Learning symbolic features for rule induction in computer aided diagnosis

Sebastijan Dumančić, Antoine Adam and Hendrik Blockeel

Department of Computer Science, Katholieke Universiteit Leuven, 3001 Heverlee, Belgium
{sebastijan.dumancic,antoine.adam,hendrik.blockeel}@cs.kuleuven.be

Abstract. In computer aided medical diagnosis (CAD), interpretability of learned models is an important concern. Unfortunately, the raw data used to train a model are often in subsymbolic form (for instance, images), which makes the application of symbolic learning methods difficult. Construction of symbolic features can bridge the gap between the symbolic and subsymbolic level. This paper presents a case study of how ILP learners can be used to learn models from visual data by using a feature construction step. The resulting model has an accuracy comparable to that of previous models, but better interpretability.

Keywords: computer aided diagnostics, inductive logic programming, deep learning, symbolic feature learning

1 Introduction

Ever more frequently, and in an increasing variety of domains, machine learning based models assist humans in decision making. For example, massive online retailers use recommender systems to suggest products to their customers, search engines rank the information by its relevance [19] , while computer vision techniques are used in biology for tracking cells (or other objects) and analyse them [20,21]. The accuracy of learned models often matches or even exceeds that of human experts. Despite this, learned models are not always suitable for usage, as for some domains interpretability of the model is essential. 'Black box' models which base their decisions on computations not understandable to humans are obviously not suitable for this kind of applications.

An example of such a domain is computer aided diagnosis (CAD) in medicine. Consider the "anti-nuclear antibodies" (ANA) test, which is used to diagnose autoimmune diseases. The diagnosis is based on visually identifying certain staining patterns of cells, which are further mapped to a specific disease. This decision is known to be subjective [28]; it depends heavily on the expertise of a physician, and on the varieties of reading systems and optics. Example patterns are shown in Figure 1. Recently, this problem attracted a lot of attention in machine learning and computer vision community, and was presented as a contest at the *International Conference on Pattern Recognition 2012*. The contest resulted in

a numerous proposed solutions compared in [1,4], leading to the state-of-the-art solution by Xu et al. [25]. However, unless the automated system can explain its decision, physicians are unlikely to blindly trust it, given the subjectiveness of the decision process.

In this paper, we want to break open the black box. In the case when automated reasoning procedures support a diagnostic procedure, we emphasize that usefulness of automated diagnostic procedure is limited with its ability to explain its decision. We illustrate the need for interpretable models on the following scenario. Consider the case in which the task of a CAD system is to double-check diagnosis made by a physician. The CAD systems are clearly useful in a case where a physician is not completely certain in his/her decision. In the case of uncertain diagnosis, if the physician is presented with a conflicting diagnosis from the CAD system, having just the decision made by the system is obviously not very useful. However, if the CAD system would provide the physician with an explanation together with the decision, such as *"The cell A is classified as* ***nucleolar*** *because it has circular shape and a few bright organelles"* (see figure 1), that could potentially clarify the physician's decision by identifying aspects that the physician (or the CAD system) have missed or misinterpreted. This scenario motivates the work in this paper.

We propose to learn interpretable models from raw image data by introducing a feature construction step that extracts symbolic features using subsymbolic learning. We use the above mentioned ANA test as a case study. The system we present here consists of three steps. First, individual cells are segmented from the images obtained by Indirect immunofluorescence imaging. Second, each segmented cell is mapped to several symbolic features describing their visual properties. Third, we learn an interpretable model by means of Inductive Logic Programming (ILP) and Statistical Relational Learning (SRL) methods. Additionally, we make use of the specific settings of the ANA test that allows a collective classification [24], a perspective that other solutions for this problem haven't investigated yet. To our knowledge, this is the first system (for this application domain) that covers the whole path from image to decision (including segmentation), and also the first that yields interpretable models.

