Visual Insights in Human Cancer Mutational Patterns: Similarity-Based Cancer Classification Using Siamese Networks

Rocco Zaccagnino, Clelia De Felice, Marco Russo and Rosalba Zizza
Dip. di Informatica, University of Salerno, Salerno, Italy

Keywords: Cancer Detection, Siamese Neural Networks, Mutational Signature, Explainable AI, Information Visualization.

Abstract: In recent years, a number of innovations concerning the diagnosis and treatment of diseases through the application of genomics have opened the door to the detailed analysis of somatic mutation patterns in human cancers. Several AI-based systems have been proposed to identify correlations between mutations and type of cancer. However, the use of AI in Bioinformatics still presents two main limitations: *(i)* the *explainability*, i.e., the ability of the methods to partially explain and motivate their behavior, and *(ii)* the *usability*, i.e., about the strong limitations that are found in the actual use of such methods in real bio-medical contexts and scenarios. In this work, we propose a novel ML-based cancer-type detection system which integrates explainability and usability techniques. To this aim, we first formulate the cancer-type detection problem using the *similarity-based classification* paradigm. Then, given a cancer sample, we assume to have a set of somatic mutation features available which can be interpreted as *cancer mutational view* of the sample itself. Finally, we propose the use of a special Machine Learning model defined for learning similarity functions, namely the Siamese Neural Network (SNN). The proposed SNN learns to take a pair of *cancer mutational views* as input, and to compute a similarity score that can be used to verify whether such samples are similar or not.

Preliminary experiments carried out to assess the effectiveness of the proposed system show high performance reaching f1 score 97.61%, and highlight how the similarity-based classification paradigm could be more suitable than the commonly used classification paradigm for the formulation of the cancer-type detection problem.

1 INTRODUCTION

1.1 Cancer and Somatic Mutations

In recent years, a number of technical innovations have been developed regarding the diagnosis and treatment of diseases through the application of genomics. The most evident result is the standardization of tumor profiling techniques based on recurrent targeted *mutations analysis*. This has led to an evident efficacy of molecularly targeted therapies on distinct types of tumor by exploiting information regarding shared genetic features. Today, based on recent large-scale exome and genome-sequencing studies, we know that major tumour types present specific patterns of somatic mutations (Kandoth et al., 2013; Lawrence et al., 2013; Ciriello et al., 2013).

In this direction, several research initiatives have developed recently. As an example, at Memorial Sloan Kettering Cancer Center[1], a NGS panel named `msk-impact` has been developed to show the feasibility and utility of large-scale prospective clinical sequencing of tumors to guide clinical management. `msk-impact` has been used to detect all protein-coding mutations, copy number alterations, and selected promoter mutations and structural rearrangements in 410 cancer-associated genes, for a total of 62 sequenced principal tumors from more than 10,000 patients. The result is a comprehensive and detailed catalog of somatic mutations for every tumor sequenced, publicly available online[2].

1.2 Contribution of this Work

Explainable and Usable AI. Artificial Intelligence (AI) and in particular Machine Learning (ML) systems are increasingly used in Bioinformatics. This because the massive amounts of bio-medical data, in-

[1] https://www.mskcc.org/

[2] http://cbioportal.org/msk-impact

Zaccagnino, R., De Felice, C., Russo, M. and Zizza, R.
Visual Insights in Human Cancer Mutational Patterns: Similarity-Based Cancer Classification Using Siamese Networks.
DOI: 10.5220/0012399600003657

In *Proceedings of the 17th International Joint Conference on Biomedical Engineering Systems and Technologies (BIOSTEC 2024) - Volume 1*, pages 462-470
ISBN: 978-989-758-688-0; ISSN: 2184-4305

cluding heterogeneous high-dimensional data, introduce challenges to existing ML methods (Karim et al., 2021), which are increasingly being used successfully for data analysis and interpretation.

To date, the use of AI techniques in Bioinformatics has two main limitations. The first is the so-called *explainable AI* (XAI), i.e., the ability of the methods to partially explain or motivate their behavior, while the second is about the *usable AI*, i.e., the actual use of such systems in real-world scenarios.

While ML models are able to address complex problems, their "black-box" nature raises concerns about transparency and accountability, which also overshadow their ability to solve the problems themselves. The field of XAI aims to make AI systems more transparent by explaining how they make decisions and so to enhance the human-comprehensibility, reasoning, transparency, and accountability.

