Ensemble of Deep Learning Approaches for ATC Classification

Loris Nanni[1], Sheryl Brahnam*[2], and Alessandra Lumini[3],

[1] DEI - University of Padova, Via Gradenigo, 6 - 35131- Padova, Italy. loris.nanni@unipd.it.
[2] Management and Information Systems, Glass Hall, Room 387, Missouri State University, 901 S. National, Springfield, MO 65804, USA. sbrahnam@missouristate.edu
[3] DISI, Università di Bologna, Via Sacchi 3, 47521 Cesena, Italy. alessandra.lumini@unibo.it.

Abstract. Anatomical Therapeutic Chemical (ATC) classification of unknown compounds is essential for drug development and research. In this paper, we propose a multi-label classifier system for ATC prediction based on convolutional neural networks (CNN) and Long Short-Term Memory Networks (LSTM). The CNN approach extracts a 1D feature vector from the compounds utilizing information about their chemical-chemical interaction and structural and fingerprint similarities to other compounds belonging to the ATC classes. The 1D vector is then reshaped into a 2D matrix. A CNN network is trained on the matrix and used to extract new features. LSTM is trained on the 1D vector and likewise used to extract features. These features are then trained on two general-purpose classifiers designed for multi-label classification and results are fused. Rigorous experimental evaluation demonstrates the superiority of our method compared to other state-of-the-art approaches.

Keywords: ATC classification, Deep learning, Convolutional neural networks, Long Short-Term Memory Networks.

1 Introduction

Drug discovery cost millions of dollars (USD) and can take more than a decade to accomplish with no guarantee of success after clinical trials. New drugs fail primarily due to lack of efficacy and adverse side-effects [1]. It is crucial, therefore, to develop methods for accurately predicting drug therapeutic indications and side-effects. One feasible avenue for accomplishing this goal is to develop systems that automatically predict efficacy and side-effects based on the Anatomical Therapeutic Chemical (ATC) classes of a given compound. The ATC system, proposed by the World Health Organization (WHO), is a multi-label classification system that categorizes drugs by simultaneously considering their anatomical distribution, therapeutic effects, and chemical characteristics [2]. Automatic compound prediction based on ATC could potentially speed up drug development and significantly reduce the costs of developing new drugs.

The ATC system categorizes drugs based on five levels of classification that include a set of overlapping classes. The first level has fourteen main classes (see Table 1). Unfortunately, only a small portion of drugs have been labeled with ATC codes—mainly because the traditional experimental methods used for identifying ATC classes for new drugs and compounds is highly complex. As a result, many machine learning systems and web servers performing ATC classification have been proposed [3, 4], with most methods, including the one proposed here, focusing on the first ATC level with its fourteen classes.

Early research in this area mostly performed exclusive classification, with Dunkel et al. [3], for example, proposing a system utilizing the compound's structural fingerprint information and Wu et al. [4] making predictions based on discovering relationships among the ATC classes. Chen et al. [2] was one of the first to propose a multi-label classification method based on chemical-chemical interactions, producing, in addition, a benchmark dataset of 4,376 drugs obtained by selecting ATC-coded drugs from the publicly available drug databank KEGG [5]. In the last couple of years, Cheng et al. [6, 7] have proposed two approaches (iATCmISF [6] and iATC-mHyb [7]) based on the fusion of different sets of 1D descriptors, such as the chemical-chemical interaction, structural similarity, and fingerprint similarity, to effectively handle class overlapping. In [8] the same 1D descriptors used in [6] were reshaped by Nanni and Brahnam as a set of 2D matrices that were then fed into a multi-label classifier system. In [9] Lumini and Nanni used the 2D matrices to train a Convolutional Neural Network (CNN).

CNN [10] is a deep learning approach that has gained widespread use in image classification where such networks analyze input images by evaluating features that have been learned directly from observations of the training set and preprocessed using a pyramidal approach. Several papers (see, e.g. [11]) have demonstrated that the first layers of a CNN are generalizable in their ability to represent the semantics of the data. Moreover, these layers provide great robustness to intra-class variability [12]. Trained CNNs can also be effectively reused in other problems as either feature extractors or as classifiers after ad hoc retraining [13].

In this paper, we propose a multi-label classifier system for ATC prediction based on CNN and Long Short-Term Memory Networks (LSTM) [14]. With the CNN approach, we extract a 1D feature vector from a compound by utilizing information about its chemical-chemical interaction and structural and fingerprint similarities to other compounds belonging to the different ATC classes, as in [6]. The 1D vector is then reshaped into 2D matrices, as in [8]. A CNN network is trained on the matrices and used to extract sets of new features. LSTM is trained on the 1D vector and likewise used to extract features. These features are then trained on two general-purpose classifiers designed for multi-label classification. Results are fused by average rule.