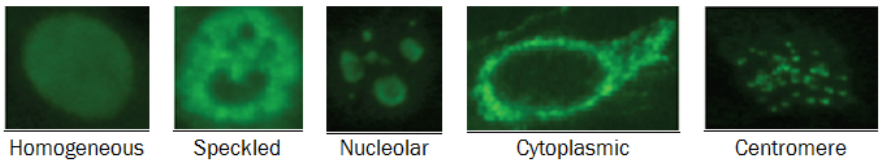


Fig. 1: Examples of HEp-2 staining patterns

The rest of the paper is structured al follows: section 2 discusses related work in this area; section 3 explains our approach; section 4 summarizes the experiments and results, while section 5 concludes the paper.

2 Related work

As it was already mention, this problem has been presented as a contest at the International Conference on Pattern Recognition 2012. The summary of the results is provided in [1]. To our knowledge, the best performance so far was reported by Xu at al [25]. The author have used a Linear Local Distance Coding method to extract the features, which were further fed into a liner Support vector machine. (description of approach? Not interpretable, very complex representation)

**** Similar approaches - can't find anything focused on interpretable feature construction(?) *****

2.1 (Probabilistic) Logic Learning

****Discuss ILP and SRL (MLN)****

2.2 Deep learning

A *Deep belief network* is an instance of a *Deep learning* methods [10]. Deep learning is a relatively new approach in machine learning which is often referred as Representation learning. It is built upon Artificial neural networks and imitates the human brain in representing data. The motivation for learning representation is quite clear – the form in which data is represented is important. The success of our classifier depends on the quality of data used for training.

The Deep belief network can be seen as a multi-layer generative model where each layer consists of multiple nodes, similar to a neural network. The first layer, often referred as the *visible layer*, represents the raw input data, while every higher-level layer is referred to as the *hidden layer*. It is trained is two steps - first unsupervised then supervised.

For the unsupervised phase, *Restricted Boltzmann machines* [11] are used. The Restricted Boltzmann machine is a generative energy-based model that shares the parametrization with the neural network. The Restricted Boltzmann machines are trained by maximizing the probability of data:

$$\arg\max_{W} \prod_{v \in V} P(\mathbf{v}) \tag{1}$$

where $\mathbf{v}$ represent a raw data, or visible layer of pixels when trained on images, while probability is represented as an energy

$$P(\mathbf{v},\mathbf{h}) = \frac{e^{-\mathtt{E}(\mathbf{v},\mathbf{h})}}{\mathtt{Z}} \tag{2}$$

$$\mathtt{E}(\mathbf{v},\mathbf{h}) = -\mathbf{b}^T\mathbf{v} - \mathbf{c}^T\mathbf{h} - \mathbf{h}^T\mathbf{W}\mathbf{v} \tag{3}$$

where $\mathbf{v}$ represents raw data, $\mathbf{h}$ response of hidden units, $\mathbf{b}_i$ and $\mathbf{c}_i$ are the offsets associated with a single element from $\mathbf{x}$ or $\mathbf{h}$ and W_{ij} are weights associated with each pair of units from different layers. $\mathtt{Z}$ is a normalization factor. After unsupervised training, the deep belief network is *fine tuned* by backpropagation [12].

3 Proposed approach

The approach we propose is illustrated in figure 2. Our approach closely follows the steps taken by a physician and consists of three steps. First, individual cells are segmented from the images obtained by Indirect immunofluorescence imaging. Second, each segmented cell is then mapped to several symbolic features describing their visual appearance. This mapping is partially hard-coded and partially learned using Support vector machine and Deep learning. Finally, we use a relational machine learning system to learn an interpretable model on these extracted features.

An important thing to notice here is that the original images contain only a single antibody type, so a physician making a diagnosis considers all the antibodies from one image collectively, not individually. Solutions proposed so far use this property only on a level of aggregation - the class of an image is chosen to be the most frequent single cell class predicted by a classifier. However, the collective classification in statistical relational learning takes into account that information - it collectively classifies all objects related to each other, or in this case, all cells contained in one image. Thus, by taking into account all cells in the image, it can improve the performance even on the individual cell level.

