Detection of the PIK3CA mutation in breast cancer

Ulysse Trin
EPITECH
ulysse.trin@hotmail.fr
&
Matteo Sammut
Paris-Dauphine
matteo.sammut@dauphine.eu

Abstract

Breast cancer is a complex disease that can be challenging to diagnose and treat accurately. The PIK3CA gene is known to play a critical role in the development of breast cancer, and identifying mutations in this gene is essential for effective treatment. However, the analysis of histological slides is time-consuming and requires manual annotation by anatomical pathologists, which can be a bottleneck in the diagnosis and treatment process.

To address this issue, we propose a novel weakly supervised learning approach that involves extracting smaller images (tiles) from histological slides and associating them with a single annotation of mutation presence or absence. We then use multiple-instance learning to intelligently aggregate these tiles and produce a global label, which reduces the noise associated with weakly supervised learning.

Our approach was evaluated on the public test set from Owkin Data Challenge, and we achieved an AUC of 0.73, significantly outperforming the logistic regression approach that produced an AUC of 0.6018. Our results demonstrate the effectiveness of our approach in accurately predicting the presence or absence of mutations, while also reducing the need for manual annotation by anatomical pathologists.

This approach has the potential to significantly improve the efficiency and accuracy of cancer diagnosis and treatment by enabling faster and more cost-effective analysis of histological slides. Our proposed method can potentially benefit patients by providing accurate diagnoses, reducing the time and costs involved in analysis, and aiding in the development of personalized treatment plans. By leveraging weakly supervised learning and multiple-instance learning, our approach has the potential to revolutionize the field of cancer diagnosis and treatment.

1. Introduction

Histopathology is a critical step in diagnosing and treating diseases, particularly in oncology. In breast cancer, one of the genomic alterations of particular interest is the PIK3CA mutation, which can be identified through the analysis of histopathology slides. However, the current method of detecting PIK3CA mutations requires technical and bioinformatic expertise that is not readily accessible in all laboratories, limiting the ability to provide personalized therapies for patients with better outcomes.

To address this challenge, automated solutions for detecting PIK3CA mutations have been developed. The Owkin Data Challenge proposed a weakly-supervised binary classification problem for detecting PIK3CA mutations directly from histopathology slides. The challenge dataset includes a set of tile images randomly chosen inside the tissue as .jpg files, the feature vectors extracted from each of the tiles using a pre-trained ResNet model, and metadata related to the original slide image. Metadata are provided as csv files, that contain the following columns: ”Sample ID”, ”Center ID” that indicates the hospital of origin and ”Patient ID”, the unique identifier of patients, as some patients may have several slides. Each patient is associated to a Matrix, which is of size N x 2,051, with N being the number of tiles for the given sample and 2.051 being the number of features extract from tiles. The MoCo v2 features have been extracted using a Wide ResNet-50-2 pre-trained on TCGA-COAD. The classification task involves predicting whether a patient has a mutation of the gene PIK3CA, directly from a slide. The output expected is a float number between 0.0 and 1.0, representing the probability of PIK3CA mutation for each slide.This approach alleviates the burden of obtaining locally annotated tiles and allows the use of extracted features to help people who do not have the computing resources or time to train directly from images.

MIL is a machine learning technique that maps multiple items to a single global label by predicting the presence of mutation if one of the pooled tiles exhibits a mutation pattern, and the absence of mutation if none of the tiles exhibit the pattern. In many real-life applications multiple instances are observed and only a general statement of the category is given. This scenario is called multiple instance learning (MIL) (Dietterich et al., 1997; Maron Lozano-Perez ´ , 1998) or, learning from weakly annotated data (Oquab et al., 2014). The problem of weakly annotated data is especially apparent in medical imaging (Quellec et al., 2017) (e.g., computational pathology, mammography or CT lung screening) where an image is typically described by a single label (benign/malignant) or a Region Of Interest (ROI) is roughly given. MIL deals with a bag of instances for which a single class label is assigned. Hence, the main goal of MIL is to learn model that predicts a bag label, e.g., a medical diagnosis. An additional challenge is to discover key instances (Liu et al., 2012), i.e., the instances that trigger the bag label.