2 Materials and Methods

2.1 2D Representation of 1D Descriptors

A common approach for solving pattern recognition problems is to implement some feature selection strategy starting from the input data and feed the extracted 1D feature vector into a classifier system. In [15] the authors demonstrate the value of reshaping

the 1D feature vector into a 2D representation to exploit the correlation available in some well-known texture descriptors. This approach has already proven successful in ATC classification in [8].

In order to obtain a 2D matrix to be used as the input of a pre-trained CNN, we perform random reshaping. Given the original 1D feature vector $f \in \Re^n$ (where $n = 42$ for the ATC problem), we obtain the output matrix $\mathbf{M} \in \Re^{d \times d}$ (where d is determined by the chosen pre-trained CNN architecture: either $d = 227$ or $d = 224$ in this work). $\mathbf{M}$ is obtained by performing a random rearrangement (the same for all patterns) of the original 1D input vector into a square matrix $\mathbf{U} \in \Re^{u \times u}$ (where $u = n^{0.5}$) and by resizing $\mathbf{U}$ to $d \times d$ using bicube interpolation. Because performance varies using different feature dispositions, a simple approach for improving performance is to design an ensemble based on feature perturbation by performing K reshaping operations (where $K = 5$ in our experiments), thereby fine-tuning CNN K times as described below.

2.2 Deep Features

Features Extracted from Convolutional Neural Networks. CNNs contain several specialized layers (e.g., the convolutional, pooling, and fully-connected layers) whose weights are trained with the backpropagation algorithm on a large dataset with labels. Some well-known CNN architectures include LeNet [16], AlexNet [17], VGGNet [18], GoogleNet [19], and ResNet [20].

As demonstrated in [11], CNN has great generalization power so that in cases where the training set is small transfer learning [21] can be applied.

As in [9], AlexNet is fine-tuned here on the ATC benchmark dataset using the testing protocol detailed in section **Error! Reference source not found.**. The fine-tuning of AlexNet on the training set for the ATC problem is performed by changing the number of nodes in the last fully-connected layer to the number of level one ATC classes ($c = 14$). The maximum number of epochs for training is set to 40, the mini-batch size is BS$\in\{10, 30, 50\}$, and the fixed learning rate LR is either 0.001 or 0.0001. The second fully connected layer, which contains 4096 nodes, is used for feature extraction, thereby extracting a 1D descriptor of length 4096. If a training pattern belongs to more than one class (e.g. to m classes), the training pattern is replicated m times in the training set, each time with a different label.

Long Short-Term Memory Networks. LSTM, first proposed in 1997 [14], is a traditional Recurrent Neural Network (RNN) that replaces the hidden units with a memory block (gated cells).

In this study, we use the MATLAB LSTM implementation with the following parameter settings: $inputsize = 12$, $numHiddenUnits = 100$, $maxEpochs = 100$, and $miniBatchSize = 27$. If a training pattern belongs to more than one class (e.g. to m classes), the training pattern is replicated m times in the training set, each time with a different label. The last layer of LSTM is used to train the multilabel classification systems detailed in section 2.3.

2.3 Data Classification and Fusion

We use two multi-label classifiers: LIFT and RR.

LIFT (multi-label learning with Label specIfic FeaTures) [22]: this classifier is a two-step method. The first step is aimed at selecting features specific for each class by

means of clustering analysis. The second step trains a set of Support Vector Machines (SVMs) using the features selected for each class. In this work we use linear kernel SVMs. The final response for an unknown sample is obtained by comparing the response of each classifier to a fixed threshold τ (unless otherwise specified, $\tau = 0.5$).

RR (Ridge Regression classifiers) [23]: RR is an extension for linear regression that includes a regularization term in the loss function that is used to penalize high values of the learned coefficients according to the parameter λ ($\lambda = 1$ in our experiments).

3 Materials and Methods

We use the benchmark dataset provided in [2]. This dataset (see Table 1) contains a total of 3883 drugs divided into the 14 first level nonexclusive ATC-classes (3295 samples belong to only one class, 370 belong to two classes, 110 belong to three classes, 37 belong to four classes, 27 belong to five classes, and 44 belong to six classes).