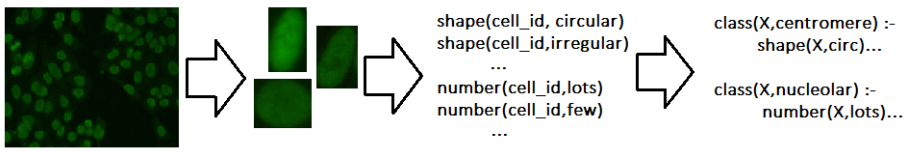
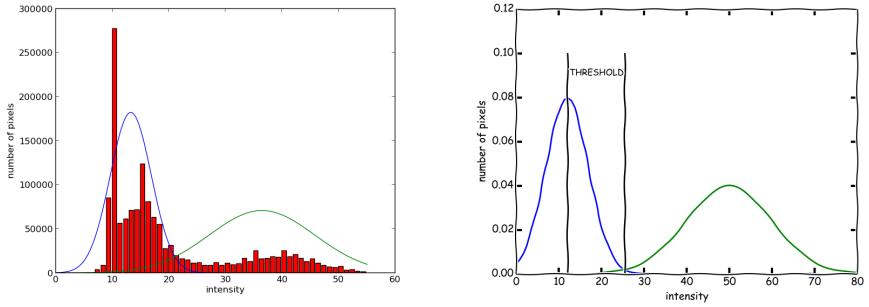


Fig. 2: Illustration of a process

3.1 Segmentation

Our segmentation procedure is motivated by the problems encountered by previous work mentioned in Section 2. Time-dependent cell colouring procedure causes the main problem of unequal intensity of cell across a single image. This poses a main difficulty in segmenting them. Our methods starts with an observation that, although cell express different intensities across image, the background is always constant and darker compared to cells. The major assumption taken here is that each image histogram (a distribution of grayscale colors or intensities across an image) can be segmented in two distinct parts - one representing the background and the second one representing the cells, as illustrated in Figure 3a.

Thus, the first step is to determine the background color. As the major part of an image is dark background, we identify a narrow peak in the image histogram (see figure 3a), and declare as background everything with the grayscale intensity less or equal to the peak. Next, a region growing algorithm finds a large connected region with this color; this large region is considered to be the background. Region growing extends current region by examining the pixels on the border of a current region - if a candidate pixel satisfies a predefined criteria, it is added to the region.



(a) (b)

Fig. 3: Illustration of the assumption made by our approach. Figure *(a)* represents an image histogram example (in red) and its approximation with two Gaussian functions (blue and green). Figure *(b)* illustrates the threshold estimation for the region growing algorithm.

The examination proceeds iteratively until no new pixels can be added to the region. We adopt a criteria based on a difference in grayscale intensity between the candidate pixel and the current region:

$$\text{criteria}(x) = \begin{cases} 1 & \text{if } x - \mu_{region} \leq t \\ 0 & \text{if } x - \mu_{region} > t \end{cases}$$

where x represents the intensity of a candidate pixel, μ_{region} is the mean intensity of the pixels belonging to the region and t is the predefined threshold. The threshold t is an image-dependent parameter and it is not possible to define a value that will suite every image. To estimate the threshold parameter, we follow the assumption stated above and approximate the histogram with two Gaussian components. This gives us an estimate of distributions across an image and allows us to estimate the parameter adaptively. We define t to be equal to the difference between the point x where $p(x|\mu_1, \sigma_1) = p(x|\mu_2, \sigma_2)$ and the mean intensity of the Gaussian with a lower mean μ_1:

$$t = |\mu_1 - x_{p(x|\mu_1,\sigma_1)=p(x|\mu_2,\sigma_2)}|,$$

as illustrated in figure 3b.