The objective of this research paper is to propose an approach to detecting PIK3CA mutations directly from histopathology slides. The classification task involves predicting whether a patient has a mutation of the gene PIK3CA, directly from a slide. The extracted features can also help people who do not have the computing resources or time to train directly from images. Overall, the proposed approach has the potential to significantly improve the efficiency and accuracy of cancer diagnosis and treatment, enabling faster and more cost-effective analysis of histopathology slides.

2. Méthodologie

2.1. Multiple instance learning (MIL)

Problem formulation : In the classical binary supervised learning problem one aims at finding a model that predicts a value of a target variable, y $\in \{0,1\}$, for a given instance, x$\in \mathbb{R}^D$. In the case of the MIL problem, however, instead of a single instance there is a bag of instances, X = $\{x_1, ..., x_K\}$, that exhibit neither dependency nor ordering among each other. We assume that K could vary for different bags. There is also a single binary label Y associated with the bag. Furthermore, we assume that individual labels exist for the instances within a bag, i.e., $\{y_1, ..., y_K\}$ and y$_K$ $\in \{0,1\}$, for $k \in \{1, ..., K\}$. However, there is no access to those labels and they remain unknown during training. We can re-write the assumptions of the MIL problem in the following form: amsmath

$$Y = \begin{cases} 0 & \text{if } \sum_{k=1}^{K} y_k = 0 \\ 1 & \text{otherwise} \end{cases}$$

These assumptions imply that a MIL model must be permutation-invariant. Further, the two statements could be re-formulated in a compact form using the maximum operator: Y = $\max_k y_k$

Learning a model that tries to optimize an objective based on the maximum over instance labels would be problematic at least for two reasons. First, all gradient-based learning methods would encounter issues with vanishing gradients. Second, this formulation is suitable only when an instancelevel classifier is used. In order to make the learning problem easier, we propose to train a MIL model by optimizing the log-likelihood function where the bag label is distributed according to the Bernoulli distribution with the parameter $\theta(X) \in [0,1]$, i.e., the probability of Y = 1 given the bag of instances X.

MIL approaches : In the MIL setting the bag probability $\theta(X)$ must be permutation-invariant since we assume neither ordering nor dependency of instances within a bag. Therefore, the MIL problem can be considered in terms of a specific form of the Fundamental Theorem of Symmetric Functions with monomials given by the following theorem

Theorem 1 : A scoring function for a set of instances X, $S(X) \in \mathbb{R}$, is a symmetric function (i.e., permutationinvariant to the elements in X), if and only if it can be over, in decomposed in the following form:

$$S(X) = g\left(\sum_{\mathbf{x} \in X} f(\mathbf{x})\right)$$

where f and g are suitable transformations.

This theorem provides a general strategy for modeling the bag probability using the decomposition given in (3). A similar decomposition with max instead of sum is given by the following theorem (Qi et al., 2017).

Theorem 2 : For any $\varepsilon > 0$, a Hausdorff continuous symmetric function $S(X) \in \mathbb{R}$ can be arbitrarily approximated by a function in the form $g\left(\max_{\mathbf{x} \in X} f(\mathbf{x})\right)$, where $\max$ is the element-wise vector maximum operator and f and g are continuous functions, that is:

$$\left| S(X) - g\left(\max_{\mathbf{x} \in X} f(\mathbf{x})\right) \right| < \varepsilon$$

The difference between Theorems 1 and 2 is that the former is a universal decomposition while the latter provides an arbitrary approximation. Nonetheless, they both formulate a general three-step approach for classifying a bag of instances: (i) a transformation of instances using the function f, (ii) a combination of transformed instances using a symmetric (permutation-invariant) function , (iii) a transformation of combined instances transformed by f using a function g. Finally, the expressiveness of the score function relies on the choice of classes of functions for f and g.

