Neural networks for anatomical therapeutic chemical (ATC) classification

Loris Nanni
Department of Information Engineering, University of Padua, Padua, Italy
Alessandra Lumini
Department of Computer Science and Engineering, University of Bologna, Bologna, Italy, and
Sheryl Brahnam
Department of Information Technology and Cybersecurity, Missouri State University, Springfield, Missouri, USA

Received 10 November 2021
Revised 26 January 2022
28 February 2022
Accepted 2 March 2022

Abstract

Purpose – Automatic anatomical therapeutic chemical (ATC) classification is progressing at a rapid pace because of its potential in drug development. Predicting an unknown compound's therapeutic and chemical characteristics in terms of how it affects multiple organs and physiological systems makes automatic ATC classification a vital yet challenging multilabel problem. The aim of this paper is to experimentally derive an ensemble of different feature descriptors and classifiers for ATC classification that outperforms the state-of-the-art.

Design/methodology/approach – The proposed method is an ensemble generated by the fusion of neural networks (i.e. a tabular model and long short-term memory networks (LSTM)) and multilabel classifiers based on multiple linear regression (hMuLab). All classifiers are trained on three sets of descriptors. Features extracted from the trained LSTMs are also fed into hMuLab. Evaluations of ensembles are compared on a benchmark data set of 3883 ATC-coded pharmaceuticals taken from KEGG, a publicly available drug databank.

Findings – Experiments demonstrate the power of the authors' best ensemble, EnsATC, which is shown to outperform the best methods reported in the literature, including the state-of-the-art developed by the fast.ai research group. The MATLAB source code of the authors' system is freely available to the public at https://github.com/LorisNanni/Neural-networks-for-anatomical-therapeutic-chemical-ATC-classification.

Originality/value – This study demonstrates the power of extracting LSTM features and combining them with ATC descriptors in ensembles for ATC classification.

Keywords Machine learning, Multilabel classifier, Bidirectional long short-term memory, ATC classification, Learned features

Paper type Research paper

1. Introduction

From start to market, the price for engineering new drugs, which can take decades before final approval, is now estimated to be 2.8 billion USD [1]. Of all drugs currently under

This study ran experiments on a TitanX GPU donated by the NVIDIA GPU Grant Program.

Author Contributions: Conceptualization, L.N.; methodology, L.N.; software L.N.; writing—original draft preparation, S.B., A.L. and L.N.; writing—review and editing, S.B., A.L. and L.N. All authors have read and agreed to the published version of the manuscript.

Applied Computing and Informatics
Emerald Publishing Limited
e-ISSN: 2210-8327
p-ISSN: 2634-1964
DOI 10.1108/ACI-11-2021-0301

development, approximately 86% will fail to be better than placebo [2] or will prove to cause more harm than good [3]. In order to weed out new drugs with a low probability of being efficacious and safe, researchers have investigated methods for automatically classifying compounds according to their anatomical therapeutic chemical (ATC) classes.

The ATC classification system [4], proposed by the World Health Organization, is a widely accepted drug classification scheme that categorizes drugs into multiple classes according to their therapeutic, pharmacological and chemical attributes. A given compound can be classified into one or more classes at five levels in terms of the drug's effects on different organs or physiological systems. Most relevant to the automatic ATC classification problem is the first ATC level, which determines the general anatomical groups, as coded with 14 semi-mnemonic letters that a particular compound targets. These alphabetic codes range from *A* (alimentary tract and metabolism) to *V*, a category that includes various groups. Levels 2 and 3 are pharmacological subgroups, and levels 4 and 5 contain chemical subgroups. A compound is assigned to as many ATC codes as relevant within each of these five levels.

Despite the serviceability of the ATC classification system for assessing the clinical value of a compound, most pharmaceuticals have yet to be assigned ATC codes. Accurate coding involves expensive, labor-intensive experimental procedures. Hence, the pressing need for machine learning (ML) to be applied to this problem. Automatic prediction of the ATC classes of a new compound can also provide researchers with deeper insights into its therapeutic indications and side effects, thus accelerating basic research and drug development [5].

