks. During fine-tuning, all parameters are fine-tuned, i.e., optimized for target tasks. [CLS] is a special symbol added in front of every input example, and [SEP] is a special separator token (e.g., separating questions/answers).

representing the class labels. Mathematically, this classification process can be expressed as:

$$y = \mathrm{C} \cdot M_{BERT}(s)$$

where $M_{BERT}(s)$ is the BERT model applied to the input sequence s, and C is a weight matrix in $\mathbb{R}^{d \times c}$, with d being the dimensionality of the BERT output. The classifier is typically trained using a cross-entropy loss function, defined as:

$$\text{Loss} = -\sum_{i=1}^{c} y_i \log(p_i)$$

Here, y_i is the true class label in one-hot encoded form, and p_i is the predicted probability of each class. This loss function optimizes the model parameters, ensuring the accurate classification of input sequences. The straightforward application of BERT has shown better results than previous state-of-the-art models on a wide spectrum of natural language processing tasks (Lin et al., 2022).

3.2. Geneformer: applying the BERT model in genomics

Building upon the systematic approach of pre-training transformer-based models as demonstrated in approaches such as BERT, the Geneformer architecture introduces a novel approach in genomics (Theodoris et al., 2023). Moving from the success of BERT in processing vast collections of unannotated texts, Geneformer represents a significant stride forward in the field of genetic sequence analysis. This architecture diverges from traditional text processing by focusing on gene symbols, such as EIF1, CLCA1, EEF1A1, MUC2, TFF3, and ITLN1, instead of words or word-pieces. Thus, to apply such architecture, we need to adapt input gene sequences that are initially provided as a list of pairs (g_i, e_i), where:

- $g_i \in \mathfrak{G}$ is a gene, with $\mathfrak{G}$ representing the dictionary of possible genes, such as EIF1 or CLCA1, or possible expression patterns (e.g., splicing isoforms, alternative promoters, post-translational modifications).
- $e_i \in \mathbb{R}$ is a real number describing the expressiveness of gene g_i, e.g., measured in terms of TPM (transcripts per million), FPKM (fragments per kilobase of transcript per million fragments mapped), and normalized counts using coefficient of variation, intraclass correlation coefficient, and cluster analysis (Zhao et al., 2021).

Clearly, the development of Geneformer is hindered by several key challenges. Training the model effectively is one of these challenges as it requires careful consideration of the adopted pre-training strategy as well as the selection of appropriate data. The model must be exposed to a wide range of genetic information to ensure robust learning and effective performance. Basically, the main idea behind Geneformer is to pre-train an encoder-based architecture using a sequence of genes expressed as: $S = \{(g_1, e_1), (g_2, e_2), \dots, (g_n, e_n)\}$ where g_i is the ith gene in the sequence and e_i is the corresponding expressiveness value. Geneformer's pre-training involves utilizing Genecorpus-30M (Theodoris et al., 2023), an extensive dataset comprising 30 million cell transcriptions, which enables the integration and analysis of data from 561 publicly available datasets. During this pre-training phase, an approach similar to BERT's masked language model is employed, in which 15% of genes are masked and then predicted by the architecture. This process allows Geneformer to learn contextual relations between genes, similarly to how BERT learns meaningful patterns as relations between words.

The representation of genes sequences is another crucial aspect. The model needs to accurately interpret both the order and the expression levels of each gene within a set. This understanding is necessary for the model to analyze expression data correctly and make precise predictions. Traditionally, transcriptomic sequences S consist of genes paired with a numerical value indicating their expressiveness. Geneformer addresses this challenge by ranking genes based on their expression, thus the most expressed genes are placed at the beginning of the sequence. More formally, we define a total ordering on S based on the expression values such that (g_i, e_i) precedes (g_j, e_j) in S if and only if $e_i > e_j$. Thus, S is ordered in descending order of expression values. This ranking allows the model to consider the most representative genes first, leveraging their context-dependent information.

However, an additional critical issue arises, namely the presence of a large set of expressed housekeeping genes. These are not candidate genes to provide relevant information to model task-specific phenomena, yet to a token-processing network, they look like important bits of information. To address this, Geneformer incorporates a normalization step that utilizes the entire Genecorpus-30M dataset. Thus, for each gene, the system relies on the non-zero median expression evaluated on the entire Genecorpus-30M dataset, which serves as the normalization factor. This normalization ensures that a gene with a high normalized expression input into Geneformer is one whose original observed mean expression is significantly higher than the overall average. Moreover, since all genes undergo the same normalization process, this step is uniformly applicable. Consequently, genes with high differential expression, but average values much higher than the norm, will not appear unusually significant when analyzed by Geneformer. Conversely, genes with moderate but significantly higher than average activation levels will receive more attention in the analysis. This approach balances the data, ensuring that Geneformer focuses on genes that show truly distinctive expression patterns.

For example, the original sequence in transcriptomics might initially appear as: $S^O = \{(\mathtt{MUC2}, 1.3), (\mathtt{TFF3}, 0.3), (\mathtt{EIF1}, 2.0), (\mathtt{CLCA1}, 1.9), (\mathtt{EEF1A1} : 0.8) \dots\}$ which is then ranked according to expressiveness, resulting in a sequence like $S^R = \{(\mathtt{EIF1}, 2.0), (\mathtt{CLCA1}, 1.9), (\mathtt{MUC2}, 1.3), (\mathtt{EEF1A1}, 0.8), (\mathtt{TFF3}, 0.3) \dots\}$

In this rearranged sequence, the gene EIF1 appears in the first position due to its high expression value. However, it is important to note that EIF1 is a housekeeping gene involved in the initiation of protein synthesis (Fletcher, Pestova, Hellen, & Wagner, 1999). While crucial for cellular function, its high expression is not necessarily indicative of specific disease states, as it is a gene routinely active in various cellular processes. After normalization (N) and re-ranking (R), the sequence might be transformed to:

$S^{NR} = \{(\mathtt{CLCA1}, 2.37), (\mathtt{MUC2}, 1.3), (\mathtt{TFF3}, 1.2), (\mathtt{EIF1}, 1.1), (\mathtt{EEF1A1}, 0.88) \dots\}$

These resulting genes can be provided to the Geneformer. Each gene symbol is assigned to an embedding, refined from the pre-training phase, and extended with positional embedding, allowing the model to track the gene's position in the ranking. As a result, being in the top positions implies not only high expression at a local cellular level but also significant expressiveness compared to the average observed in that gene across a 30-million-cell collection. This means that whenever a gene is in the top positions for a given phenomenon (e.g., cells of a given tissue), it can be considered highly informative.

Another noteworthy aspect is that this representation method uniquely disregards the local expression values of genes within individual cells. Local expression levels can be quite specific to the measurement method or the device used for gene expression detection. The methodology utilizes first the absolute expressiveness values of genes for ranking their relevance, and then selects the most informative ones, neglecting the others as well as all the expressiveness values. This approach effectively sidesteps potential biases and inconsistencies that might arise from varying measurement techniques, ensuring a more reliable and universal representation of gene expressiveness in the context of large-scale genomic data analysis.

However, despite all their pros, methods like Geneformer, and transformers in general, are still affected by computational complexity. Traditional transformers have a computational complexity for estimating attention that is quadratic in terms of the length of the observed sequences. This means that doubling the sequence length quadruples the processing time, and tripling it makes it eight times slower. This complexity also affects the amount of memory needed for the computation process. In BERT, originally designed to handle sentences, the maximum sequence length is capped at 512 word pieces, sufficient for small sentences or paragraphs. In Geneformer, the decision has been made to handle sequences of up to 2048 symbols in length. While this length was not a problem during the pre-training steps with Genecorpus30M, where it could accommodate most of the sequences observed in the dataset, it is important to note that a full transcriptome may have up to 20,000 actively transcribed genes (features, symbols) – roughly an order of magnitude larger than those effectively observed by Geneformer. In scenarios where complete sequences are recovered, especially with advancing technology, it becomes crucial to overcome these limitations.