As mentioned earlier, another strong limitation of the use of AI in Bioinformatics is about the actual "usability" of such systems in real-world scenarios. Advanced ML models facing really complex problems often suffer from scalability problems. In some cases, the motivation could be found in the "classification" paradigm used to formulate the problem faced: there are *n* classes of samples, and the model is trained on a training set to classify a new sample in one of such *n* classes. This approach, especially in Bioinformatics, could suffers from some issues, including the enormous amount of data on which the model must be trained, the strong imbalance of the classes that can arise when working on real data, and above all the problem of scaling the model when new classes of samples must be classified. In this case, the model must be retrained on the whole set of data, with severe impact on the computational effort, but also in contexts where a timely response can be crucial.

Proposed Strategy. We propose a novel ML-based cancer-type detection system with the the aim of integrating it with explainability and usability techniques. We first formulate such a problem in terms of *similarity-based classification* (Chen et al., 2009).

Given a cancer sample, we assume to have a set of somatic mutation features available which can be interpreted as a *cancer mutational view* of the sample itself. Then, according to the central idea of the similarity-based classification paradigm, we define a model which does not simply learn to classify a cancer sample by observing its cancer mutational view, but which is able to learn, starting from a set of sample pairs, a similarity function and which therefore is able to tell whether two samples are similar or not. Clearly, the more the starting set of samples is representative of the problem, the more accurate the function is. The advantage of this approach is that once the similarity function has been calculated, the model can also be used on new samples (even of a cancer-type never seen during the training) of which to find out which classes are more similar to. Furthermore, to make the system scalable on large amounts of data, we keep track, for each cancer-type class, of one single representative view, and using them to find out which classes are more similar to a test view, with great benefits both in terms of memory and privacy.

There are numerous examples of works in Bioinformatics based on the similarity-based classification paradigm (Mathai and Kirchmair, 2020). In this paper, we propose the usage of special ML models defined for learning similarity functions, i.e., *Siamese Neural Networks* (SNN). We define a novel SNN which given a pair of cancer mutational views outputs a similarity score that can be used to verify that they are similar. The proposed solution is based on the following two main ideas that, in our opinion, could limitate the issues discussed above. First, the somatic mutation features of a cancer sample could be used as "similarity view" that can be exploited as effective feature embedding for ML methods. Second, we show that the SNN increases the level of discrimination strength within the proposed cancer mutational views (Bell and Bala, 2015).

Several studies have been proposed in the literature to face the problem of using ML techniques to determine tumour organ of origin and histology using the patterns of somatic mutation identified by whole genome DNA sequencing, such as (Jiao et al., 2020). However, most of these are based on the classification paradigm. Furthermore, several works use SNNs in Bioinformatics (Bechar et al., 2023; Narmatha et al., 2023), but to the best of our knowledge this is the fist attempt to propose a similarity-based classification paradigm based on SNNs exploiting somatic mutation features for the cancer-type detection problem.

Our Contributions:

- A novel *cancer-type detector* integrating explainability and usability techniques, and based on cancer mutational views for training SNNs at verifying the similarity between cancer samples.
- Preliminary experiments to assess the effectiveness of the proposed method; results obtained on a dataset of somatic mutation features show accuracy 89.25%, precision 97.60%, recall 97.63%, and f1 score 97.63%, highlighting the advantages of the similarity-based classification paradigm.

Source code and files are available online[3].

2 THE PROPOSED SYSTEM

In this section, we describe a novel ML-based cancer-type detection system. We assume that the reader is familiar with ML notions. For further details, refer to (Tan et al., 2016).

2.1 Overview

Here, we provide an overview of the scenario in which the proposed system can be placed (Figure 1).

- *Usability*. The system must be designed to be able to manage views in a scalable and efficient way. To this aim, the typical scenario in which we imagine it could be used is the one in which it is used to store cancer mutational views to be compared from time to time with new test cancer samples that are analyzed to find out their type. More in detail, at every moment it has in memory a representative view of each type of cancer analyzed up to that moment. Each time a new cancer sample c_t must be detected, the corresponding cancer mutational view s_t, named *test view*, is provided to the system; during the search, s_t is compared with every stored enrollment view; then, the system returns the type of cancer corresponding to the enrollment view s_e (corresponding to a specific cancer sample c_e) which is most similar to s_t, formally denoted with $c_t \sim c_e$. We assume that if this level of similarity does not exceed a threshold (established during the training of the S), then s_t is a sample of a new type of cancer and therefore will be memorized as a view of this new type.

  The advantages of such a system are numerous. First, there is no need to keep in memory a huge amount of data relating to samples to be used for a re-training of the ML model, but for each type of cancer only the view of a representative sample is stored. Furthermore, a significant implication is that of data privacy, which in this case must focus on the privacy of a very small set of data.