In the MIL problem formulation the score function in both theorems is the probability $\theta(X)$ and the permutation-invariant function is referred to as the MIL pooling. The choice of functions f, g and determines a specific approach to modeling the label probability. For a given MIL operator there are two main MIL approaches:

(i) The instance-level approach: The transformation f is an instance-level classifier that returns scores for each instance. Then individual scores are aggregated by MIL pooling to obtain $\theta(X)$. The function g is the identity function.

(ii) The embedding-level approach: The function f maps instances to a low-dimensional embedding. MIL pooling is used to obtain a bag representation that is independent of the number of instances in the bag. The bag representation is further processed by a bag-level classifier to provide $\theta(X)$

It is advocated in (Wang et al., 2016) that the latter approach is preferable in terms of the bag level classification performance. Since the individual labels are unknown, there is a threat that the instance-level classifier might be trained insufficiently and it introduces additional error to the final prediction. The embedding-level approach determines a joint representation of a bag and therefore it does not introduce additional bias to the bag-level classifier. On the other hand, the instance-level approach provides a score that can be used to find key instances i.e., the instances that trigger the bag label. Liu et al. (2012) were able to show that a model that is successfully detecting key instances is more likely to achieve better bag label predictions. We propose to modify the embedding-level approach to be interpretable by using a new MIL pooling.

2.2. Attention-based MIL pooling

All MIL pooling operators mentioned in the previous section have a clear disadvantage, namely, they are predefined and non-trainable. For instance, the max-operator could be a good choice in the instance-based approach but it might be inappropriate for the embedding-based approach. Similarly, the mean operator is definitely a bad MIL pooling to aggregate instance scores, although, it could succeed in calculating the bag representation. Therefore, a flexible and adaptive MIL pooling could potentially achieve better results by adjusting to a task and data. Ideally, such MIL pooling should also be interpretable, a trait that is missing in all operators mentioned in later section.

Attention mechanism: We propose to use a weighted average of instances (low-dimensional embeddings) where weights are determined by a neural network. Additionally, the weights must sum to 1 to be invariant to the size of a bag. The weighted average fulfills the requirements of the Theorem 1 where the weights together with the embeddings are part of the f function. Let H = $\{h_1, ..., h_K\}$ be a bag of K embeddings, then we propose the following MIL pooling:

$$\mathbf{z} = \sum_{k=1}^{K} a_k \mathbf{h}_k$$

where:

$$a_k = \frac{\exp\left\{\mathbf{w}^\top \tanh\left(\mathbf{V}\mathbf{h}_k^\top\right)\right\}}{\sum_{j=1}^{K} \exp\left\{\mathbf{w}^\top \tanh\left(\mathbf{V}\mathbf{h}_j^\top\right)\right\}},$$

where w $\in \mathbb{R}^{L \times 1}$ and $\mathbf{V} \in \mathbb{R}^{L \times M}$ are parameters.

Moreover, we utilize the hyperbolic tangent $\tanh(\cdot)$ element-wise non-linearity to include both negative and positive values for proper gradient flow. The proposed construction allows to discover (dis)similarities among instances.

The proposed MIL pooling corresponds to a version of the attention mechanism (Lin et al., 2017; Raffel Ellis, 2015). The main difference is that typically in the attention mechanism all instances are sequentially dependent while here we assume that all instances are independent. Therefore, a naturally arising question is whether the attention mechanism could work without sequential dependencies among instances, and if it will not learn the mean operator.