In the next section, we propose a classification method to address the previously mentioned limitations. This method focuses on selecting the most informative subset of genes for a given classification task involving a sequence of genes. We aim to identify a key group of genes, potentially up to 2048, that are most relevant to the analysis. This approach is particularly important in scenarios where complete gene sequences are extensive and computational resources are limited, ensuring a more targeted and efficient analysis.

An additional advantage of this approach is its alignment with the tenets of explainable AI. By focusing on the selection of the most representative genes, our method not only enhances the efficiency of gene sequence classification, but also aids in uncovering causal connections between the system's inferences and underlying causes, such as the presence of specific genes. This aspect is particularly significant in genomics, where understanding the causal relationships between genetic components and observed phenotypic traits or diseases is crucial.

3.3. Task-specific optimal gene-selection

Our research addresses the intricacy of analyzing genetic sequences by considering the gene set $\mathfrak{G}$, which comprises approximately 20,000 symbols, representing the total number of protein-coding genes in the human genome. We aim to refine this to the subset of the most k informative genes. Our goal is to compress the original space of possible 20,000 symbols by performing feature selection in order to single out the most representative dimensions. Unlike traditional dimensionality reduction methods such as SVD (Bishop, 2007), which might select dimensions based on variability, our approach leverages machine learning techniques tailored to a task at hand.

In particular, we propose a linear classifier, which aligns with the notion of PAC learnability (Vapnik, 1995), yet still achieves a commendable level of generalization. Specifically, we adopt a Support Vector Machine (SVM), a linear discriminative Machine Learning paradigm based on statistical learning theory. An SVM is effectively utilized for binary classification tasks, operating by constructing a hyperplane or set of hyperplanes in a high-dimensional space.

The classification function employed by the SVM is expressed as $f(x) = \text{sgn}(wx + b)$, where x represents the input sequences while the parameters w and b define the hyperplane used for categorizing each example into positive (+1) or negative (−1) classes. In our case, these are not gene sequences but points in a geometric space $\mathbb{R}^n$ with $n = 20,000$. This approach, while neglecting the interactions among dimensions, offers a clear view of each dimension's impact. Regarding the construction of feature vectors, there are two pathways: retaining activation values post-normalization or following the approach used by Geneformer, which involves transforming these values into boolean representations (one-hot encoding).

Despite its simplicity and potentially lower performance compared to more complex models like Geneformer, SVM offers a significant advantage in terms of interpretability. The classification function $f(x) = \text{sgn}(wx + b) = \text{sgn}\ \left(\sum_i w_i x_i + b\right)$ provides explicit insights into how individual genes influence the classification outcome, enhancing the model's transparency and understandability, as hereafter discussed. Moreover, the individual x_i values represent gene expression levels, where each x_i is set to zero if the corresponding gene is not transcribed, or otherwise holds a positive value. This positive value can be a binary 1, following a boolean approach, or a greater than zero expression level value, which by design is positive due to the nature of gene expression measurements.

Consequently, the weights w_i in the SVM play a pivotal role. In detail, they adhere to the following schema:

- Each dimension corresponds to the same gene across different instances.
- A positive weight ($w_i > 0$) is assigned to the ith gene that 'supports' a particular class, indicating a positive correlation between the gene's expression and the class. Conversely, negative weights ($w_i < 0$) are associated to those ith genes that are "not supportive" of the class, implying an inverse relationship.
- Weights close to zero ($w_i \approx 0$) are indicative of genes whose presence or absence is not significantly informative toward class determination.
- The magnitude of the weight ($|w_i|$) reflects the degree of support or opposition a gene offers to a class. A higher absolute value denotes a stronger influence, either supporting or contrasting, on class categorization.

This weighting system in the SVM model provides a nuanced view of how each gene contributes to the classification task. It not only identifies the relevant genes, but also quantifies their impact, allowing for a more comprehensive understanding of the underlying biological processes influencing the classification.

Following the analysis of gene expression and the corresponding SVM weights, an important aspect of our study involves limiting the later analysis to the most influential genes in the classification decision. To achieve this, we restrict our focus by selecting the top k dimensions of the weight vector w, which exhibit the highest absolute values. This approach aims to pinpoint the genes that are most significant in distinguishing between the classes, whether through strong positive or negative associations.

In detail, given the weight vector $w = (w_1, w_2, \dots, w_n)$, the process for selecting the top k dimensions is as follows:

1. Compute the absolute values of the weights: $W_{\text{abs}} = (|w_1|, |w_2|, \dots, |w_n|)$.
2. Sort the weights by their absolute values in descending order to obtain W_{sorted}.
3. Select the first k dimensions from W_{sorted}.

The resulting set of dimensions corresponds to the k genes that have the most significant impact on the SVM's classification decision across one or more datasets. This method allows us to effectively identify and focus on the genes that play a crucial role in the classification process, thus providing a clearer understanding of the underlying biological mechanisms.

Genes identified by selecting the top k weights of the SVM have a pivotal function in refining the input dictionary for Geneformer. By concentrating on these genes, we effectively narrow down the dictionary of symbols in the input to those most relevant for the classification task. This approach ensures that Geneformer focuses on the most impactful genetic elements, enhancing the model's efficiency and relevance to the specific biological context.

The advantage of this supervised selection method, as opposed to unsupervised approaches, is its direct correlation with the task and the dataset at hand. While unsupervised methods might identify a broad range of features, they do not necessarily prioritize these features based on their relevance for different classification tasks. In contrast, our SVM-based selection is inherently task-driven, ensuring that the dimensions we focus on are the most informative for the classification case at hand. This task-oriented approach to feature selection, not only streamlines Geneformer's input, but also aligns the model more closely with the specific objectives and nuances of the dataset, i.e., clinical phenomena, being analyzed.

The methodology we have formalized thus far is primarily focused on binary classifiers. However, it can be readily extended to multiclass classification scenarios involving c classes by employing the One-Versus-All (OVA) strategy for training c separate binary classifiers. In the OVA approach, for each class c_i, a dedicated binary classifier is defined. For this classifier, examples belonging to class c_i are treated as positive instances, while all other examples are considered negative. This results in c distinct binary classifiers, each specialized in distinguishing its corresponding class from all others. During the classification phase, an input example is evaluated by all c classifiers. The classifier yielding the highest value of the classification function $f(x)$ determines the class assignment for that example. This method ensures that each example is classified into the class for which it has the strongest positive association, as per the classifier's learned parameters.

In the context of multiclass classification using the One-Versus-All strategy, the gene selection process can be straightforwardly adapted. For each gene, we assign a weight w_i^* that maximizes its absolute value across all classes. This means that, instead of considering the weight of a gene in a single binary classifier, we evaluate its impact across all c classifiers, choosing the weight that demonstrates the strongest influence (either positive or negative) in any class.

More formally, given a set of classifiers $\{f_1(x), f_2(x), \dots, f_c(x)\}$ for c classes, and corresponding weight vectors $\{w^1, w^2, \dots, w^c\}$, the weight assigned to each gene i is determined by:

$$w_i^* = \max_{j=1,\dots,c} |w_i^j|$$

Here, w_i^j represents the weight of gene i in the classifier for class j, and w_i^* is the chosen weight for gene i across all classes, based on its maximal absolute value. This approach ensures that the most influential genes, considering their ranking across all possible classifications, are selected for further analysis with Geneformer.