In this work, we tackle ATC classification of drugs into first-level classes by experimentally deriving ensembles. EnsATC, the name of our highest performing ensemble, is a data-driven method based on the fusion of different feature descriptors and classifiers, with the best result obtained by combining a bidirectional long short-term memory network (BiLSTM) [6] with a multilabel classifier and a tabular model. EnsATC, along with other candidate ensembles, was evaluated on a popular ATC benchmark developed by Chen *et al.* [7] using the jackknife test. The results obtained by EnsATC strongly outperform the current state of the art.

2. Literature review

Early ML systems tended to simplify the complexity of the ATC classification problem by reducing the level 1 multi-class problem to a single class problem. Dunkel *et al.* [5], for example, took advantage of a compound's unique structure to identify its class, while Wu *et al.* [8] based their approach on extracting relationships among level 1 subgroups.

In the past 10 years, however, researchers have proposed methods for determining multi-class first-level assignments of drugs by taking a multilabel classification approach. Chen *et al.* [7] was one of the first to address the multilabel complexity of ATC classification by examining a drug's chemical–chemical interactions, thereby producing a baseline result for the multilabel approach. The authors also established the de facto benchmark data set for ATC classification. Later, in [9, 10], Cheng *et al.* designed ML systems to handle class overlapping by fusing different descriptors: structural similarity, fingerprint similarity, and chemical–chemical interaction. Nanni and Brahnam [11] transformed Cheng *et al.*'s 1D vectors into images (matrices) and extracted texture descriptors from them, which were then trained on ensembles of multilabel classifiers.

As far as deep learning approaches go, convolutional neural networks (CNNs) were trained on 2D descriptors in [12, 13]. In Lumini and Nanni [13], a set of features were extracted from deep learners for training two successful multilabel classifiers, an approach that was extended in Nanni, Brahnam and Lumini [14], where ensembles of CNNs were constructed by adjusting batch sizes and learning rates, with different methods applied to handle multilabel inputs. Recently, Wang *et al.* [15] proposed a classifier called ATC-NLSP that was trained on

similarity-based features such as chemical–chemical interaction and the structural and fingerprint similarities of a compound, which were compared to other compounds belonging to the different ATC categories. Recent state-of-the-art approaches include the work of Zhou *et al.* [16], who proposed a network embedding method to encode drugs and RAndom k-labELsets to build a classifier named iATC-NRAKEL. Additionally, Zhao *et al.* [17] developed a multilabel classifier called CGATCPred that used CNN for feature extraction. The authors constructed the correlation graph of ATC classes after which a graph convolutional network (GCN) was applied on the graph for label embedding abstraction.

3. Methods

The approach taken in this study is to produce experimentally ensembles that combine multilabel classifiers based on multiple linear regression (hMuLab) [18] and LSTM classification. LSTM is employed both as a classifier and as a feature extractor. Features taken from LSTM are trained on hMuLab and on a tabular model. LSTM features fed into hMuLab classifiers generate an ensemble called eLh (see section 3.3).

As illustrated in Figure 1, all four classifiers, LSTM, eLh, a tabular model, and hMuLab, are trained on X, a set of three different descriptors (DDI, FRAKEL, and NRAKEL), each detailed in section 4.1. The results of different combinations of the above approaches are then evaluated on the Chen *et al.* benchmark [7] described in section 4.1.

LSTM feature extraction and classification are detailed in section 3.1; hMuLab stacking (from original features) and LSTM stacking are presented in sections 3.2-3.3, respectively. The tabular model explored here is the high-performing FastAI tabular model (Tab) [19], which is discussed in section 3.4; the FastAI tabular model was selected because it has thus far obtained the best classification result on the Chen *et al.* benchmark [7].

3.1 LSTM multilabel classifier and feature extractor

LSTM is a recurrent neural network that makes a decision for what to remember at every time step. As illustrated in Figure 2, this network contains three gates: (1) input gate I, (2) output gate O, and (3) forget gate f, each of which consists of one layer with the sigmoid (σ) activation function. LSTM also contains a specialized single layer network candidate $\overline{C}$, which has a activation function. In addition, there are four state vectors: (1) memory state C with (2) its previous memory state C_{t-1} and (3) hidden state H with (4) its previous state H_{t-1}. The variable X in Figure 2 represents the current input at time step t.