Gated attention mechanism : Furthermore, one should notice that the tanh$(\cdot)$ non-linearity could be inefficient to learn complex relations. Our concern follows from the fact that tanh(x) is approximately linear for x = $\{-1, 1\}$, which could limit the final expressiveness of learned relations among instances. Therefore, we will additionally use the gating mechanism (Dauphin et al., 2016) together with tanh$(\cdot)$ non-linearity that yields:

$$a_k = \frac{\exp\left\{\mathbf{w}^\top \left(\tanh\left(\mathbf{V}\mathbf{h}_k^\top\right) \odot \mathrm{sigm}\left(\mathbf{U}\mathbf{h}_k^\top\right)\right)\right\}}{\sum_{j=1}^{K} \exp\left\{\mathbf{w}^\top \left(\tanh\left(\mathbf{V}\mathbf{h}_j^\top\right) \odot \mathrm{sigm}\left(\mathbf{U}\mathbf{h}_j^\top\right)\right)\right\}},$$

where $\mathbf{U} \in \mathbb{R}^{L \times M}$ are parameters, $\odot$ is an element-wise multiplication and sigm(s.) is the sigmoid non-linearity.

The gating mechanism introduces a learnable non-linearity that potentially removes the troublesome linearity in $\tanh(\cdot)$.

Flexibility : In principle, the proposed attention-based MIL pooling allows to assign different weights to instances within a bag and hence the final representation of the bag could be highly informative for the bag-level classifier. In other words, it should be able to find key instances. Moreover, application of the attention-based MIL pooling together with the transformations f and g parameterized by neural networks makes the whole model fully differentiable and adaptive. These two facts make the proposed MIL pooling a potentially very flexible operator that could model an arbitrary permutation-invariant score function.

Interpretability : Ideally, in the case of a positive label (Y = 1), high attention weights should be assigned to instances that are likely to have label y_k = 1 (key instances). Namely, the attention mechanism allows to easily interpret the provided decision in terms of instance-level labels. In fact, the attention network does not provide scores as the instance-based classifier does but it can be considered as a proxy to that. The attention-based MIL pooling bridges the instance-level approach and the embedding-level approach. From the practical point of view, e.g., in the computational pathology, it is desirable to provide ROIs together with the final diagnosis to a doctor. Therefore, the attention mechanism is potentially of great interest in practical applications.

3. Content

3.1. Model Architecture

We proposed a MIL model parametrized with an attention-based pooling layer and a neural network. The following graph (ie Figure 1) gives a better understanding of the model architecture.

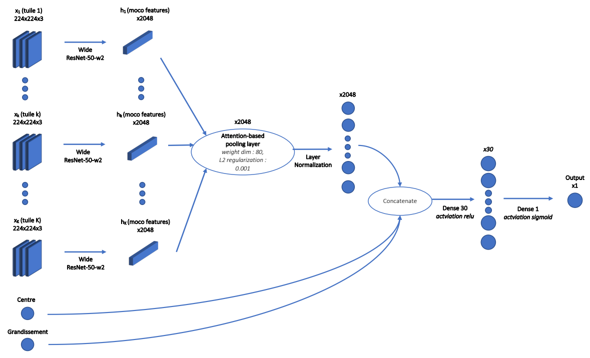


Figure 1. Model Architecture

A Wide RestNet-50-w2 is first used to find an embedding for our 1000 tiles. This step helps the model by decreasing its complexity : tiles of dimension 224x224x3 are now representing in a 2048 dimension vector space. How ever, embbedings are given such that we can not fine tune the ResNet. The attention-based pooling layer with weights of dimension 80 is used to obtain a bag representation of the embedded instances before passing through a Normalization Layer. At this point we introduce the center and the magnification level by concatenation. The bag representation is further processed by the neural network which acts as the bag-level classifier to provide $\theta(X)$. The later is simply composed of one hidden dense layer with 30 neurones and a ReLU activation, followed by a one dense output layer with sigmoid activation.

One should notice that instance coordinates are neither used in the attention layer nor in the neural network. The reason for that is that our MIL problem assumptions states that the model should be permutation-invariant. In other words, the bag of instances should exhibit neither dependancy nor ordering among each other. Integrating the coordinates would contradict those assumptions. On top of that, in our context, it seems that integrating those variables could add bias in our model.