This extension to multiclass scenarios allows our approach to maintain effectiveness and interpretability across a wider range of classification tasks. The OVA strategy provides a straightforward yet powerful means to adapt the binary classification framework to complex multiclass problems, retaining the core benefits of the SVM-based feature selection and its integration with Geneformer.

3.3.1. Scalable linear model for efficient feature selection

One of the primary limitations of algorithms such as Support Vector Machines (SVMs) is related to the optimization complexity in finding the optimal hyperplane. Specifically, the complexity of the optimization process in standard SVMs tends to grow almost quadratically with the number of examples (Platt, 1998). This can become a significant hindrance, especially when dealing with large datasets, as it directly impacts the computational efficiency and scalability of the model without adding overhead to Geneformer.

To address this additional complexity, we adopt the Dual Coordinate Descent (DCD), defined in Hsieh, Chang, Lin, Keerthi, and Sundararajan (2008), which is a batch and linear learning algorithm similar to the Support Vector Machine (SVM). Given d instances $s \in S$, their labels $y_i \in \pm 1$ and their corresponding $x_i \in \mathbb{R}^n$ counterparts, the DCD acquires the function $f : X \rightarrow R$ which minimizes the misclassification error by minimizing the probability that $y_i f(x_i) = y_i w x_i \leq 0$, just like a binary classification function.

The so-called primal formulation to determine w can be written as follows:

$$\underset{w \in \mathbb{R}^n}{\text{minimize}} \ \frac{1}{2}\|w\|^2 + C \sum_{i=1}^{d} \max\{0, 1 - y_i w^\top x_i\} \tag{1}$$

The above problem can be rewritten in its dual form

$$\begin{aligned} &\underset{\alpha}{\text{minimize}} \ D(\alpha) := \frac{1}{2}\alpha^\top Q \alpha - \alpha^\top \mathbb{1} \\ &\text{subject to } 0 \leq \alpha \leq C\mathbb{1} \end{aligned} \tag{2}$$

Here, Q is a $d \times d$ matrix whose entries are given by $Q_{ij} = y_i y_j x_i^\top x_j$, and $\mathbb{1}$ is the vector of all ones. The minimizer w^* of Eq. (1) and the minimizer α^* of Eq. (2) are related by the primal/dual connection: $w^* = \sum_{i=1}^{d} \alpha_i^* y_i x_i$. The dual problem in Eq. (2) is a Quadratic Program (QP) with box constraints, and the ith coordinate α_i corresponds to the ith instance (x_i, y_i).

According to Hsieh et al. (2008), the following coordinate descent scheme can be used to minimize Eq. (2):

- Initialize $\alpha^1 = (0, \dots, 0)$
- At iteration t select coordinate i_t
- Update α^t to α^{t+1} via

$$\begin{aligned} \alpha_{i_t}^{t+1} &= \underset{0 \leq \alpha_{i_t} \leq C}{\text{argmin}} \ D(\alpha^t + (\alpha_{i_t} - \alpha_{i_t}^t) e_{i_t}) \\ \alpha_i^{t+1} &= \alpha_i^t \ \forall i \neq i_t. \end{aligned} \tag{3}$$

Here, e_i denotes the ith standard basis vector. Since $D(\alpha)$ is a QP, the above problem can be solved exactly:

$$\alpha_{i_t}^{t+1} = \min\Big\{\max\{0, \alpha_{i_t}^t - \frac{\nabla_{i_t} D(\alpha^t)}{Q_{i_t} Q_{i_t}}\}, C\Big\}. \tag{4}$$

In the above equation, $\nabla_i D(\alpha)$ denotes the i_t-th coordinate of the gradient. The updates in each step are also closely related to implicit updates. If we maintain $w^t := \sum_i^d \alpha_i^t y_i x_i$, then the gradient $\nabla_{i_t} D(\alpha)$ can be computed efficiently using

$$\nabla_{i_t} D(\alpha) = e_{i_t}^\top (Q\alpha - 1) = w^t y_{i_t} x_{i_t} - 1 \tag{5}$$

and kept related to α^{t+1} by computing $w^{t+1} = w^t + (\alpha_i^{t+1} - \alpha_i^t) y_i x_i$. In each iteration, the entire dataset is used to optimize Eq. (1) and a practical choice is to access the examples randomly.

In Hsieh et al. (2008), the proposed method has been shown to reach an ϵ-accurate solution in $O\left(log(1/\epsilon)\right)$ iterations, so we can bound the number of iterations. As a result, we can fix a-priori the computation cost of the training time, still linear in terms of the number of training examples.

These linear SVM formulations are especially relevant in the context of genomic data, where the number of examples (gene sequences) can be exceedingly large. By utilizing a linear approach to SVM optimization, we can maintain the model's robustness and interpretability while significantly enhancing its applicability to larger datasets. This development represents an important stride in making SVMs more versatile and practical for a wider range of data-intensive applications. In practice, in the experimental evaluation, training a Geneformer architecture takes approximately 10 min using a dataset of 10,000 examples, while training the adopted linear classifier takes a few seconds on the same dataset.

Finally, to mitigate the potential bias in weight estimation inherent in SVMs, we employed L2 regularization in our experiments (Hsieh et al., 2008). The advantage of using L2 regularization is that it helps control the magnitude of the weights, distributing the importance across multiple features and reducing the risk of over-representation of significant weights. This approach aims to ensure that the selected genes are robust and biologically meaningful, enhancing the interpretability and reliability of our model.

4. Experimental evaluation

The experimental section of this study is designed with two primary goals in mind. First, we aim to evaluate the effectiveness of a combined SVM-Transformer approach in two distinct tasks: cell classification and tumor type classification. The second goal is to assess how effectively the gene selections made by SVM can be integrated with the modeling provided by Transformer. Through the whole experimentation, we seek to demonstrate the potential of this hybrid approach in enhancing the accuracy and relevance of gene selection in medical diagnostics.

4.1. Cell type classification

Task and Data. In an initial set of experiments, we endeavored to determine whether an explainable AI method based on SVM is capable of identifying the most relevant genes for the final decision in a classification task. A further objective was to ascertain whether Geneformer, when applied to filtered sequences that consider only the classification-relevant genes selected by the SVM, can compete with a Geneformer that analyses complete sequences. We hypothesize that if the explainable AI method has successfully identified the most representative genes, then a method like Geneformer should not experience a significant loss of information. On the contrary, it should be able to maintain the same scores even with a more drastic gene reduction.

Currently, Geneformer is capable of processing sequences up to 2048 symbols in length, each composed of genes from a complete dictionary of about 20,000 known functional genes. Our strategy focuses on assessing the Geneformer's performance when trained under the constraint of markedly shorter sets of genes. These sets of genes, although reduced in length, encapsulate a selection of genes that are highly pertinent to the task at hand. By deliberately limiting the training to a few hundred genes, we compel Geneformer to concentrate on a subset of genes that are most closely associated with the task, potentially enhancing its ability to extract meaningful patterns from a more concentrated genetic signal.

To empirically test these hypotheses, we structured the experiments as follows. We first used an SVM-based explainable AI method to select genes that were deemed most crucial for the classification task. We then created a filtered gene set based on this selection and trained Geneformer on these subsets. The performance of Geneformer on these reduced sets was compared to its performance on the full gene set to measure any potential loss of predictive accuracy.

Table 1
Dataset statistics for the cell type classification task. The train/test split is 80% and 20%, respectively.