- *Explainability*. S has been designed to integrate the *attemption mechanism*, through special layers. “Attention” was first used in computer vision, inspired by the idea to mimic the attention ability of the human brains to deal with the massive amount of visual input. Attention layers mainly consist in a weighted mean reduction, where each element is weighted in proportion to its contribution to the mean. One way to interpret the attention weights is to plot them as a *feature heatmap*, where each row corresponds to an output item and each column corresponds to an input feature, and the color or intensity of each cell indicates the level of the attention weight. This can help you visualize which parts of the input are more important for each output. Thus, by showing the visualization of the feature heatmap of the attemption layer we can interpret the relation between the features and better understand the key issues which affect the performance of S. As we will see in Section 3, such heatmaps can be used to highlight the most relevant somatic mutations in the several cancer types.

2.2 Cancer Mutational View

The dataset used for our experiments is extracted from the `msk-impact` (Kübler et al., 2019), a genomic profiling dataset generated by Memorial Sloan Kettering Cancer Center. It contains molecular profiling data of 10,945 successfully sequenced tumor samples from 10,336 individuals, for 62 principal tumor types. The dataset, generated using NGS technologies, includes molecular features that are relevant for cancer diagnosis, prognosis, and treatment, such as protein-coding mutations, copy number alterations (CNAs), and selected promoter mutations and structural rearrangements in 410 cancer-associated genes.

To extract the data used for our experiments, first we downloaded such a dataset[4], and then we merged the following files to into a `csv` file: `data_cna.txt`, `data_sv.txt`, `data_clinical_sample.txt`, `data_clinical_patient.txt`. The dataset obtained consists of 433 features, organized into:

- *Clinical info* (13): `Sample ID`, `Cancer Type`, `Mutation Count`, `Sex`, `Sample Type`, `DNA Input`, `Matched Status`, `Oncotree Code`, `Overall Survival Status`, `Patient's Vital Status`, `Sample Collection Source`, `Smoking History`, `Somatic Status`.
- *Structural variations info* (10): `Site1 Chr`, `Site1 Region`, `Site1 Hugo Symbol`, `Site2 Chr`, `Site2 Region`, `Site2 Hugo Symbol`, `Class`, `Connection Type`, `Tumor Variant Count`, `Breakpoint Type`.
- *Copy Numbers* (410).

[3] https://github.com/FLaTNNBio/few-shot-learning-for-cancer-detection/tree/master

[4] https://www.cbioportal.org/study/summary?id=msk_impact_2017

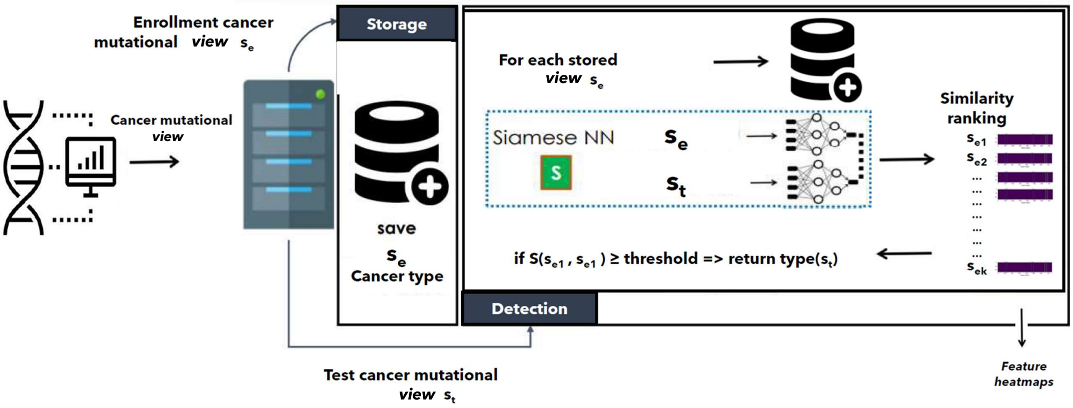
Enrollment cancer mutational view s_e
Storage
Cancer mutational view
save
s_e
Cancer type
For each stored view s_e
Siamese NN
S
s_e
s_t
Similarity ranking
s_{e1}
s_{e2}
s_{ek}
if $S(s_{e1}, s_{e1}) \geq$ threshold => return type(s_t)
Detection
Test cancer mutational view s_t
Feature heatmaps

Figure 1: The *overall scenario*. The system manages cancer mutational views. To *store* a view s_e, it must be entered, and if the corresponding cancer type is not in the system then to save it together with the cancer type; to detect a view s_t, it must compared with all the stored views; a similarity ranking $s_{e1}, \ldots, s_{ek}$ is built by using the SNN $\boxed{\mathsf{S}}$, and if the similarity score between s_t and s_{e1} is greater or equal to a threshold, then the cancer-type of s_t is the same of s_e.