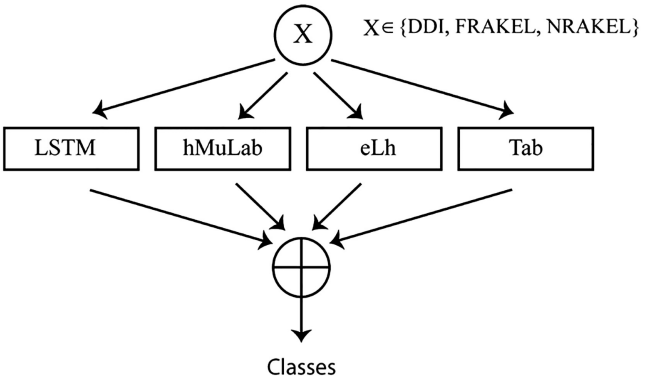

Figure 1.
Schematic of proposed ATC classification approach

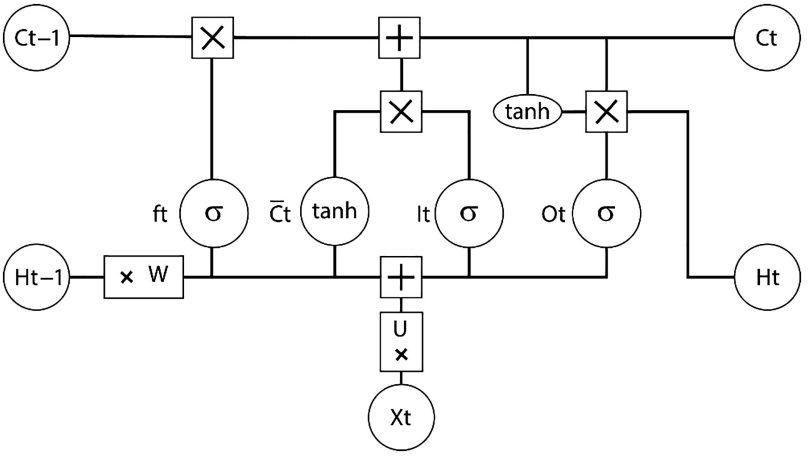

Figure. 2.
Long short-term memory (LSTM) classifier

The process for updating LSTM at time t is as follows. Given X_t and H_{t-1} and letting U, W, b be the learnable weights of the network (each independent of t), the candidate layer $\overline{C_t}$ is

$$\overline{C_t} = (U_c X_t + W_c H_{t-1} + b)_c. \tag{1}$$

The next memory cell is

$$C_t = f_t * C_{t-1} + I_t * \overline{C_t}, \tag{2}$$

where * is element-wise multiplication.
The gates are defined as

$$f_t = \sigma(U_f X_t + W_f H_{t-1} + b_f), \tag{3}$$

$$I_t = \sigma(U_i X_t + W_i H_{t-1} + b_i); \tag{4}$$

$$O_t = \sigma(U_o X_t + W_o H_{t-1} + b_o). \tag{5}$$

The output is $H_t = O_t * \sigma(C_t)$ of O_t and the sigmoid of C_t.

Regarding input, all sequences for this task are of the same length, so sorting input by length is not required. The output of LSTM can be the entire sequence H_t (this permits several layers to be stacked in a single network) or the last term of this sequence.

An LSTM that has two stacked layers trained on the same set of samples is called a bidirectional LSTM (BiLSTM). The second LSTM connects to the end of the first sequence and runs in reverse. BiLSTM is best used to train data not related to time. Accordingly, this study uses the BiLSTM, as implemented in the MATLAB LSTM toolbox. Parameters were set to the following values: $numHiddenUnits = 100$, $numClasses = 14$, and $miniBatchSize = 27$.

LSTM is not ordinarily considered a multilabel classifier but can perform multilabel classification if the training strategy outlined in [14] is implemented, which involves replicating a sample m times for each of its m labels. To assign a test pattern to more than one class, a rule is applied in the final softmax layer where a given pattern is assigned to each of the classes whose score is larger than a given threshold.

LSTM can function not only as a classifier but also as a feature extractor. As noted in the discussion of Figure 1, in this study, LSTM functions in both capacities. Feature extraction

with LSTM is accomplished by representing each pattern using the activations from the last layer, a process that produces a feature vector with a dimension equal to the number of classes. The length of the feature vector is, therefore, *numClasses* = 14.