Organ	Num. of classes	Num. of train examples	Num. of test examples	Avg. Seq. length
Brain	6	10,656	2,664	447
Immune	10	20,562	5,140	427
Kidney	15	35,199	8,800	561
Large Intestine	16	39,678	9,920	400
Liver	12	22,427	5,607	487
Lung	16	26,098	6,525	492
Pancreas	15	21,934	5,484	420
Placenta	3	7,415	1,854	603
Spleen	6	12,330	3,083	413
Average	*11*	*21,811*	*5,453*	*472*

First, we considered the task of cell classification. Given an organ and the cellular transcription of its cells, the task is to assign individual transcriptions to the corresponding cell type. For example, given a cell from the large intestine and its transcriptome profile, we need to determine whether the cell is one of 16 possible types such as `Enterocyte progenitor`, `Hepatocyte`, or a `B-cell`.

For this dataset, we relied on an example provided by the authors of Geneformer.[3] In this example, the authors demonstrate how Geneformer can be applied to the task of cell classification on cells belonging to 9 different organs. Specifically, examples are selected directly from Genecorpus-30M, and the distribution of the number of cell types (classes in the classification task), the number of examples in the training and test dataset, and the average length of sequences are reported in Table 1. The data provided are already encoded to be suitable for Geneformer; therefore, for the genes belonging to each cell, all genes are normalized using the Genecorpus-30M distribution and are locally ordered based on this value. It is also clear in this case that the average length of the sequences (here on average 472) is less than the maximum value that Geneformer can handle (2048).

Firstly, we evaluated the performance of Geneformer without any truncation or selection, alongside the "simple" assessment of SVM, which does not utilize contextual information provided by the Transformer-based architecture of Geneformer and the pre-training on the entire Genecorpus-30M. The analysis was replicated separately for each organ, meaning a cell could only be classified into the types anticipated for its respective organ.

Results and Discussion. The performance of the classification task has been measured using accuracy, defined as the percentage of test set examples correctly assigned to their class, and macro F1 score, calculated as the arithmetic mean of the F1 scores computed for each class. The latter metric is particularly relevant as it accounts for class imbalance.

The results are detailed in Table 2, where each row represents the outcomes corresponding to each organ, and the final row is the average of the performance statistics averaged across all organs. As a preliminary step, we established a baseline determined by the Most Frequent Class (MFC), where each test set example was assigned to the most prevalent class given the organ. The table results reveal how certain highly imbalanced classes, including organ examples such as `Brain`, show that 86% of examples belong to class `Erythroid progenitor cell` (and are thus correctly classified even by this naive classifier), rendering the accuracy potentially unrepresentative of

[3] https://huggingface.co/ctheodoris/Geneformer/blob/main/examples/cell_classification.ipynb

Table 2
Comparison of classification accuracy and F1 scores across models.

Organ	Accuracy			F1 Score		
	Baseline	SVM	Geneformer	Baseline	SVM	Geneformer
Brain	86.2%	97.1%	**97.6%**	28.5%	**80.6%**	80.4%
Immune	24.8%	92.6%	**94.4%**	18.0%	86.0%	**89.4%**
Kidney	29.4%	90.0%	**92.1%**	12.5%	84.7%	**87.1%**
Large Intes.	23.1%	89.0%	**92.5%**	11.7%	81.6%	**84.9%**
Liver	33.1%	83.1%	**91.2%**	15.3%	73.0%	**79.3%**
Lung	22.3%	92.7%	**93.4%**	11.7%	**86.5%**	84.1%
Pancreas	26.5%	90.0%	**93.8%**	12.5%	85.6%	**87.6%**
Placenta	74.1%	97.8%	**98.0%**	48.8%	96.6%	**96.8%**
Spleen	74.3%	98.6%	**99.0%**	28.4%	96.3%	**97.1%**
Average	*43.8%*	*92.3%*	***94.7%***	*20.8%*	*85.6%*	***87.4%***

true model performance. On the other hand, the F1 measure tends to be more resilient to class imbalance. In this case, it yields considerably lower results (on average 20.8% compared to 43.8% accuracy). This latter result provides a more nuanced view of the classifier's effectiveness in handling skewed data distributions.

The initial model considered in our experiments was a linear SVM. As said, we employed the Dual Coordinate Descent (DCD) algorithm implemented in the KELP library,[4] which operates directly in the primal observation space, and ensures that the model remains linear in its decision-making process. Each SVM model underwent optimization by fine-tuning the trade-off parameter C of the SVM on the dataset. The parameter C was selected from a range of values: $[10^{-3}, 10^{-2}, 0.1, 1, 10]$.

The final model we tested was Geneformer. We utilized the implementation and the pre-trained model made available in PyTorch through the Huggingface framework.[5] The parameters configured for Geneformer were as follows: a learning rate set to 5×10^{-4}, a training and evaluation batch size of 12, and the AdamW optimizer with a Linear scheduler that incrementally improved the learning rate for the first 10% of the training steps. We included a weight decay of 0.001. The models were trained over 20 epochs, selecting the models that maximized accuracy on the Development Set.

In general, the results showcased Geneformer's excellent performance. The transformer achieves approximately 94.7% accuracy and an impressive 87.4% F1 score. This outcome is remarkable considering the class imbalance in many cases, such as samples from Brain, Placenta, or Spleen, where more than 75% of examples belong to a single class, or in organs like Kidney, Large Intestine, Lung, and Pancreas, where there are more than 15 possible classes. Interestingly, the linear SVM achieved slightly lower but very close results, with an accuracy of 92.3% and an F1 score of 85.6%. This closeness suggests that even simple linear models can be enhanced using the pre-processing steps used in Geneformer. In some instances (such as Brain and Lung), the two approaches were very close, while in others it was significantly lower, as in the case of Liver. In these latter examples, the contextual information and pre-training provided by Geneformer were beneficial. Overall, these results suggest that SVM is capable of proficiently tackling the task, implying that the genes with the most positive/negative influence from the individual dimensions of the hyperplane are likely to be useful.

For instance, the three most discriminative genes for the large intestine, according to the SVM, are PIGR,[6] JCHAIN,[7] and ITLN1,[8] all of which are clearly present in gastrointestinal tissues.[9]

At this point, we utilized the hyperplane dimensions (the genes) ordered from most to least informative to reduce the number of genes to feed into Geneformer. The result of this analysis is reported in Table 3. We applied different values of k for cuts in the rows, from more aggressive cuts, in which Geneformer's analysis focuses only on the 128 most relevant genes per organ, going through the 256 most informative, then 512 (an order of magnitude fewer than all known functional genes), up to ALL (meaning no cut was applied). Subsequently, we sought to determine whether this analysis allowed the most informative genes to be at the head of the sequences provided to Geneformer. Therefore, we applied cuts by selecting from each sequence only the topmost l genes, e.g., 16 genes, then 32, 64, 128, and so on, up to not applying any cut, meaning all 2048 genes present in the original data are included, as shown in the columns of the table. For each combination of k (rows) and l (columns), we re-ran the entire analysis, and Table 3 presents synthetically the average F1 scores obtained across all organs.

The outcome is particularly striking since the F1 score of 87.4% reported in the experiments in Table 2, corresponding to the measure with $k = 20{,}000$ and $l = 2048$, is essentially replicated with $k = 1024$ and $l = 256$ (F1 of 87.0%), and even surpassed with a selection of $k = 2048$ and $l = 512$, achieving an F1 score of 88.0%. Evidently, the use of an SVM-informed filter is consistently informative for achieving an F1 score of at least 84% (which is practically equivalent to the original Geneformer's performance). This explainable AI method allows, for example, to reduce from about 20,000 to only 512 possible symbols, and sequences as short as $l = 64$ genes remain highly informative, with an F1 score of 84.7%. Selecting only $l = 64$ genes would not be feasible without selection; that is, with $k = 20{,}000$ and $l = 64$, an F1 score of 68.1% would be achieved, which is lower than the limit case where only $l = 16$ genes are used to discriminate cells, yet still obtaining an F1 score of 72%. In general, then, working with short sequences requires informed gene selections. It is impressive to note that by looking at only 256 genes (and this applies to all organs, so some may even need fewer genes) the system manages to achieve an F1=80.9%.