At the end of this extraction phase, each cancer sample was represented by a set of 433 features.

Then, the dataset underwent a normalization procedure for numeric features, and a one-hot encoding for non-numeric features, so reaching a number of features equal to 2181. The reason for the increased number of features is due to the use of one-hot coding which notoriously could generate huge vectors since the size of a generated feature vector is equal to the number of possible values. To reduce the high dimensionality of the input data, several techniques in the literature can be applied, such as the *feature selection* procedure. However, in order to limit the loss of information that could occur by choosing which input features to keep and which to discard, in this work we have decided to use the *Principal Component Analysis* with several values for the number of components parameter. Results showed that best results have been obtained with 1403 components. For each cancer sample, this set of 1403 components is the *cancer mutational view*.

Finally, since one of the goals of this work is to compare detection by classification with that by similarity-based classification, we tried to maintain, of the 62 types of cancer managed in the starting dataset, only those that have a minimum number of instances that maximize the capability of classification models. This is because, as is known, a strong class imbalance is a problem when training classification models. From empirical observations and preliminary experiments, we have observed that by guaranteeing a minimum number of instances equal to 30, this allows us to obtain a classification model, with which we will compare ourselves, with excellent performance (see Section 3) for further details).

Table 1 reports the 16 types of cancer, i.e., the classes of our problem, which have at least 30 instances, by indicating for each of them the exact number of instances (`#instances`).

Table 1: Number of instances for each cancer-type class.

`Cancer type`	`#instances`
Prostate Cancer	336
Non-Small Cell Lung Cancer	313
Breast Cancer	242
Soft Tissue Sarcoma	104
Colorectal Cancer	100
Glioma	158
Hepatobiliary Cancer	70
Melanoma	66
Esophagogastric Cancer	63
Pancreatic Cancer	56
Bone Cancer	54
Cancer of Unknown Primary	43
Bladder Cancer	42
Ovarian Cancer	40
Head and Neck Cancer	35
Endometrial Cancer	32

2.3 The Proposed Siamese NN

In this section, we first describe the SNN $\boxed{\mathsf{S}}$ trained to compute the similarity between two cancer mutational views, and then details of the pseudo-code.

Siamese Architecture and One-Shot Learning. Given a pair of cancer mutational views $\mathtt{s}(c_i)$ and $\mathtt{s}(c_j)$, where c_i and c_j are cancer samples, $\boxed{\mathsf{S}}$ computes a similarity score $\boxed{\mathsf{S}}(\mathtt{s}(c_i), \mathtt{s}(c_j))$. Then, to verify that c_i and c_j are of the same type, the following rule is used by the system:

$$\boxed{\mathsf{S}}(\mathtt{s}(c_i), \mathtt{s}(c_j)) \geq \delta \Longrightarrow c_i \sim c_j$$

where $\delta \in [0,1]$ is the *cancer mutational view threshold* empirically estimated during the training of S. In the following, we provide details about the architecture and the training of S. S consists of three sections: the *branches*, the *info*, and the *similarity*.

The *branches* section consists of two identical subnetworks, each one defined as follows. It starts with a *Linear* layer using *ReLu* activation, which takes as input the cancer mutational view and returns a vector of size 1754. Such a layer is then followed by 5 blocks each one consisting of: *(i)* one *Linear* with *ReLu* activation function and returning a vector of size 750, *(ii)* one *Dropout* layer with probability 0.1, and *(iii)* one *BatchNormalization* layer. The *info* section essentially consists of two layers, each taking as input the concatenation of the outputs o_1 and o_2 of the two identical subnetworks described above: one *Attention* layer used to integrate S with the attention mechanism described in Section 2.1, and one *Lambda* layer used to compute the Euclidean distance between o_1 and o_2. As for the *similarity* section, the concatenation of the *Attention* layer output and of the *Lambda* layer output is given as input to 3 blocks where each block consists of: *(i)* one *Linear* layer with *ReLu* activation function and returning a vector of size 320, *(ii)* one *Drouput* layer with probability 0.1, *(iii)* one *BatchNormalization* layer. Then, the blocks are followed by *Linear* layer with output of size 1 (“similar or not similar”) and *Sigmoid* activation function.