Choosing a neural network for the bag-level classifier is quite imposed here. Indeed, the tuning of the attention-based layer parameters must be part of a stochastic gradient descent algorithm, and hence is not compatible with a decision-tree based model for example. One solution could be splitting the training process into two parts : first training a simple logistic classifier after the attention-based layer to fix the a_k coefficients, than training any Machine Learning classifier on the attention layer outputs. We have tried this approach by using Random Forest or SVM algorithms, but with no success, with respectively 0,63 and 0,53 of AUC.

The proposed model performed well with an AUC of 0.73 on the public test. Performance is resumed in the following table.

Metric	Train AUC	Test AUC	Public Test AUC
AUC	1	0,75	0,73
Binary cross-entropy	0.00708	1.1591	

Figure 2. Performance table

Our model is over-fitting and we dedicate the latter section to this subject. Comparing to the benchmark which was performing 0.6018 on the public test, we significantly improve the score. Furthermore, regarding to our rank on this challenge, which is second on public test, we find convenient our approach.

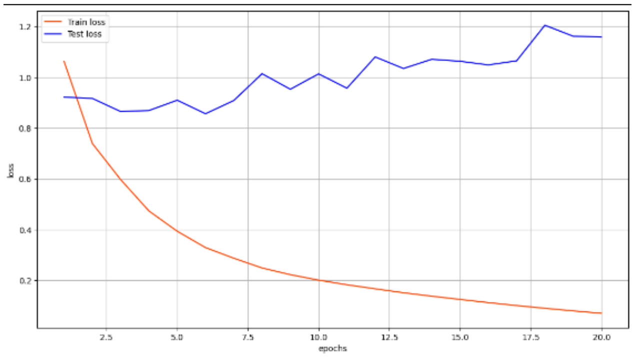


Figure 3. Evolution of Binary cross-entropy during training process

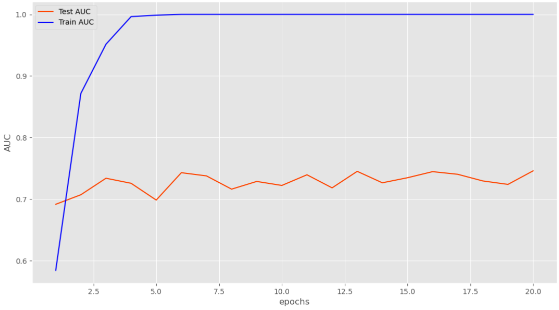


Figure 4. Evolution of AUC during training process

Figures 3 and 4 gives a better look of the training process. The algorithm is performing very well on the training set : binary cross-entropy become non significant while AUC reach the 1 perfection. On the test set loss and AUC are slowly increasing during epochs.

3.2. Model Generalization

The main challenge of this problem is about preveting the model from overfitting. Each 1000 instances are represented in a vector space of dimension 2048, but we have only 344 observations to train the model. The motivation of choosing a relatively simple neural network was linked to this difficulty.

One attempt has been to try to reduce the dimension of Mono features. SVD and PCA analysis has been made on the matrices of all instances in our train set.

On those two graphics, we can see that trying to reduce Mono features dimension with a linear method should be complicated. Singular Value Decomposition shows that there is not much multicollinearity between them. Indeed, even if eigen values curve in Figure 1 is first decreasing rapidly, each eigen vector seems important. In Figure 6, cumulative explained variance has a hard time becoming significant. An interesting alternative here could be t-SNE algorithm, a nonlinear dimensionality reduction method. Specifically, it models each high-dimensional object by a two- or three-dimensional point in such a way that similar objects are modeled by nearby points and dissimilar objects are modeled by distant points with high probability.