This provides us with a good estimation of the cell locations on image. Unfortunately, a lot of cells still overlap. To cope with this problem, we incorporate two standard image analysis methods to refine the borders and split the cells. First, the Hough transform [7] is used to identify roughly circular objects formed by non-background pixels. The Hough transform is a generalized voting scheme which detect circles, or any shape, by inverting the process; instead of detecting features, the method allows every feature (usually edges) to vote for every combination of parameters that could have generated the feature. Every local maxima in the voting space is then considered as a detected object. Finding roughly circular objects has the advantage that when two or more cells touch, the individual cells may still be identified. We discard every detected circle not originating in the non-background area of the image. Having candidate cells generated, each circle is further evolved to tight-fitting shape using Morphological Snakes [8]. Morphological Snakes are energy-based segmentation method that represents the inner and outer appearance of objects as energy functions and tries to minimize them. for a detailed overview of both methods, we refer to [7] and [8]. These shapes are our segmented objects.

3.2 Symbolic feature learning

The next step is the key to the interpretability of our final model. To goal of this step is to extract a meaningful features from raw pixel representation, which will be further used as an input by the classifier. However, we do not want any features to be extracted. We want features that are understandable by physicians. Therefore, we extract features that are used in the medical literature. As the ANA test is based on visual interpretation, we limit the features to the features that describe visual properties of cells. These features are extracted by examining medical literature [26,27] and detecting visual properties physicians use to describe the cells. These features and their possible values are the following:

- **shape**: circular, irregular
- **fluorescence intensity level**: positive, intermediate
- **structure**: homogeneous, speckled
- **organelle type**: dark, bright, neutral
- **organelle number**: none, few, lots
- **texture**: smooth, sparkly, blob
- **mitotic cell type**: bright middle, dark middle, neutral, speckled (for now, identified manually)

None of these features have a straightforward definition; they are based on subjective assessments by humans. We therefore learn, from the available data, a model that can predict the value of each feature from a cell image. For all features except Shape and Fluorescence Intensity Level (called Intensity from now on), we train a Deep Belief Network [10], as they are known to work well for identifying visual properties of images. One network per feature is learned, with the feature values as classes, as illustrated in figure **??** (figure TODO).

Shape and Intensity are learned in a different way, because deep learning seems unnecessarily complex for them:

Shape: Visual shape classification is a well-studied topic in computer vision and methods suitable for our goal already exist. We adopt the following method, motivated by Belongie et al [17]. Each individual cell image is divided into 4-by-4 blocks, and for each block, the proportion of pixels inside the extracted segment is calculated. A support vector machine [9] with radial basis function (RBF) kernel is next trained, using these 16 proportions as input features.

Intensity: This describes the *clarity* of the cells in an image. Determining the fluorescence intensity level is a separate task in the ANA workflow. The medical literature does not provide a precise definition for it, only a provisional ranking of four possibilities [28], described in terms of how easy is to distinguish cells from the background. All approaches mentioned in Section 2 suggest to recognize only two classes - *positive* or *high* when cells are clearly distinguishable from the background, and *intermediate* when it is difficult to distinguish cells from the background.

Our method to estimate fluorescence intensity level works as follows. Recall from Section 3.1 that each image histogram could be divided in two parts, one for the background and one for the cells, by fitting a mixture of two Gaussians to it. In images with high intensity, the two components should be well separated from each other, while in images with intermediate intensity they should be relatively close. To classify cells as having high or intermediate fluorescence intensity level, an SVM with RBF kernel is trained that uses the mean and variance of both fitted Gaussians as inputs.

3.3 Learning interpretable models

The symbolic representation obtained in the previous step is crucial for building an interpretable model. Since interpretability is important, we chose to compare several methods that use first-order logic or a set of rules as representation - a rule learner RIPPER [16] that naturally support interpretable rules; Inductive logic programming, that is FOIL [14] and Aleph [15], that learns the rules in forms of logical theories; and Markov logic networks [22] that combines probabilities with logic.