In summary, the gene selection by SVM is extremely informative and validates the soundness of the approach. However, in these tests, we are examining a case where not all sequences saturate $l = 2048$. In the next task, we will examine longer sequences, exceeding 16,000 symbols, on average.

4.2. Breast type classification

Task and Data. The objective of this experiment is to apply the proposed approach to the Breast Cancer Type Classification task. This task involves assigning a patient to one of the following classes: Basal, HER2, Luminal A (LumA), and Luminal B (LumB). A further goal is to verify the method's ability to generalize across different datasets derived from different cohorts.

In detail, our model was trained and developed using the TCGA-BRCA dataset.[10] This dataset's composition was approximately 60% White, 17% African American, 6% Asian, and 9% not reported. The original dataset comprised 1098 cases, reduced to 945 after discarding healthy cells. Each case consisted of a gene expression profile as FPKM values, with an average length of 16,960 genes per sequence. This is significantly larger than the maximum length of 2048 symbols that Geneformer can inherently manage. The testing was conducted on the SMC Dataset,[11] which comprised a 100% Korean Breast Cancer cohort. The original SMC dataset of 187 cases was narrowed down to 166, focusing on the target breast cancer types. The average length of gene sequences in this dataset was 15392. The distribution of examples for each class within the TCGA-BRCA train/dev set and the test set from SMC are reported in Table 4.

[4] https://www.kelp-ml.org/

[5] https://huggingface.co/ctheodoris/Geneformer

[6] https://www.proteinatlas.org/ENSG00000162896-PIGR

[7] https://www.proteinatlas.org/ENSG00000132465-JCHAIN

[8] https://www.proteinatlas.org/ENSG00000179914-ITLN1

[9] Although a systematic manual analysis of all these pieces of evidence is beyond the scope of this paper and is left for future work, however, these lists for each organ will be released publicly upon the acceptance of the work.

[10] https://portal.gdc.cancer.gov/projects/TCGA-BRCA

[11] https://www.cbioportal.org/study/clinicalData?id=brca_smc_2018

Table 3
F1 Scores averaged across all organs by varying sequence lengths and dictionary sizes. Note: the F1 value in the last row and last column corresponds to the Average F1 obtained in Table 2.

		Sequence Length (l)							
		16	32	64	128	256	512	1024	2048
Dict Size (k)	128	77,3%	78,0%	78,2%	78,3%	78,3%	78,3%	78,3%	78,3%
	256	76,0%	80,9%	82,6%	82,7%	82,7%	82,7%	82,7%	82,7%
	512	72.1%	79.8%	84.4%	84.7%	85.6%	85.6%	85.7%	85.0%
	1024	62.9%	75.2%	82.3%	86.3%	87.0%	86.7%	86.7%	86.7%
	2048	54.8%	66.0%	78.5%	84.7%	87.2%	**88.0%**	87.9%	87.9%
	4096	53.9%	61.2%	72.7%	81.8%	86.1%	87.6%	86.6%	86.5%
	ALL	52.7%	61.2%	68.1%	78.0%	82.7%	86.6%	87.7%	*87.4%*

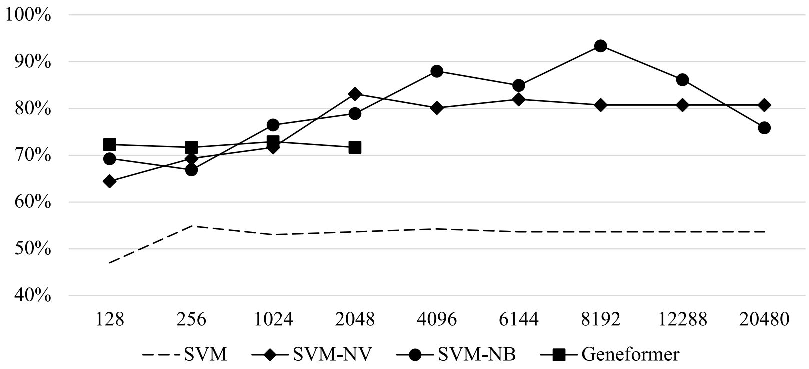


Fig. 3. Model performance metrics at selected token lengths.

Table 4
Distribution of examples for the Breast Cancer Types across different classes and datasets. The split in the percentage of training and testing examples is 85% and 15%, respectively.

Cancer type	TCGA train	TCGA Dev	SMC
Basal	155	16	36
Her2	69	9	18
LumA	453	46	47
LumB	173	24	65
Total	945		166

In this part of the study, we compare several models, all trained on the training portion of the TCGA dataset, with parameter tuning performed on the TCGA development set and testing on the SMC dataset. Each of the following SVM models have been optimized by tuning the trade-off parameter C of the SVM on the dataset, taking values from the range $[10^{-3}, 10^{-2}, 0.1, 1, 10]$.

The first model, serving as our baseline, is a Support Vector Machine (**SVM**), which operates directly on the feature vector derived from the gene expression data. This implies that the dimensionality of each vector is equal to the number of expressed genes, which is about 20,500. This method represents our approach without the statistics from Genecorpus30M, as the values used are the original ones. For efficiency reasons, we used the Dual Coordinate Descent (DCD) algorithm implemented in the KELP library,[12] which operates directly in the primal space of observations, hence it is a linear model.

The second model is $\textbf{SVM}^{NV}$, which observes values normalized according to the non-zero median value provided by Genecorpus-30M. In this case,genes with low expressiveness relative to the median observed in Genecorpus-30M have been penalized. It is noteworthy that the actual measured median values in Genecorpus30M are irrelevant since these median values are solely used to normalize our observations and will be applied consistently across all measurements.

The third model is $\textbf{SVM}^{NB}$, where values are normalized and substituted with their boolean version, with each dimension being 0 or 1. This approach is of interest as it aims to be independent not only from individual measurements in a dataset but also from any discrepancies observed between the TCGA and SMC datasets.

Finally, the last tested model was Geneformer. The parameters used for Geneformer follows: the learning rate is 5×10^{-4}; the batch size for training and evaluation is 12; the optimizer is AdamW with a Linear scheduler, which linearly improved the learning rate for the first 10% of the training steps; Weight decay is adopted and set to 0.001. The models were trained for 20 epochs, selecting the models that maximized the accuracy on the Development Set.

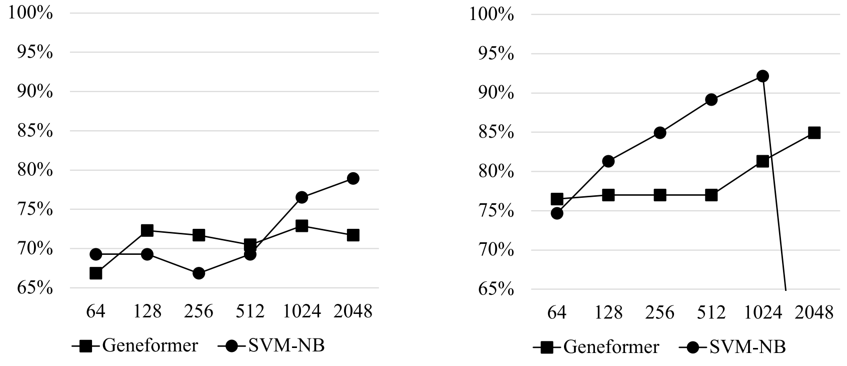
Results and Discussion. The results of the experiment are presented in Table 5 and depicted in Fig. 3. The models are listed in rows, while the columns indicate the level of cuts k. For each model, only the genes with the highest expressiveness levels were preserved; for SVM, this is before normalization, while for the other models, this follows normalization. For the SVM, input is ordered using the original "activation score" without any reordering. This approach did not prove effective, as selecting genes based on the original activation score led to a limited accuracy of around 55%. However, when reordering according to the 30-million cell statistics from Genecorpus-30M, the results exhibit a significant change. Up to 2048 dimensions, both Genformer and SVM models exhibit comparable performance; but beyond this threshold, the inclusion of more evidence appears to be impactful.