The descriptors used to represent drugs are based on drug-drug interaction and the correlation with the target classes to be predicted. Given the set of 14 first level ATC classes, each sample can be represented starting with three mathematical expressions reflecting its intrinsic correlation with each of the classes. This produces a final descriptor of $14 \times 3 = 42$ features. The three different properties considered are (1) the maximum interaction score with the drugs in each of the 14 classes, (2) the maximum structural similarity score with the drugs in each class, and (3) the molecular fingerprint similarity score in the 14 subsets. These descriptors are available for download in the supplementary material of [8].

Table 1. Summary of the Benchmark Dataset According to the First Level ATC Classes.

First Level ATC Class	Number of Drugs
Alimentary Tract and Metabolism	540
Blood and Blood Forming Organs	133
Cardiovascular System	591
Dermatologicals	421
Genito-Urinary System and Sex Hormones	248
Systemic Hormonal Preparations, Excluding Sex Hormones and Insulins	126
Anti-infectives For Systemic Use	521
Antineoplastic and Immunomodulating Agents	232
Musculo-Skeletal System	208
Nervous System	737
Antiparasitic Products, Insecticides and Repellents	127
Respiratory System	427
Sensory Organs	390
Various	211
Number of Total Virtual Drugs N(Vir)	4912
Number of total drugs	3883

All experiments according to the jackknife test. The following five metrics are defined for this task as in [24]:

$$\text{Aiming} = \frac{1}{N}\sum_{k=1}^{N}\left(\frac{\|\mathbb{L}_k \cap \mathbb{L}_k^*\|}{\|\mathbb{L}_k^*\|}\right) \quad (1)$$

$$\text{Coverage} = \frac{1}{N}\sum_{k=1}^{N}\left(\frac{\|\mathbb{L}_k \cap \mathbb{L}_k^*\|}{\|\mathbb{L}_k\|}\right) \quad (2)$$

$$\text{Accuracy} = \frac{1}{N}\sum_{k=1}^{N}\left(\frac{\|\mathbb{L}_k \cap \mathbb{L}_k^*\|}{\|\mathbb{L}_k \cup \mathbb{L}_k^*\|}\right) \quad (3)$$

$$\text{Absolute True} = \frac{1}{N}\sum_{k=1}^{N}\Delta(\mathbb{L}_k, \mathbb{L}_k^*) \quad (4)$$

$$\text{Absolute False} = \frac{1}{N}\sum_{k=1}^{N}\left(\frac{\|\mathbb{L}_k \cup \mathbb{L}_k^*\| - \|\mathbb{L}_k \cap \mathbb{L}_k^*\|}{M}\right), \quad (5)$$

where $\mathbb{L}_k$ and $\mathbb{L}_k^*$ are the "Actual" and "Predicted" labels for a given sample *k, respectively, N* is the number of samples, *M* the number of classes, and $\Delta(\cdot,\cdot)$ is an operator returning 1 if the two sets have the same elements, 0 otherwise.

In Table 2, we report the results of an experiment designed to evaluate the performance of the features proposed in this work. Performance is reported using different values for Batch Size (BS) and Learning Rates (LR) for the AlexNet CNN features. A cell containing the label *FUS* means a combination by average rule of the classifiers trained with $\text{BS} = \{10, 30, 50\}$ and $\text{LR} = 0.001$. Due to space constraints, for LIFT we only report the performance obtained with all the values of BS and LR, and for RR we only report the performance obtained using FUS and the best combinations of BS and LR.

Table 2. Success Rates Achieved by the AlexNet CNN Features.

Classifier	BR	LR	Aiming	Coverage	Accuracy	Absolute True	Absolute False
LIFT	10	0.001	0.8798	**0.6527**	**0.6692**	0.6271	0.0321
	30	0.001	0.8912	0.6421	0.6611	0.6217	0.0324
	50	0.001	0.8897	0.6423	0.6594	0.6187	0.0325
	10	0.0001	0.8886	0.6341	0.6510	0.6104	0.0334
	30	0.0001	0.8889	0.6318	0.6487	0.6083	0.0336
	50	0.0001	0.8886	0.6318	0.6493	0.6089	0.0336
LIFT	FUS		**0.8924**	0.6515	0.6686	**0.6289**	**0.0320**
RR	10	0.001	0.8713	0.7028	0.7001	0.6570	0.0278
RR	FUS		**0.8760**	**0.7079**	**0.7065**	**0.6662**	**0.0269**

In Table 3, the performance achieved by the LSTM feature extractor is reported. The label *LSTM* represents a single run of LSTM and *eLSTM* represents running LSTM twenty times, training twenty LIFT/RR, and then combining the scores by average rule.