Additionally, we evaluate how collective classification in relational systems can improve the performance in classification. For ILP systems, we simply add a predicate that takes two cell identifiers as an input and returns true if the two cells are located on the same image. For Markov logic network, we test two approaches. First, we initialize the Markov network with the rules learned by Aleph. Second approach is inspired by topic models [29]. This model is constructed without any explicit domain knowledge incorporated in the structure of MLN. Thus, we define the following rules that describe the domain:

$$\text{HasProperty}(x, +p) => \text{Class}(x, +c)$$

$$\text{Class}(x, p),\ \text{SameImage}(x, y) => \text{Class}(y, p)$$

$$\text{SameImage}(x, y) => \text{SameImage}(y, x).$$

The first rule states that if a cell x has a certain property p, then it belongs to the class c. The $+$ in the formula means that we have such formula for each combination of a property p and a class c, where each formula will be assigned with a different weight. The second formula state that the cells at the same image should belong to the same class. Note that this does not mean all cells from an image will be classified as the same class, because the rule can be broken if evidence supports some other class. The last formula states the commutative definition of the predicate SameImage. For defined formulas, we learn their weights in the discriminative manner [23].

4 Experiments

For our experiments, we use the HEp-2 dataset published for the International Conference on Pattern Recognition 2012[1]. This dataset contains raw images as well as separate individual cells extracted from them.

As a first preprocessing step, we have manually annotated the individual cells with values for the features. These manual annotations are considered the ground truth for the feature values.

Next, we have constructed two knowledge bases with symbolic descriptions of cells, using the features defined above:

[1] `http://mivia.unisa.it/hep2contest/index.shtml`

– TRUE: each data element describes one of the individual cells identified in HEp-2, using the manually determined feature values
– SYS: each element describes a cell automatically extracted using the segmentation procedure described above; the feature values are the predictions made by a learned model, as explained above

SYS can be considered a noisy version of TRUE; its description may be incorrect, due to imperfect feature prediction models. The segmentation step still requires human intervention; the images contain artefact and mitotic cells that we are not able to detect automatically at the moment. The artefact are removed manually, while the type of mitotic cells appearing on an image is further used as a feature. To allow comparison between ground truth and learned features, cells that remain overlapping after the segmentation step are split manually.

Finally, we have learned interpretable rules from these tables.

In the following, we discuss experiments that aim at evaluating: (1) how well SYS approximates TRUE; (2) how well individual cells can be classified based on the information in TRUE and in SYS, using information about the cell alone; (3) the same, using collective classification methods.

4.1 TRUE versus SYS

Table 1 hows how accurately the features can be predicted using our learned models. The features are obtained with 10-fold cross validation, as a predictions made on the test fold. Here we present only the accuracy as the output of this step is an input for the rule induction. The Deep belief network proves to be successful approach to model the features.

Table 1: Performance of symbolic feature learning

Feature	shape	intensity	structure	number of organelles	organelle type	texture
Accuracy	98.68	96.21	91.4	89.7	93.78	90.22

4.2 Classification of cells based on their own properties

Table 2 shows the classification accuracy when learning from TRUE. It confirms that the proposed symbolic representation contains enough information to allow for an accurate classification of cells, and that a small number of rules suffices.

We show accuracy of each model, together with the precision and recall values. Each row in the table represents the precision and recall values for a specific cell type. We present both precision and recall to show that induced rules cover accurately each cell type.

The most important result here is that these features are sufficient for accurate classification, but offer interpretable and explainable results. As the rules

induces for this settings turned out to be propositional, we compare the performance to the propositional learning RIPPER [16], which yields the best overall performance in this setting.

Table 2: Performance of the classifiers without relations included

	$\texttt{MLN}_{topic}$		RIPPER		FOIL		Aleph	
pattern	Prec	Rec	Prec	Rec	Prec	Rec	Prec	Rec
homogeneous	93.41	94.54	93.48	100	92.15	100	100	69.69
nucleolar	88.43	88.79	94.65	95.44	97.10	88.78	100	68
centromere	93.59	94.11	98.24	93.54	100	92.37	100	93.22
cytoplasmatic	100	93.57	96.33	96.33	96.46	90.9	100	100
speckled	93.30	93.30			98	94	99	92
accuracy	93.05		94.50		93.30		84.06	
#rules	128		10		14		18	

Table 3 lists classification performance when learning from SYS. Performance drops slightly, as compared to TRUE, but not dramatically. This shows that even with the noisy information that is inherent to automatic feature extraction, quite accurate classification can be obtained. Most importantly, the performance is still comparable to the state-of-the-art reported by Xu et al. [25] which equals 95.59 %.