3.2 Classification by hMuLab

The algorithm hMuLab, proposed in [18], is a multilabel classifier that integrates a feature score and a neighbor score. The feature score decides if a sample belongs to a particular class using the global information contained in the whole training set. In contrast, the neighbor score decides a sample's class labels based on the class assignment of its neighbors. The feature score $f_1(x, g_j)$ for a given pattern x with respect to an anatomical group g_j is calculated to evaluate whether the pattern belongs to the group g_j using a regression model. The neighbor score $f_2(x, g_j)$ calculates the significance of the class membership of K neighbors of a pattern belonging to a given group g_j: the neighbor score increases if more neighbors of x have the label g_j. Thus, is 1 if all neighbors of x belong to g_j, 0 otherwise. The final score of x is a weighted sum of the two factors:

$$f\ (x, gj) = \alpha f_1(x, g_j) + (1 - \alpha) f_2(x, g_j). \in \tag{6}$$

In our experiments, we use the default values where the weight factor α is set to 0.5, and the number of neighbors is $K = 15$.

3.3 Classification by Lh: a stacking method based on LSTM and hMuLab

Lh is the name we give to a stacking method where descriptors extracted from LSTM become the input to an hMuLab classifier, as illustrated in Figure 3 (left). Feature perturbation and extraction can be performed multiple times by randomly sorting the original features used to train the LSTM. The fusion of 10 Lh classifiers trained using the random rearrangements of the input features is labeled eLh, as illustrated in Figure 3 (right).

3.4 Classification by FastAI tabular model

FastAI tabular model [19] is a powerful deep learning technique for tabular/structured data based on the creation of some embedding layers for categorical variables. This deep learner

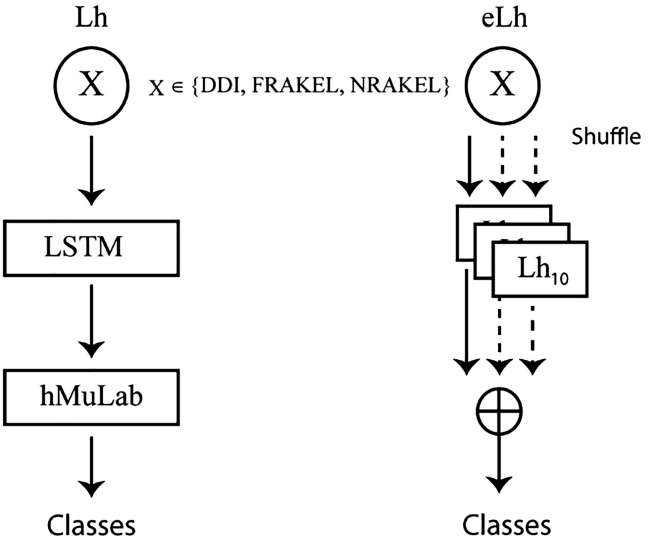

Figure. 3.
Lh (left) is a stacking of LSTM and hMuLab; eLh (right) is an ensemble of 10 Lh classifiers based on feature shuffling

uses embedding layers to represent categorical variables by a numerical vector whose values are learned during training. Embeddings allow for relationships between categories to be captured, and they can also serve as inputs to other models.

A graphic representation of a FastAI tabular model is presented in Figure 4, where one can observe that the categorical variables are transformed into N-dimensional features by categorical embeddings followed by a dropout layer to prevent overfitting. Numerical variables are simply normalized. Then all the variables are concatenated and passed as input into the following layers, which, in our experiments, are two hidden layers and one output layer. We also use a binary encoding to represent binary variables, and the resulting variable is treated as categorical.

4. Experimental results

4.1 The data set and descriptors

For the sake of comparison with state-of-the-art approaches, our method is trained and evaluated on the data set in [7] (supporting information S1). This data set is a collection of 3883 ATC-coded pharmaceuticals taken from KEGG [20], a publicly available drug databank;