One of the most interesting advantages of using SNNs is the ability to adopt the *One-Shot Learning* strategy, shown to be effective in identifying new classes based on one (or only a few) examples. The idea is to learn patterns and similarities on previously seen classes instead of fitting the ML model to fixed classes, in order to be able of classifying previously unseen classes using one instance. This strategy is very helpful in the scenario described in Section 2.1. Indeed, it allows us to define a detection system “calibrated” on a significant initial set of cancer-types, i.e, with a SNN trained on an initial set of cancer mutational views corresponding to a “representative” set of cancer-types; a new cancer-type can be added to the system without having to retrain the network, but simply by saving a reference cancer mutational view, used every time during the detection tasks. S is trained using the *One-Shot learning* (Algorithm 1).

Pseudocode:

- One-Shot Learning (Algorithm 1).

  It takes as input the dataset S of cancer samples organized into N cancer-type classes, and the chosen cancer mutational view similarity `threshold`. First, the algorithm initializes the weights of S (line 1), and an empty list `one-shot-accuracy` which will contain the accuracy obtained at each evaluation step (line 2). Then, for each cancer-type class $t_i \in C$, t_i is split in t_i^l (labelled samples), and t_i^u (unlabelled samples) (line 5). Each of the remaining $N-1$ classes is split into two balanced subsets (line 11): the first one using the methods `GetSimilarPairs` and `GetDissimilarPairs` to generate the training set of similar and dissimilar pairs, while the second one used as evaluation pool (lines 12 and 13). Thus, the training process (line 17) and the testing process (line 18) run, by excluding s_i. For the evaluation, the method `GetOtherPairs` (line 17) is used to build a set of *evaluation pairs* P_i. Then, using the method `Voting`, each instance $e_i \in P_i$ is classified using the class with the highest votes. Finally, the trained S and the average accuracy `one-shot-accuracy` are returned.

- *The overall detection system* (Algorithm 2). It takes as input a `cancer_sample`, and the type of `request` (*“storage”* or *“detect”*). At the beginning, the type of request is checked. If a *“storage”* is required, then the system first check if a the cancer-type of the sample is already stored in the database using the method `GetCancerType` (line 4). If a cancer type has been found, then the system communicates a cancer mutational view for the cancer-type of the input sample is already stored. Otherwise, this means that the the cancer-typer of the input sample is not stored. Then, the system saves the cancer mutational view of `cancer_sample` as enrollment view through the method `SaveCancerView` (line 8). Instead, if a *“detect”* is required, the most similar view is searched within the system (line 12).

3 PRELIMINARY EXPERIMENTS

Here, we report the results obtained during preliminary experiments carried out to assess the effectiveness of the proposed detection system. To this aim, we have compared the performance obtained by the proposed SNN S described in Section 2.3, with that obtained by a baseline Deep Neural Network (DNN) trained for classify the cancer-type of cancer samples. In these experiments, such a baseline DNN has been obtained by extracting only one of the subnetworks of the *branches* section of S.

Algorithm 1: S One-Shot Learning.

```
Input  : $C = \{t_1, \ldots, t_N\}$, threshold
Output: ⟨S, one-shot-accuracy⟩
S ← InitializeSiamese(S);
one-shot-accuracy ← [];
for $i$ = 1 to N do
    /* Select "new" speaker $s_i$
    $\langle t_i^l, t_i^u \rangle$ ← SplitSamplesByCancerType($t_i$, 0.5);
    training_set$_i$ ← $\emptyset$;
    testing_set$_i$ ← $\emptyset$;
    /* Build training/testing sets without $t_i$
    for $j$ = 1 to N do
        if $j \neq i$ then
            $\langle t_j^l, t_j^u \rangle$ ← SplitSamplesByCancerType($t_j$, 0.5);
            training_set$_i$ ← training_set$_i$ ∪ $t_j^l$;
            testing_set$_i$ ← testing_set$_i$ ∪ $t_j^u$;
    $P_t$ ← GetSimilarPairs(training_set$_i$);
    $P_d$ ← GetDissimilarPairs(training_set$_i$);
    /* Train and Test Siamese NN
    S ← Train(S, $P_t$, $P_d$, SV_threshold, "Triplet Loss");
    accuracy ← Test(S, testing_set$_i$, SV_threshold);
    /* One-Shot Evaluation
    $P_i$ ← GetOtherPairs($t_i^u$, $\{t_1, \ldots, t_{i-1}, t_i^l, t_{i+1}, \ldots, t_N\}$);
    correct ← 0;
    for $k$ = 1 to $|P_i|$ do
        $x$ ← Voting($P_i[k]$, S);
        if $x == i$ then
            /* Correct classification
            correct ← correct +1;
    accuracy$_i$ ← $\frac{correct}{100}$;
    one-shot-accuracy.append(accuracy$_i$);
return ⟨S, Average(one-shot-accuracy)⟩;
```