Table 3: Performance of the classifiers on data obtained by DBN

	$\texttt{MLN}_{topic}$		RIPPER		FOIL		Aleph	
pattern	Prec	Rec	Prec	Rec	Prec	Rec	Prec	Rec
homogeneous	92	90.60	92.94	99.70	94.36	76.06	99.14	70.51
nucleolar	82.81	78	86.40	92.95	100	80.91	100	60
centromere	90.93	92.71	94.67	89.64	98.48	90.78	99.39	91.57
cytoplasmatic	82.64	91.74	98.02	90.83	99	91.74	98.98	90.74
speckled	87.79	87.79	90.91	93.54	97.68	90.66	99.04	74.58
accuracy	88.31		91.3		87.16		76.57	
#rules	128		10		23		22	

4.3 Classification of cells using collective classification

Using relational learners allows us to incorporate information about other cells located on the same image. The important thing to notice here is the effect this

setting has with regards to ILP and SRL. Within ILP, it allows a lookup - if a cell can not be classified only by its properties, the it can be classified by using the information about the other cells contained in the same image. Within SRL, it uses the information about all cells simultaneously in both inference and learning. The results are presented in table 4.

The obtained results show that the information about other cells clearly helps in ILP settings. The performances of both FOIL and Aleph are significantly improved. Moreover, while performing the experiments we notice that it not only improves the performance, but also provides more stable decision - we have not noticed a difference between performances on TRUE and SYS datasets. However, when uncertainty is introduced in SRL setting, collective classification degrades the performance, which is an unexpected results. While performing the experiments, we notice that collective classification on TRUE dataset improves the performance (93.08 % without the information compared to 96% with the information), however when using the features obtained by the system, introducing the information seems to not help. Even more, the MLN approaches on the features obtained by the system performs worse than pure logic approaches. The reason behind this might be that the structure for MLN is learned by the ILP approach, while learning it specifically for MLN might be beneficial. This remains for the future work.

Table 4: Performance of the classifiers in collective classification settings

	$\texttt{MLN}_{topic}$		$\texttt{MLN}_{aleph}$		`FOIL`		`Aleph`	
pattern	Prec	Rec	Prec	Rec	Prec	Rec	Prec	Rec
homogeneous	74.67	68.78	73.25	100	98.20	99.39	99.14	79.93
nucleolar	72.39	51.08	100	76.26	100	96.68	100	77.08
centromere	93.59	94.15	100	88.36	97	99.49	99.43	98.31
cytoplasmatic	60.15	73.39	92.1	100	99.06	97.24	99.03	95.37
speckled	70.17	65.31	91	91.01	99.27	98.08	99.49	95.20
accuracy	71.20		89		97.32		89.28	

Finally, we compare our approach with the most successful approaches from the ICPR contest [1], in the same settings. The accuracy on both individual cell and image level is presented in table 5. The results suggest that both ILP and MLN approaches with collective classification outperform other approaches on the cell level, but do not provide the benefit for image level accuracy. However, these results are very pleasing as the focus of this work is not on the accurate methods, but on interpretable decision instead. By using understandable features that are more restrictive than numerical features, especially in context of image analysis, the system provides an explanation together with the decision, and retain the same performance.