Interestingly, a boolean representation of the gene expression data seems to yield more robust results across different datasets. With a sequence length of 8192, an impressive accuracy of 93% is achieved. Geneformer demonstrates its effectiveness by improving the accuracy from 55% to 72% but is inherently limited to handling up to 2048 genes. Nevertheless, normalization and the corresponding reordering based on the 30-million cell statistics have shown to be highly beneficial. By enabling the use of far more genes, accuracy is further increased from 72% to 83%. The adoption of a boolean representation appears to enhance robustness across different datasets even further, as evidenced by the jump in accuracy from 83% to 93%. This suggests that boolean normalization not only simplifies the data but also helps bridging the

[12] https://www.kelp-ml.org/

Table 5
Model performance metrics at selected token lengths.

	128	256	1024	2048	4096	6144	8192	12 288	20 480
SVM	47.0%	54.8%	53.0%	53.6%	54.2%	53.6%	53.6%	53.6%	53.6%
SVM^{NV}	64.5%	69.3%	71.7%	83.1%	80.1%	81.9%	80.7%	80.7%	80.7%
SVM^{NB}	69.3%	66.9%	76.5%	78.9%	88.0%	84.9%	93.4%	86.1%	75.9%
Genef.	72.3%	71.7%	72.9%	71.7%	–	–	–	–	–



(a) Without gene selection

(b) With gene selection

Fig. 4. Comparison of SVM and Geneformer results. On the left are the models without the selection of the most informative genes. On the right, the same models after selecting the top 2048 most informative genes using the best-performing SVM model.

gap between the TCGA and SMC datasets, making it easier for the model to be generalized.

At this point, we questioned whether the SVM^{NB} model, in addition to achieving interesting results, had also selected genes that were relevant and informative for this type of analysis. Rather than manual validation or literature analysis, we considered a quantitative approach. The underlying hypothesis was that if the genes identified by SVM^{NB} are indeed informative, a classifier that focuses solely on these genes should demonstrate superior performance. To test this hypothesis, we selected the top m genes (e.g., 2048 to align with Geneformer's capacity) that exhibited the highest absolute value in their contribution to the classification decision across all classifiers. This selection also included genes with original negative weights, as they reflect genes strongly opposing a decision. We then repeated the analysis considering only these 2048 genes.

The results of this focused analysis are depicted in Fig. 4. On the left, we present the accuracy of the model when observing a "few genes", ranging from 64 (very few) to 2048 (the maximum cutoff for Geneformer). On the right, we display the results of both systems when the analysis is confined to the 2048 genes deemed informative by the best-performing model. Both models exhibit considerable improvement when this filter is applied. For instance, the SVM model achieves an accuracy of 92% at a dimension of 1024, a substantial increase compared to the 8192 dimensions required previously. Similarly, Geneformer improves from approximately 72% to 85% accuracy.

Finally, to provide a deeper understanding of the performance differences between the linear SVM and Geneformer, we present an error analysis through some confusion matrices. In particular, the tables below show the confusion matrices for the linear SVM and Geneformer under the best performing conditions: the SVM model at a dimension of 1024 selected genes (see Table 6), and Geneformer at a dimension of 2048 (see Table 7).

The confusion matrices reveal that, while both models follow a similar pattern in their predictions, there is a notable increase in confusion between the Her2 and LumB classes in the Geneformer model. Specifically, Geneformer shows a higher number of misclassifications between Her2 and LumB (11 Her2 instances misclassified as LumB and 9 LumB instances misclassified as Her2) compared to the linear SVM (2 Her2 instances misclassified as LumB and 2 LumB instances misclassified as Her2).

Table 6
Confusion Matrix for the SVM model at a dimension of 1024 selected genes (92% accuracy).

	Basal	Her2	LumA	LumB
Basal	36	0	0	0
Her2	0	17	0	1
LumA	0	0	45	2
LumB	1	2	7	55

Table 7
Confusion Matrix for Geneformer at a dimension of 2048 selected genes (85% accuracy).

	Basal	Her2	LumA	LumB
Basal	33	3	0	0
Her2	3	11	1	3
LumA	0	0	44	3
LumB	1	9	2	53

Several factors might explain this increased confusion in the Geneformer model. The Geneformer model, handling a larger number of genes, might overfit specific patterns within the training data that do not generalize well to the test data, leading to more misclassifications. Geneformer might be capturing non-linear relationships that, while generally beneficial, introduce noise into the classification of closely related classes like Her2 and LumB. The inherent imbalance in the number of samples for each class could be contributing to the misclassification, as the model might have a harder time distinguishing between underrepresented classes.

Anyway, despite these minor issues, both models demonstrate strong overall performance, highlighting the effectiveness of our gene selection and model training approach.

4.3. Final remarks

In our experimental analysis, both Geneformer and SVM models have achieved commendable results, with accuracy reaching up to 93% in the task of Breast Cancer Type Classification. This high level of accuracy underscores the potential of machine learning models in precision medicine and, more specifically, in the genomic analysis of cancer.

Despite its good performance, Geneformer does not appear to significantly change the game in our experimental setup. One of the main reasons for this is its built-in limitation of processing a maximum of 2048 genes. This constraint is somewhat mitigated by the application of re-ordering based on the analysis of the 30-million cell dataset, which has proven to be advantageous. By prioritizing genes according to their relevance as indicated by large-scale cellular statistics, we enhance the model's ability to focus on the most impactful features.

The linear SVM model, on the other hand, demonstrates its effectiveness in deriving "informative" genes. This capability is essential for concentrating the analysis on a crucial subset of genes, effectively compressing the data without a loss in performance. Such a focused approach is not only beneficial for Geneformer, but it also enhances the SVM model's efficiency, allowing it to achieve high accuracy with a significantly reduced feature set.

The concept of data "compression" that emerges from our experiments suggests a promising avenue for handling vast genomic datasets. By identifying and retaining only the most informative genes, we can streamline the analytical process, reducing computational costs while maintaining, or even improving, the accuracy of the predictions. This strategy is particularly relevant when dealing with large-scale genomic data, opening the door to more efficient and effective analysis in the field of bioinformatics.

Our research, which combined SVM classifiers with the Geneformer model in genetic sequence analysis for cancer diagnostics, has shown promising results in terms of accuracy and efficiency. From a computational standpoint, the incorporation of SVM using the Dual Coordinate Descent (DCD) learning algorithm proved to be minimally burdensome. On average, it took approximately 10 s per organ to train an SVM model on a standard laptop CPU, without the need for GPU acceleration. In contrast, Geneformer required around 10–12 min for training on a dataset of roughly 10,000 examples using an Nvidia T4 with 16 GB of RAM. This stark difference in computational requirements highlights the efficiency of our proposed methodology.

5. Conclusion

In this study, we showed the potential of combining transformer-based learning and explainable AI methods in medical AI, particularly for breast cancer type detection. By focusing on identifying and prioritizing relevant gene subsets, we enhanced the capabilities of the Geneformer model, extending its reach beyond its inherent symbol-size limitation. This approach not only demonstrated the model's adaptability across diverse cohorts, but also emphasized its effectiveness in maintaining high levels of accuracy regardless of the population under study. Our experimental findings reveal that integrating explainable AI methods with the Geneformer model significantly advances the field of cancer genomics. The methodology provides a more nuanced understanding of genetic markers associated with different types of breast cancer.