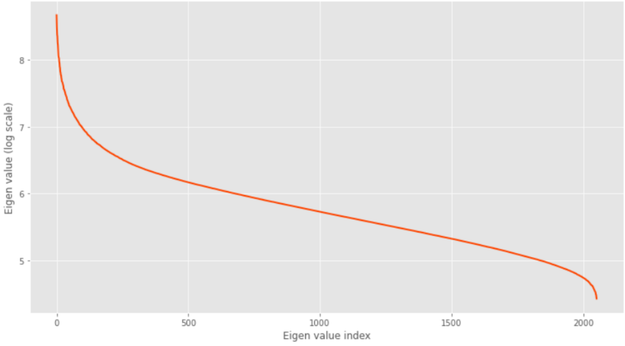


Figure 5. Singular Value Decomposition

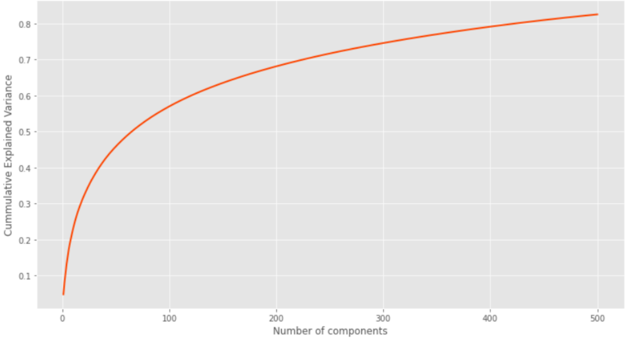
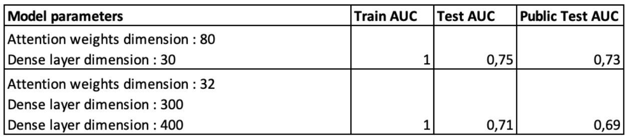


Figure 6. PCA - Cumulative explained variance

Another very important source of poor generalization is the disparity of data between centers. Indeed, the data for this challenge (training and testing) come from 5 different centers. The training set contains data from 3 centers, centers 1,2 and 5 represented respectively with 138, 92, and 114 observations. The test set contains data from the remaining 2 centers, but he don't have access to this variable in the test set. One solution considered was to impute those values using k-Nearest Neighbors algorithm. How ever the curse of dimension led us to a very time consuming process. At the end, no intelligent proposal has been found, and we decided to impute test centers with the value 5.

The main improvement to reduce over-fitting has been to assign less weight to the neural network and to increase the dimension of the attention-based pooling layer. Our interpretation is that we want our model to focus on finding dissimilarities between instances, weather than on processing the exact value of Moco features. The following table shows the impact of the proposed solution to improve the generalization of the model.



Model parameters	Train AUC	Test AUC	Public Test AUC
Attention weights dimension : 80 Dense layer dimension : 30	1	0,75	0,73
Attention weights dimension : 32 Dense layer dimension : 300 Dense layer dimension : 400	1	0,71	0,69

Figure 7. Increasing attention weights performance

We've compared two models : the one retained as well as another one on which we conducted a lot of research, which is composed of an attention layer of dimension 32 and 2 dense layers of 300 and 400 neurons respectively. Considering the second model, in addition to the lower performance, the difference in performance between our test set and the public test is more pronounced than for the retained model.

4. References

Andrews, Stuart, Tsochantaridis, Ioannis, and Hofmann, Thomas. Support vector machines for multiple-instance learning. In NIPS, pp. 577–584, 2003.

Quellec, Gwenole, Cazuguel, Guy, Cochener, Beatrice, and Lamard, Mathieu. Multiple-instance learning for medical image and video analysis. IEEE Reviews in Biomedical Engineering, 2017.

Raffel, Colin and Ellis, Daniel PW. Feed-forward networks with attention can solve some long-term memory problems. 2015.
Raykar, Vikas C, Krishnapuram, Balaji, Bi, Jinbo, Dundar, Murat, and Rao, R Bharat. Bayesian multiple instance learning: automatic feature selection and inductive transfer. In ICML, pp. 808–815, 2008.

Quellec, Gwenole, Cazuguel, Guy, Cochener, Beatrice, and Lamard, Mathieu. Multiple-instance learning for medical image and video analysis. IEEE Reviews in Biomedical Engineering, 2017.