Table 5: Comparison in the ICPR settings

	FOIL	MLN_{aleph}	Nosaka	Xiangfei	Kuan
cell level	70.07	79.54	68.7	66	62
image level	78.57	85.71	85.71	78.57	78.57

At the end, we analysed the rules induced by the systems. The examples are provided in figure 4 (TODO: add better rule examples, more amplifying the relational structure). By analysing induced rules, we noticed rules tend to capture the most indicative features about the cell types. In the examples in figure 4, the rules that cover most of the *cytoplasmatic* cells identifies the cells by its irregular shape, which is indeed the property specific for this type of cells. The *centromere* cells, on the other hand, are the only ones that contain a large number of organelles in the cell.

We notice that the rules induced by all systems are highly redundant - the rules cover overlapping regions of examples. This property might be one of the reasons why mislabelled features do not affect significantly compared to the ground truth performance, together with the information about the cells from the same image.

Also, we noticed that the mitotic cells play an important role in the decision. Out of 14 rules induced by FOIL at TRUE dataset, 8 of them use the information about the mitotic cells (on SYS, 17 out of 23). With Aleph, mitotic cells appear in 14 out of 18 rules on TRUE dataset, and 17 out of 22 on SYS. The information about mitotic cells might be one of the reasons to have quite accurate classification even with the noisy features.

**** analyze the weights in topic model ****

```
cytoplasmatic(X) :- sameImage(X,Y) ^ shape(Y, irregular)
cytoplasmatic(X) :- texture(X, blob)
centromere(X, centromere) :- organelles_type(X, bright), organelle_number(X, lots)
homogeneous(X) :- not organelle_number(X, lots), mitotic_cells(X, bright_middle)
speckled(A) :- organelle_type(A,dark_on_bright), number_of_obj(A,few)
nucleolar(A) :- mitotic_related(A,mitotic_neutral), same_image(A,C), organelle_type(C,bright_on_dark)
nucleolar(A) :- intensity(A,positive), organelle_type(A,bright_on_dark), mitotic_related(A,mitotic_da
```

Fig. 4: Examples of induced rules

5 Conclusion

Construction of interpretable models from subsymbolic data can be a non-trivial task. In this paper, we have proposed a method that does this in the context of computer aided diagnosis (CAD). The method consists of (1) defining symbolic

features that are interpretable to humans; (2) learning a first layer of models that map raw data (images) onto a description based on these features; (3) learning an interpretable model that maps the feature-based description to some target variable. Experiments show that, on the domain considered, this interpretable model can achieve accuracy comparable to black-box models.

Within this particular application, other possible future work includes automatic mitotic cell detection and artefact removal, broader experimentation with deep learning approaches and introduction of uncertainty in rules.

Acknowledgements

This work is funded by the KU Leuven Research Fund (project IDO/10/012).

References

1. Foggia, P. et al.: Benchmarking HEp-2 Cells Classification Methods. IEEE Trans. Med. Imaging, vol. 32, 1878-1889 (2013)
2. Huang, Y.L. et al.: Adaptive Automatic Segmentation of HEp-2 Cells in Indirect Immunofluorescence Images. IEEE International Conference on Sensor Networks, Ubiquitous, and Trustworthy Computing, IEEE Computer Society, 418–422 (2008)
3. Huang, Y.L. et al.: Outline Detection for the HEp-2 Cell in Indirect Immunofluorescence Images Using Watershed Segmentation. IEEE International Conference on Sensor Networks, Ubiquitous, and Trustworthy Computing, IEEE Computer Society, pp. 423–427 (2008)
4. Agrawal,P., Vatsa, M., Singh, R.: HEp-2 Cell Image Classification: A Comparative Analysis. Lecture Notes in Computer Science, Machine Learning in Medical Imaging, vol. 8184, Springer International Publishing, 195-202, (2013)
5. Williem, A. et al: Classification of Human Epithelial Type 2 Cell Indirect Immunofluoresence Images via Codebook Based Descriptors. 2013 IEEE Workshop on Applications of Computer Vision (WACV), IEEE Computer Society, 95–102, (2013)
6. Dempster, A. P., Laird, N. M., Rubin,D. B.: Maximum likelihood from incomplete data via the EM algorithm. Journal of the Royal Statistical Society, series B, vol. 39, pp. 1–38 (1977)
7. Hough, P. V. C.: Machine Analysis of Bubble Chamber Pictures. International Conference on High Energy Accelerators and Instrumentation, vol. C590914, 554-558 (1959)
8. Marquez-Neila, P., Baumela, L., Alvarez, L.: A Morphological Approach to Curvature-Based Evolution of Curves and Surfaces. IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 36, IEEE Computer Society, 2–17 (2014)
9. Vapnik, V., Cortes, C.: Support-Vector Networks. Machine learning, vol. 20, Kluwer Academic Publishers, 273–297 (1995)
10. Bengio, Y.: Learning Deep Architectures for AI. Found. Trends Mach. Learn., vol. 2, Now Publishers Inc., 1–127 (2009)
11. Larochelle, H., Bengio,Y.: Classification using discriminative restricted boltzmann machines. 25th international conference on Machine learning, ACM (2008)