We also report the results of three other ensembles:

- FUS1, average rule between eLSTM_LIFT and eLSTM_RR;
- FUS2, average rule between eLSTM_LIFT and AlexNET_RR_FUS
- FUS3, average rule among eLSTM_LIFT, eLSTM_RR, AlexNET_RR_FUS,

Table 3. Success Rates Achieved by LSTM Features.

Ensemble	Classifier	Aiming	Coverage	Accuracy	Absolute True	Absolute False
LSTM	LIFT	0.8154	0.7148	0.7470	0.6853	0.0209
eLSTM	LIFT	0.8181	**0.7157**	0.7512	0.6899	0.0207
LSTM	RR	0.8655	0.6287	0.7109	0.6686	0.0270
eLSTM	RR	0.8670	0.6317	0.7132	0.6706	0.0269
FUS1		0.8353	0.6887	0.7427	0.6871	0.0218
FUS2		0.8465	0.7321	**0.7549**	**0.7018**	**0.0205**
FUS3		**0.8755**	0.6973	0.7346	0.6871	0.0238

Finally, in Table 4 we compare our approach with the current state of the art. In addition, we combine our system with the drug ontologies (DO) proposed in [6] (labelled as FUS $\otimes$ DO. The fusion process was very simple: if DO features were present, LIFT was trained with DO (i.e. when available since only 1,144 drug compounds in the benchmark dataset have a DO descriptor; the other 2,689 samples were classified considering only the first representation); otherwise, the score was given by our ensemble.

Moreover, in Table 4, we report the performance of FUS3, fixing τ for obtaining a coverage similar to the previous best method (set for easy comparison of the two approaches).

Table 4. Comparison with the Literature.

Ensemble	Aiming	Coverage	Accuracy	Absolute True	Absolute False
FUS2	0.8465	0.7321	0.7549	0.7018	0.0205
FUS3	0.8755	0.6973	0.7346	0.6871	0.0238
FUS3 $\otimes$ DO ($\tau = 0.27$)	0.7716	0.8245	0.7785	0.7049	0.0205
FUS3 $\otimes$ DO ($\tau = 0.25$)	**0.7979**	**0.8422**	**0.7964**	**0.7304**	**0.0209**
FUS3 $\otimes$ DO ($\tau = 0.5$)	0.9011	0.7309	0.7603	0.7211	0.0226
Chen et al. [2]	0.5076	0.7579	0.4938	0.1383	0.0883
EnsLIFT [8]	0.7818	0.7577	0.7121	0.6330	0.0285
iATC-mISF [6]	0.6783	0.6710	0.6641	0.6098	0.0585
iATC-mHYb [7]	0.7191	0.7146	0.7132	0.6675	0.0243
EnsANET_LR [9]	0.7536	0.8249	0.7512	0.6668	0.0262
EnsANET_LR $\otimes$ DO ($\tau = 0.25$) [9]	**0.7957**	**0.8335**	**0.7778**	**0.7090**	**0.0240**
EnsANET_LR $\otimes$ DO ($\tau = 0.5$) [9]	0.9011	0.7162	0.7232	0.6871	0.0267

In Table 4, the following state-of-the-art methods are reported:

- EnsLIFT [8]: ensemble of 50 LIFT classifiers trained using HoG
- iATC-mISF [6]: a predictor based on the fusion of different descriptors
- iATC-mHyb [7]: a hybrid approach based on the combination of iATC-mISF and the predictor iATC-mDO based on drug ontologies (DO).
- EnsANet_LR, ensemble proposed in [9].

- EnsANet_LR ⊕ DO: the combination of EnsANet_LR and the DO features used in [7].

An examination of the results in Table 4 demonstrates that combining deep descriptors improves performance compared to recent state-of-the art approaches. It has been already shown in [7] that mapping compounds into the DO database space and fusing such information with other descriptors significantly enhances the quality of ATC classification. Our final system, which combines deep features and DO (when such information is available) obtains the best performance with respect to all other methods published in the literature.

4 Conclusion

In this paper we experimentally generate a new ensemble for predicting a compound's ATC class/classes. This is a difficult multi-label problem. The proposed ensemble is based on two approaches:

1. Reshaping a 1D ATC input feature vector into a 2D matrix so that transfer learning from the fine-tuned CNN (AlexNet) can be used as both a classifier and feature extractor;
2. Training the deep neural network LSTM with the 1D ATC feature vector.

The decisions of the two approaches are then fused by average rule.

Extensive experiments demonstrate that this new approach, though not performing as well as [9] across all five metrics for evaluating the performance for multi-label systems. The *absolute true* rate and the *absolute false* rate prove to be the two most significant indexes.