3.1 Results

We have split, using a *stratified* approach, the dataset into *training* set, consisting of the 70% of cancer samples of the dataset (1,403 samples), and *testing* set consisting of the 30% (351 samples). Then, the training set has been split into two subsets: *(i)* the first one consisting of the 80% (1,122 samples) and used to train both S and the baseline DNN, and *(ii)* the second one consisting of the 20% (281 samples) and used to validate both S and the baseline DNN. Finally, the testing set has been used to test the two networks.

Algorithm 2: The proposed detection system.

```
Input  : cancer_sample, request
Output: outcome
/* Check type of request
if $request$ == "storage" then
    /* Storage request
    test-view ← GetCancerType(cancer_sample);
    if $test\text{-}view$ != $null$ then
        return "cancer-type already exists!";
    else
        SaveCancerView(cancer_sample);
        return "cancer-type stored!";
else
    /* Detection request
    most_similar_view ← Back-End(cancer-sample);
    if $most\_similar\_view$ != $None$ then
        return most_similar_view.type();
    else
        return "Cancer type not found!";
```

Table 2 (resp. Table 3) reports the average performance achieved during the testing of the baseline DNN (resp. S). As we can see, the average performances achieved by S are evidently superior to those achieved by the baseline DNN.

Table 2: Baseline DNN average testing performance.

Accuracy	Precision	Recall	F1 score
0.7380	0.8499	0.7977	0.7879

Table 3: S average testing performance.

Accuracy	Precision	Recall	F1 score
0.8925	0.9760	0.9763	0.9761

This is even more evident if we look at the data reported in Table 4, which the accuracy achieved by both the models for each of the 16 cancer-type class. Notice that for 6 classes (Bladder Cancer, Bone Cancer, Breast Cancer, Cancer of Unknown Primary, Hepatobiliary Cancer, Non-Small Cell Lung Cancer) the baseline DNN shows performances superior to those achieved by S, while for the remaining 10 classes S proves to be more efficient. However, the maximum gap between the performance by the baseline DNN and that by S when the baseline DNN is better than S, i.e, $1.0000 - 0.9069 = 0.0931$ for the class Bladder Cancer, is lower of the the gap calculated in the opposite case, i.e., $0.6949 - 0.1818 = 0.5131$ for the class `Endometrial Cancer`.

Furthermore, the baseline DNN tends to overfit for the classes that have a higher number of instances, while the S network has a more stable behavior, trying to distribute the accuracy more uniformly among the various classes. This can be deduced from the performances achieved in the worst cases, which are much lower for the baseline DNN (0.1818 for the `Endometrial Cancer`) than for S (0.6949 for the `Endometrial Cancer`).

Table 4: Accuracy achieved by the baseline DNN and S for each of the 16 cancer-type class.

`Cancer type`	`DNN accuracy`	S `accuracy`
Prostate Cancer	0.8513	0.9605
Non-Small Cell Lung Cancer	0.9824	0.9524
Breast Cancer	1.0000	0.9392
Soft Tissue Sarcoma	0.3030	0.7037
Colorectal Cancer	0.8823	0.9581
Glioma	0.7391	0.8280
Hepatobiliary Cancer	0.9230	0.8666
Melanoma	0.8666	0.9318
Esophagogastric Cancer	0.7000	0.7900
Pancreatic Cancer	0.7000	0.8333
Bone Cancer	0.9166	0.9537
Cancer of Unknown Primary	0.8333	0.8314
Bladder Cancer	1.0000	0.9069
Ovarian Cancer	0.4285	0.7984
Head and Neck Cancer	0.5000	0.9296
Endometrial Cancer	0.1818	0.6949

3.2 Attention Feature Heatmaps

As explained in Section 2.1, one of the main goals of this work is to design a cancer-type detection system one that is explainability oriented. To this aim, in the structure of S has been integrated an attention layer used to produce special feature heatmaps which can help to visualize which parts of the input are more important for the detection. Figure 2 shows the feature heatmaps generated using the attention layer of S[5]. We remark that to facilitate the viewing and interpretation of the heatmaps, we have we have superimposed special dotted rectangles whose color is that indicated by the heatmap and the size is proportional to the intensity of the highlighted areas.