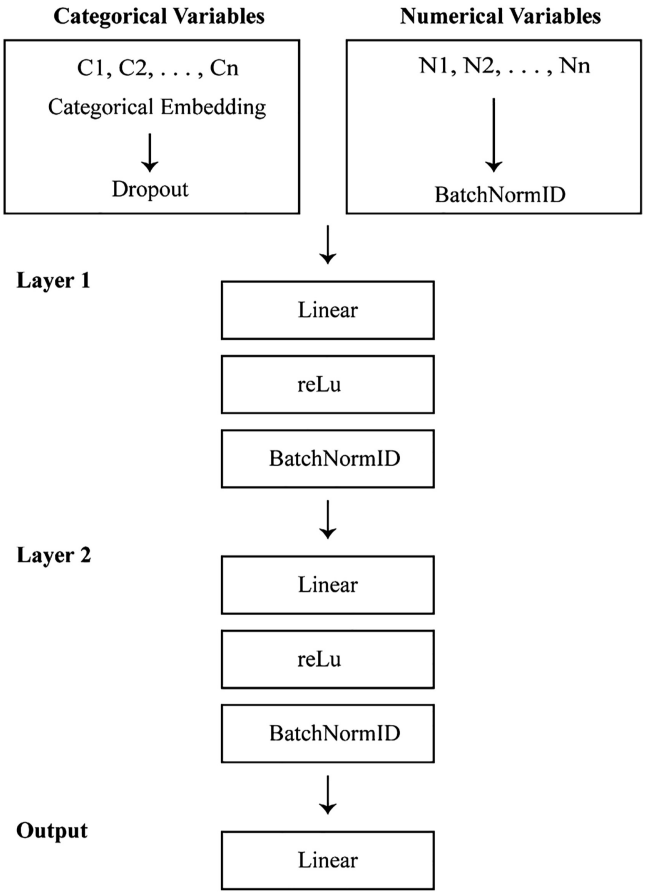

Figure 4.
Schematic of the FastAI tabular model

the drug subset was obtained by selecting labeled samples with no missing values and contradictory records [7]. As noted in the introduction, samples can belong to more than one of the 14 level 1 ATC classes. In the Chen data set, 3295 drugs belong to one class, 370 to two classes, 110 to three classes, 37 to four classes, 27 to five classes and 44 belong to six classes. The average number of labels per sample is 1.27.

The following three sets of descriptors represent the drugs in this data set:

(1) *DDI* represents each drug by concatenating three types of features [9]: the maximum interaction score with the drugs, the maximum structural similarity score, and the molecular fingerprint similarity score, with each expression based on its correlation with the 14 level 1 classes. Thus, the resulting descriptor is size $14 \times 3 = 42$ (available in the supplementary material in Nanni and Brahnam [11]).

(2) *FRAKEL* represents each drug by its ECFP fingerprint [16], which is a 1024-dimensional binary vector (located at http://cie.shmtu.edu.cn/iatc/index). The descriptor is obtained by feeding the drug into RDKit (http://www.rdkit. org/), a free ML toolkit for chemistry informatics. From this 1024-dimensional binary vector, a 64-dimensional categorical descriptor is obtained, representing each group in 16 bits as an integer. This version of FRAKEL has been used with the FastAI tabular model.

(3) *NRAKEL* represents a drug by a 700-dimensional descriptor obtained from the Mashup algorithm [21], which generates output from seven drug networks (five based on chemical–chemical interaction and two on drug similarities).

4.2 Testing protocol

The jackknife testing protocol [7] is used here to generate both the training and testing sets. At each iteration of this protocol, one sample is placed in the testing set and the remainder in the training set. Iteration continues until each pattern has taken a turn in the testing set. The jackknife protocol was selected for facilitating comparison with other approaches as stipulated in [22]. All the experiments have been performed using MATLAB and Python.

4.3 Performance indicators

ATC classification is evaluated using the standard performance indicators defined in [22] and repeated below:

$$\text{Aiming} = \frac{1}{N}\sum_{k=1}^{N}\left(\frac{\|\mathbb{L}_k \cap \mathbb{L}_k^*\|}{\|\mathbb{L}_k^*\|}\right), \tag{7}$$

$$\text{Coverage} = \frac{1}{N}\sum_{k=1}^{N}\left(\frac{\|\mathbb{L}_k \cap \mathbb{L}_k^*\|}{\|\mathbb{L}_k\|}\right), \tag{8}$$

$$\text{Accuracy} = \frac{1}{N}\sum_{k=1}^{N}\left(\frac{\|\mathbb{L}_k \cap \mathbb{L}_k^*\|}{\|\mathbb{L}_k \cup \mathbb{L}_k^*\|}\right), \tag{9}$$