12. Murphy, K. : Machine Learning: A Probabilistic Perspective (Adaptive Computation and Machine Learning series). MIT Press (2012)
13. Lavrač, N. and Džeroski,S. : Inductive Logic Programming: Techniques and Applications. Routledge (1993)
14. Quinlan, J. R. earning logical definitions from relations Machine Learning, vol. 5, 239–266 (1990)
15. Muggleton, S. Inverse Entailment and Progol New Generation Computing, vol 13, 245–286 (1995)
16. Cohen, William W. Fast Effective Rule Induction Proceedings of the Twelfth International Conference on Machine Learning, Morgan Kaufmann, 115–123 (1995)
17. Belongie, S., Malik, J. and Puzicha, J. Matching Shapes International Conference on Computer Vision, vol. 1, 454–461 (2001)
18. Palm, R. B. : Prediction as a candidate for learning deep hierarchical models of data. Master's Thesis (2012)
19. Radlinski, F. and Joachims, T. : Query Chains: Learning to Rank from Implicit Feedback International Conference On Knowledge Discovery and Data Mining, ACM SIGKDD, 239–248 (2005)
20. Harder,N. et al. : Automatic analysis of dividing cells in live cell movies to detect mitotic delays and correlate phenotypes in time Genome Research, vol. 19(11), 2113–2124 (2009)
21. Godinez, W. et al : Deterministic and probabilistic approaches for tracking virus particles in time-lapse fluorescence microscopy image sequences Medical Image Analysis, vol. 13, 325-342 (2009)
22. Richardson, M. and Domingos, P. : Markov Logic Networks MAchine learning, volume 62 (1-2), 107-136 (2006)
23. Singla, P. and Domingos, P. : Discriminative Training of Markov Logic Networks Proceedings of the 20th International Conference on Artificial Intelligence, vol. 2, 868–873 (2005)
24. Sen, P. and Namata, G. and Bilgic,M. and Getoor, L. and Gallagher, B. and Eliassi-Ra, T. : Collective Classification in Network Data AI Magazine, vol. 93, 93 – 106 (2008)
25. Xu, X. et al: Linear Local Distance coding for classification of HEp-2 staining patterns Winter Conference of Application of Computer Vision, 393-400 (2014)
26. Wiik, A.S. and Høier-Madsen, M. and Forslid, J. and Charles, P. and Meyrowitsch, J.: Antinuclear antibodies: A contemporary nomenclature using HEp-2 cells Journal of Autoimmunity , vol. 35 (3), 276-290 (2010)
27. Bolon, P.: Cellular and Molecular Mechanisms of Autoimmune Disease Toxicologic Pathology, vol. 40 (2), 216-229 (2012)
28. Rigon, A. and Soda, P. and Zennaro, D. and Iannello, G. and Afeltra, A. : Indirect immunofluorescence in autoimmune diseases: Assessment of digital images for diagnostic purpose Cytometry Part B-clinical Cytometry, vol. 72B, 472-477 (2007)
29. Blei, D. M. : Probabilistic Topic Models Commun. ACM, vol. 55 (4), 77 – 84 (2012)