As we can see, for each cancer-type class C_i, the corresponding heatmap has size $750 \times |C_i|$ where 750 is the size of the input vector of the attention layer, and $|C_i|$ indicates the number of instances of C_i. The most evident aspect that emerges from the visualization of the heatmaps is that each class activates a specific set of features of the vector given in input to the attention layer. This allows to identify a sort of visual pattern extracted from the cancer mutational views given in input. However, it is important to underline that, in this preliminary version of the work, this explainability component still needs a lot of work so that it can be profitably used for analysis. What is missing at the moment is a correspondence between the areas highlighted in the heatmaps and the corresponding features in the view which in fact determine the activation of the various areas.

In the same way, however, it is important to underline how the production of visual information to support the analysis of this type of problem, as well as orienting the system towards the question of explainability, makes it open to the possibility of integrating Information Visualization (IV) techniques. IV techniques consist in computerized methods that involve selecting, transforming and representing data in a visual form that facilitates human interaction for analyzing and understanding the data (Tao et al., 2004). IV techniques have been used in many areas of Bioinformatics. Although they have been successfully used in many biological domains, such as structure visualization, expression profile analysis, sequence analysis, visualization of genome, pathway and hierarchical data, in our opinion the study of the specific patterns of somatic mutations in the major cancer types is still challenging. We believe that a system such as the one proposed in this paper, i.e., oriented towards an explainable and usable approach, although still incomplete and in a preliminary form, can provide interesting starting points for future work in this direction.

4 DISCUSSION AND CONCLUSION

Although the obtained results are interesting, there are some obvious limitations that need to be addressed.

The proposed method is a preliminary attempt to simultaneously satisfy explainability and usability needs when applying AI techniques in Bioinformatics. In our opinion, the potential in the use of feature maps, on which however to date there is insufficient evidence to demonstrate their effectiveness in terms of explainability, is amplified by the use of SNNs whose advantage in terms of usability is evident. However, we plan to use explainability techniques that can return a heatmap with respect to the input sequence, which is easier to interpret.

We used the term “view” and not “signature” as the latter was already introduced in the literature, and there are different methods to calculate them. We cannot consider what we obtained as a real “signature” since on the downloaded data set we only considered

[5] https://github.com/FLaTNNBio/few-shot-learning-for-cancer-detection/tree/master