All MATLAB code used in our proposed system is available at https://github.com/LorisNanni.

Acknowledgments

We would like to acknowledge the support that NVIDIA provided us through the GPU Grant Program. We used a donated TitanX GPU to train the CNNs used in this work.

References

1. Pitts, R.C.: Reconsidering the concept of behavioral mechanisms of drug action. Journal of the Experimental Analysis of Behavior 101, 422–441 (2014)
2. Chen, L.: Predicting anatomical therapeutic chemical (ATC) classification of drugs by integrating chemical-chemical interactions and similarities. PLoS ONE 7, (2012)
3. Dunkel, M., Günther, S., Ahmed, J., Wittig, B., Preissner, R.: SuperPred: update on drug classification and target prediction. Nucleic Acids Research 36, W55-W59 (2008)

4. Wu, L., Ai, N., Liu, Y., Fan, X.: Relating anatomical therapeutic indications by the ensemble similarity of drug sets. Journal of Chemical Information and Modeling 53, 2154-2160 (2013)
5. Ogata, H., Goto, S., Sato, K., Fujibuchi, W., Bono, H., Kanehisa, M.: KEGG: Kyoto Encyclopedia of Genes and Genomes. Nucleic Acids Research 27, 29-34 (1999)
6. Cheng, X., Zhao, S.-G., Xiao, X., Chou, K.-C.: iATC-mISF: a multi-label classifier for predicting the classes of anatomical therapeutic chemicals. Bioinformatics 33, 341-346 (2017)
7. Cheng, X., Zhao, S.-G., Xiao, X., Chou, K.-C.: iATC-mHyb: a hybrid multi-label classifier for predicting the classification of anatomical therapeutic chemicals. Oncotarget 8, 58494–58503 (2017)
8. Nanni, L., Brahnam, S.: Multi-label classifier based on histogram of gradients for predicting the anatomical therapeutic chemical class/classes of a given compound. Bioinformatics 33, 2837-2841 (2017)
9. Lumini, A., Nanni, L.: Convolutional neural networks for ATC classification. Current Pharmaceutical Design (In Press)
10. Schmidhuber, J.: Deep learning in neural networks: An overview. Neural Networks 61, 85-117 (2015)
11. Nanni, L., Ghidoni, S., Brahnam, S.: Handcrafted vs non-handcrafted features for computer vision classification Pattern Recognit 71, 158-172 (2017)
12. Chan, T.-H., Jia, K., Gao, S., Lu, J., Zeng, Z., Ma, Y.: Pcanet: a simple deep learning baseline for image classification? IEEE Transactions on Image Processing 24, 5017-5032 (2015)
13. Nanni, L., Ghidoni, S.: How could a subcellular image, or a painting by Van Gogh, be similar to a great white shark or to a pizza? Pattern Recognit Lett 85, 1-88 (2017)
14. Hochreiter, S., Schmidhuber, J.: Long short-term memory. Neural Computation 9, 1735-1780 (1997)
15. Nanni, L., Brahnam, S., Lumini, A.: Matrix representation in pattern classification. Expert Systems with Applications Appl. 39.3, 3031-3036 (2012)
16. LeCun, Y., Bottou, L., Bengio, Y., Haffner, P.: Gradient-based learning applied to document recognition. Proceeding of the IEEE 86, 2278-2323 (1998)
17. Krizhevsky, A., Sutskever, I., Hinton, G.E.: ImageNet Classification with Deep Convolutional Neural Networks. In: Pereira, F., Burges, C.J.C., Bottou, L., Weinberger, K.Q. (eds.) Adv Neural Inf Process Syst, pp. 1097-1105. Curran Associates, Inc., Red Hook, NY (2012)
18. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. Cornell University (2014)
19. Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., Rabinovich, A.: Going deeper with convolutions. IEEE Computer Society Conference on Computer Vision and Pattern Recognition, pp. 1-9 (2015)
20. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 770-778. IEEE, Las Vegas, NV (2016)
21. Yosinski, J., Clune, J., Bengio, Y., Lipson, H.: How transferable are features in deep neural networks? , Cornell University (2014)
22. Zhang, M.-L., Wu, L.: Lift: multi-label learning with label-specific features. IEEE Trans Pattern Analysis Mach Intell 37, 107-120 (2015)
23. Kimura, K., Sun, L., Kudo, M.: MLC toolbox: A MATLAB/OCTAVE library for multi-label classification. ArXiv arXiv:1704.02592, (2017)
24. Chou, K.C.: Some remarks on predicting multi-label attributes in molecular biosystems. Molecular Biosystems 9, 10922-11100 (2013)