In general, the proposed methodology offers several advantages, including enhanced interpretability, improved performance, generalizability, and computational efficiency. By leveraging explainable AI techniques, our approach identifies and prioritizes the most relevant genes for each specific task, making the model's decisions more transparent and interpretable. This is crucial in medical applications where understanding the underlying decision process can aid in clinical validation and acceptance. Our method demonstrates high accuracy in both cell type and breast cancer type classification tasks, ensuring that the model maintains or improves its performance even when the input dimensionality is significantly reduced. The model's ability to generalize across different datasets and cohorts highlights its robustness and potential applicability in diverse clinical settings. Additionally, the use of linear SVM for initial gene selection significantly reduces the computational burden, making the overall approach more efficient.

Despite these advantages, our approach has some limitations. The effectiveness of the Geneformer model relies heavily on the quality and comprehensiveness of its pre-training on large-scale datasets. Any biases or gaps in the pre-training data can affect the model's performance. Geneformer can handle sequences up to 2048 genes, which may not capture the full complexity of the transcriptome in certain cases. Although we mitigate this by selecting the most relevant genes, this limitation can affect the model's ability to utilize all available data.

In addition, while our use of linear SVM simplifies interpretability and computation, it may miss capturing complex non-linear relationships between genes. To address these complexities, an interesting avenue for future work is the exploration of Kernel-based SVMs (Vapnik, 1995), such as the Fisher kernel (Shawe-Taylor & Cristianini, 2004). Although they are less efficient than the linear approaches used in this study and do not offer straightforward interpretability, Kernel-based SVMs can capture complex, non-linear relationships among genes that linear models may miss. The trade-off between computational efficiency and the ability to model non-linear interactions needs careful consideration. Future research will investigate the integration of non-linear SVMs with explainable AI techniques to enhance interpretability, as suggested by studies like Sanz, Valim, Vegas, Oller, and Reverter (2018).

In conclusion, while our experimental results demonstrate promising outcomes in molecular and breast cancer type classification, our ongoing and future work is committed to further enhancing the model's robustness and practicality. We are currently undertaking a thorough literature and experimental evaluation of all discriminative genes extracted by our model. Additionally, we plan to address potential overfitting through advanced cross-validation techniques, integrate the model into clinical workflows, and extend our testing to a broader range of cancer types and populations. These steps are fundamental for confirming the model's effectiveness in diverse medical contexts.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

Data are publicly available.

References

Akhavan, M., & Hasheminejad, S. M. H. (2023). A two-phase gene selection method using anomaly detection and genetic algorithm for microarray data. *Knowledge-Based Systems*, *262*, Article 110249. http://dx.doi.org/10.1016/j.knosys.2022.110249.

Amelio, I., Bertolo, R., Bove, P., Buonomo, O. C., Candi, E., Chiocchi, M., et al. (2020). Liquid biopsies and cancer omics. *Cell Death Discovery*, *6*(1), 131.

Amelio, I., Bertolo, R., Bove, P., Candi, E., Chiocchi, M., Cipriani, C., et al. (2020). Cancer predictive studies. *Biology Direct*, *15*(1), 1–7.

Aziz, R., Verma, C., & Srivastava, N. (2017). A novel approach for dimension reduction of microarray. *Computational Biology and Chemistry*, *71*, 161–169.

Aziz, R., Verma, C., & Srivastava, N. (2018). Artificial neural network classification of high dimensional data with novel optimization approach of dimension reduction. *Annals of Data Science*, *5*, 615–635.

Bello, I., Zoph, B., Vaswani, A., Shlens, J., & Le, Q. V. (2019). Attention augmented convolutional networks. In *Proceedings of the IEEE/CVF international conference on computer vision* (pp. 3286–3295).

Bishop, C. M. (2007). *Pattern recognition and machine learning (Information science and statistics)* (1). Springer.

Bouraoui, Z., Camacho-Collados, J., & Schockaert, S. (2020). Inducing relational knowledge from BERT. In *The thirty-fourth AAAI conference on artificial intelligence, AAAI 2020, the thirty-second innovative applications of artificial intelligence conference, IAAI 2020, the tenth AAAI symposium on educational advances in artificial intelligence* (pp. 7456–7463). AAAI Press, http://dx.doi.org/10.1609/AAAI.V34I05.6242.

Carion, N., Massa, F., Synnaeve, G., Usunier, N., Kirillov, A., & Zagoruyko, S. (2020). End-to-end object detection with transformers. In *European conference on computer vision* (pp. 213–229). Springer.

Consortium, R. E., Kundaje, A., Meuleman, W., Ernst, J., Bilenky, M., Yen, A., et al. (2015). Integrative analysis of 111 reference human epigenomes open. *Nat.*, *518*(7539), 317–330. http://dx.doi.org/10.1038/NATURE14248.

Consortium, E. P., et al. (2012). An integrated encyclopedia of DNA elements in the human genome. *Nature*, *489*(7414), 57.

Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. (2019). BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 conference of the North American chapter of the association for computational linguistics: human language technologies, volume 1 (long and short papers)* (pp. 4171–4186). Minneapolis, Minnesota: Association for Computational Linguistics, URL https://www.aclweb.org/anthology/N19-1423.

Došilović, F. K., Brčić, M., & Hlupić, N. (2018). Explainable artificial intelligence: A survey. In *2018 41st international convention on information and communication technology, electronics and microelectronics* (pp. 0210–0215). http://dx.doi.org/10.23919/MIPRO.2018.8400040.

Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., et al. (2020). An image is worth 16x16 words: Transformers for image recognition at scale. arXiv preprint arXiv:2010.11929.

Esteva, A., Chou, K., Yeung, S., Naik, N., Madani, A., Mottaghi, A., et al. (2021). Deep learning-enabled medical computer vision. *NPJ digital medicine*, *4*(1), 5.

Fletcher, C., Pestova, T. V., Hellen, C. U., & Wagner, G. (1999). Structure and interactions of the translation initiation factor eIF1. *EMBO Journal*, *18*(9), 2631–2637. http://dx.doi.org/10.1093/emboj/18.9.2631.

Ganini, C., Amelio, I., Bertolo, R., Bove, P., Buonomo, O. C., Candi, E., et al. (2021). Global mapping of cancers: The cancer genome atlas and beyond. *Molecular Oncology*, *15*(11), 2823–2840.

Girshick, R. B., Donahue, J., Darrell, T., & Malik, J. (2013). Rich feature hierarchies for accurate object detection and semantic segmentation. *CoRR*, arXiv:1311.2524.

Hassan Zadeh, A., Alsabi, Q., Ramirez-Vick, J. E., & Nosoudi, N. (2020). Characterizing basal-like triple negative breast cancer using gene expression analysis: A data mining approach. *Expert Systems with Applications*, *148*, Article 113253. http://dx.doi.org/10.1016/j.eswa.2020.113253, URL https://www.sciencedirect.com/science/article/pii/S0957417420300786.

Hsieh, C.-J., Chang, K.-W., Lin, C.-J., Keerthi, S. S., & Sundararajan, S. (2008). A dual coordinate descent method for large-scale linear SVM. In *Proceedings of the ICML 2008* (pp. 408–415). New York, NY, USA: ACM.

Kawai, J., Shinagawa, A., Shibata, K., Yoshino, M., Itoh, M., Ishii, Y., et al. (2001). The RIKEN genome exploration research group phase II team and the FANTOM consortium: functional annotation of a full-length mouse cDNA collection. *Nature*, *409*(6821), 685–690.