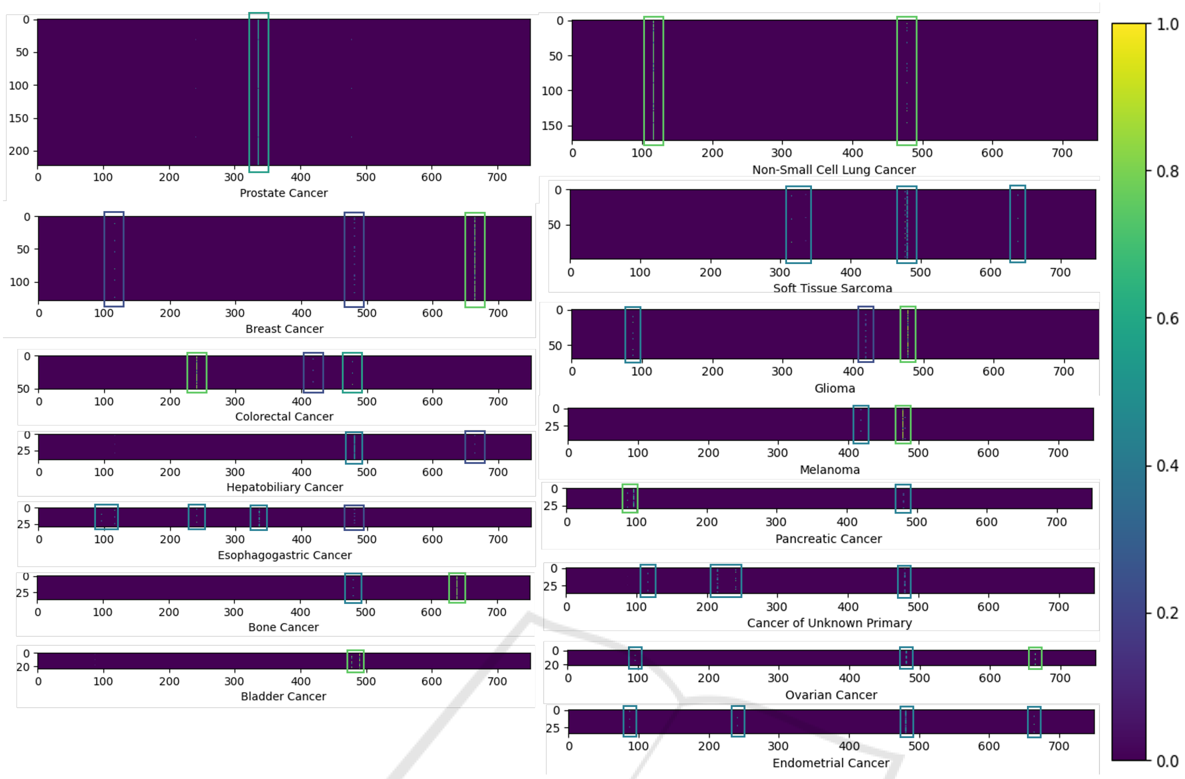

Figure 2: Attention feature heatmaps generated by S for each of the cancer-type classes.

molecular features (of which 13 variables with clinical information) and performed PCA.

Further investigations will be carried out with the aim to collect larger datasets to evaluate the performance of the model in a wider range of contexts.

REFERENCES

Bechar, M. E. A., Guyader, J.-M., El Bouz, M., Douet-Guilbert, N., Al Falou, A., and Troadec, M.-B. (2023). Highly performing automatic detection of structural chromosomal abnormalities using siamese architecture. *Journal of Molecular Biology*, 435(8):168045.

Bell, S. and Bala, K. (2015). Learning visual similarity for product design with convolutional neural networks. *ACM transactions on graphics (TOG)*, 34(4):1–10.

Chen, Y., Garcia, E. K., Gupta, M. R., Rahimi, A., and Cazzanti, L. (2009). Similarity-based classification: Concepts and algorithms. *Journal of Machine Learning Research*, 10(3).

Ciriello, G., Miller, M. L., Aksoy, B. A., Senbabaoglu, Y., Schultz, N., and Sander, C. (2013). Emerging landscape of oncogenic signatures across human cancers. *Nature genetics*, 45(10):1127–1133.

Jiao, W., Atwal, G., Polak, P., Karlic, R., Cuppen, E., Danyi, A., De Ridder, J., van Herpen, C., Lolkema, M. P., et al. (2020). A deep learning system accurately classifies primary and metastatic cancers using passenger mutation patterns. *Nature communications*, 11(1):728.

Kandoth, C., McLellan, M. D., Vandin, F., Ye, K., Niu, B., Lu, C., Xie, M., Zhang, Q., McMichael, J. F., Wyczalkowski, M. A., et al. (2013). Mutational landscape and significance across 12 major cancer types. *Nature*, 502(7471):333–339.

Karim, M. R., Beyan, O., Zappa, A., Costa, I. G., Rebholz-Schuhmann, D., Cochez, M., and Decker, S. (2021). Deep learning-based clustering approaches for bioinformatics. *Briefings in bioinformatics*, 22(1):393–415.

Kübler, K., Karlić, R., Haradhvala, N. J., Ha, K., Kim, J., Kuzman, M., Jiao, W., Gakkhar, S., Mouw, K. W., Braunstein, L. Z., et al. (2019). Tumor mutational landscape is a record of the pre-malignant state. *BioRxiv*, page 517565.

Lawrence, M. S., Stojanov, P., Polak, P., Kryukov, G. V., Cibulskis, K., Sivachenko, A., Carter, S. L., Stewart, C., Mermel, C. H., Roberts, S. A., et al. (2013). Mutational heterogeneity in cancer and the search for new cancer-associated genes. *Nature*, 499(7457):214–218.

Mathai, N. and Kirchmair, J. (2020). Similarity-based methods and machine learning approaches for target prediction in early drug discovery: performance and scope. *International Journal of Molecular Sciences*, 21(10):3585.

Narmatha, P., Gupta, S., Lakshmi, T. V., and Manikavelan, D. (2023). Skin cancer detection from dermoscopic images using deep siamese domain adaptation convolutional neural network optimized with honey badger algorithm. *Biomedical Signal Processing and Control*, 86:105264.

Tan, P.-N., Steinbach, M., and Kumar, V. (2016). *Introduction to data mining*. Pearson Education India.

Tao, Y., Liu, Y., Friedman, C., and Lussier, Y. A. (2004). Information visualization techniques in bioinformatics during the postgenomic era. *Drug Discovery Today: BIOSILICO*, 2(6):237–245.