Khan, A., & Lee, B. (2023). DeepGene transformer: Transformer for the gene expression-based classification of cancer subtypes. *Expert Systems with Applications*, *226*, Article 120047. http://dx.doi.org/10.1016/j.eswa.2023.120047, URL https://www.sciencedirect.com/science/article/pii/S0957417423005493.

Kumar, Y., Gupta, S., Singla, R., & Hu, Y.-C. (2022). A systematic review of artificial intelligence techniques in cancer prediction and diagnosis. *Archives of Computational Methods in Engineering*, *29*(4), 2043–2070.

Van der Laak, J., Litjens, G., & Ciompi, F. (2021). Deep learning in histopathology: the path to the clinic. *Nature Medicine*, *27*(5), 775–784.

Lander, E. S., Linton, L. M., Birren, B., Nusbaum, C., Zody, M. C., Baldwin, J., et al. (2001). Initial sequencing and analysis of the human genome. *Nature*, *409*(6822), 860–921.

Lin, T., Wang, Y., Liu, X., & Qiu, X. (2022). A survey of transformers. *AI Open*, *3*, 111–132. http://dx.doi.org/10.1016/J.AIOPEN.2022.10.001.

Miguel, J. P. M., Neves, L. A., Martins, A. S., do Nascimento, M. Z., & Tosta, T. A. A. (2023). Analysis of neural networks trained with evolutionary algorithms for the classification of breast cancer histological images. *Expert Systems with Applications*, *231*, Article 120609. http://dx.doi.org/10.1016/j.eswa.2023.120609, URL https://www.sciencedirect.com/science/article/pii/S0957417423011119.

Momeni, Z., Hassanzadeh, E., Saniee Abadeh, M., & Bellazzi, R. (2020). A survey on single and multi omics data mining methods in cancer data classification. *Journal of Biomedical Informatics*, *107*, Article 103466. http://dx.doi.org/10.1016/j.jbi.2020.103466, URL https://www.sciencedirect.com/science/article/pii/S1532046420300939.

Osama, S., Shaban, H., & Ali, A. A. (2023). Gene reduction and machine learning algorithms for cancer classification based on microarray gene expression data: A comprehensive review. *Expert Systems with Applications*, *213*, Article 118946. http://dx.doi.org/10.1016/j.eswa.2022.118946.

Platt, J. (1998). *Sequential minimal optimization: A fast algorithm for training support vector machines: Technical Report MSR-TR-98-14*, Microsoft.

Rajpal, S., Rajpal, A., Saggar, A., Vaid, A. K., Kumar, V., Agarwal, M., et al. (2023). XAI-MethylMarker: Explainable AI approach for biomarker discovery for breast cancer subtype classification using methylation data. *Expert Systems with Applications*, *225*, Article 120130. http://dx.doi.org/10.1016/j.eswa.2023.120130, URL https://www.sciencedirect.com/science/article/pii/S0957417423006322.

Sanz, H., Valim, C., Vegas, E., Oller, J. M., & Reverter, F. (2018). SVM-RFE: selection and visualization of the most relevant features through non-linear kernels. *BMC Bioinform.*, *19*(1), 432:1–432:18. http://dx.doi.org/10.1186/S12859-018-2451-4.

Saxena, A., Chouhan, S. S., Aziz, R. M., & Agarwal, V. (2024). A comprehensive evaluation of marine predator chaotic algorithm for feature selection of COVID-19. *Evolving Systems*, 1–14.

Schuster, M., & Nakajima, K. (2012). Japanese and Korean voice search. In *International conference on acoustics, speech and signal processing* (pp. 5149–5152).

Shawe-Taylor, J., & Cristianini, N. (2004). *Kernel methods for pattern analysis* (illustrated edition). Cambridge University Press, URL http://www.amazon.com/Kernel-Methods-Pattern-Analysis-Shawe-Taylor/dp/0521813972.

Stephens, Z. D., Lee, S. Y., Faghri, F., Campbell, R. H., Zhai, C., Efron, M. J., et al. (2015). Big data: Astronomical or genomical? *PLoS Biol*, *13*(7), Article e1002195.

Sung, H., Ferlay, J., Siegel, R. L., Laversanne, M., Soerjomataram, I., Jemal, A., et al. (2021). Global cancer statistics 2020: GLOBOCAN estimates of incidence and mortality worldwide for 36 cancers in 185 countries. *CA: A Cancer Journal for Clinicians*, *71*(3), 209–249.

Theodoris, C. V., Xiao, L., Chopra, A., Chaffin, M. D., Sayed, Z. R. A., Hill, M. C., et al. (2023). Transfer learning enables predictions in network biology. *Nature*, *618*(7965), 616–624. http://dx.doi.org/10.1038/s41586-023-06139-.

Vanitha, C. D. A., Devaraj, D., & Venkatesulu, i. (2015). Gene expression data classification using support vector machine and mutual information-based gene selection. *Procedia Computer Science*, *47*, 13–21.

Vapnik, V. (1995). *The nature of statistical learning theory*. Springer.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., et al. (2017). Attention is all you need. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, R. Garnett (Eds.), *Advances in neural information processing systems 30* (pp. 5998–6008). Curran Associates, Inc., URL http://papers.nips.cc/paper/7181-attention-is-all-you-need.pdf.

Vitale, I., Pietrocola, F., Guilbaud, E., Aaronson, S. A., Abrams, J. M., Adam, D., et al. (2023). Apoptotic cell death in disease—Current understanding of the nccd 2023. *Cell Death & Differentiation*, 1–58.

Watson, J. D., & Crick, F. H. (1953). Molecular structure of nucleic acids: a structure for deoxyribose nucleic acid. *Nature*, *171*(4356), 737–738.

Wittgenstein, L. (1953). *Philosophical investigations*. Oxford: Basil Blackwell.

Xin, C., Liu, Z., Zhao, K., Miao, L., Ma, Y., Zhu, X., et al. (2022). An improved transformer network for skin cancer classification. *Computers in Biology and Medicine*, *149*, Article 105939. http://dx.doi.org/10.1016/j.compbiomed.2022.105939, URL https://www.sciencedirect.com/science/article/pii/S0010482522006746.

Yaqoob, A., Verma, N. K., & Aziz, R. M. (2024). Metaheuristic algorithms and their applications in different fields: A comprehensive review. *Metaheuristics for Machine Learning: Algorithms and Applications*, 1–35.

Yaqoob, A., Verma, N. K., Aziz, R. M., & Saxena, A. (2024). Enhancing feature selection through metaheuristic hybrid cuckoo search and harris hawks optimization for cancer classification. *Metaheuristics for Machine Learning: Algorithms and Applications*, 95–134.

Yue, T., Wang, Y., Zhang, L., Gu, C., Xue, H., Wang, W., et al. (2023). Deep learning for genomics: From early neural nets to modern large language models. *International Journal of Molecular Sciences*, *24*(21), http://dx.doi.org/10.3390/ijms242115858, URL https://www.mdpi.com/1422-0067/24/21/15858.

Zhao, Y., Li, M.-C., Konaté, M. M., Chen, L., Das, B., Karlovich, C., et al. (2021). TPM, FPKM, or normalized counts? A comparative study of quantification measures for the analysis of RNA-seq data from the NCI patient-derived models repository. *Journal of Translational Medicine*, *19*(1), 269. http://dx.doi.org/10.1186/s12967-021-02936-w.

Zhou, H., Chen, H., Yu, B., Pang, S., Cong, X., & Cong, L. (2024). An end-to-end weakly supervised learning framework for cancer subtype classification using histopathological slides. *Expert Systems with Applications*, *237*, Article 121379. http://dx.doi.org/10.1016/j.eswa.2023.121379, URL https://www.sciencedirect.com/science/article/pii/S095741742301881X.