$$\text{Absolute True} = \frac{1}{N}\sum_{k=1}^{N}\Delta\left(\mathbb{L}_k, \mathbb{L}_k^*\right), \tag{10}$$

$$\text{Absolute False} = \frac{1}{N}\sum_{k=1}^{N}\left(\frac{\|\mathbb{L}_k \cup \mathbb{L}_k^*\| - \|\mathbb{L}_k \cap \mathbb{L}_k^*\|}{M}\right), \tag{11}$$

where M is the number of classes, N is the number of samples, $\mathbb{L}_k$ is the true label, $\mathbb{L}_k^*$ is the predicted label, and $\Delta(\cdot, \cdot)$ returns 1 if the two sets have the same elements, 0 otherwise.

4.4 Experiments

The first experiment (see Table 1) compares the multilabel classifiers described in section 3. Also compared are three other standard classifiers, each trained on the three sets of features (DDI, FRAKEL, and NRAKEL). As already mentioned, LSTM is not a native multilabel classifier. For multilabel decisions, thresholding was used, as described in section 3.1, to adapt LSTM to the ATC classification problem.

The methods reported in Table 1 are the following:

(1) RR, a ridge regression ensemble using the MATLAB/OCTAVE library for multi-class classification in the MLC Toolbox [23];

(2) LIFT, multilabel learning with Label specIfic FeaTures) [24];

(3) Group preserving label embedding (GR) [25];

(4) LSTM;

(5) Tab (FastAI tabular model) [19];

(6) hMuLab [18].

(7) Lh, the stacking method described in section 3.3

(8) eLh, the ensemble of 10 Lh classifiers described in section 3.3.

In addition, the fusion by average rule of some of the above-mentioned methods is reported in Table 1. Each ensemble is specified via the concatenation symbol + (thus, LSTM + hMuLab is the fusion of LSTM and hMuLab). When the weight of an approach is higher than the other it is preceded by a weighing factor (thus, 3×Tab means that Tab is weighted by 3 before fusion).

In the cell labeled Tab-FRAKEL, the reported value was obtained by transforming the original 1024 bit feature vector into 64 int16 features, since the original descriptor gained very low performance (0.3165). To avoid overfitting, default parameters were used for the classifiers.

Examining the results in Table 1, Tab is the best standalone approach, producing an outstanding 0.7422 absolute true rate using NRAKEL descriptors. Of note as well is LSTM, which produced good results on all three descriptors. A strong performance improvement for hMuLab is obtained by using LSTM descriptors (method Lh).

Table 1.
Absolute true rates achieved by the classifiers trained on the three descriptors DDI, NRAKEL, and FRAKEL

Absolute true	DDI	NRAKEL	FRAKEL
RR	0.5127	0.6062	0.5006
LIFT	0.6111	0.5282	0.3579
GR	0.4991	0.6093	0.4963
LSTM	0.6626	0.6585	0.6330
Tab	0.6441	0.7422	0.6760
hMuLab	0.5710	0.6791	0.5977
Lh	0.6902	0.7092	0.6709
eLh	*0.6995*	0.7177	0.6853
LSTM + hMuLab	0.6647	0.7371	0.6716
eLh + LSTM + hMuLab	0.6915	0.7358	0.6894
eLh + LSTM + hMuLab + Tab	0.6928	0.7538	0.7072
eLh + LSTM + hMuLab + 3 × Tab	0.6952	0.7575	0.7095

As far as the fusion methods are concerned, eLh outperforms the standalone Lh, but it is from the fusion of different approaches that we gain the best improvements: the best ensemble is eLh + LSTM + hMuLab + 3 × Tab, which is the fusion of methods with the greatest diversity, compared to the others. This ensemble produces the highest performance in this classification problem, outperforming all the standalone approaches for each of the three descriptors.

In the second experiment (see Table 2), fusion at the feature level is tested. The starting descriptor is the concatenation of two or three sets of features for the Tab approach, while for the other classifiers, the combination is the average rule applied to each of them (e.g. LSTM trained on DDI combined by average rule with LSTM trained on NRAKEL).

When a cell in Table 2 spans more than one column, that indicates that the related classifier is trained using more features, and, for each feature, a different classifier is trained with results fused using the average rule.

Moreover, we have run the following test, for each fold we run a grid search using an internal 2-fold in the training data set, the following hyper parameters have been tested, choosing the best ones in each iteration:

$$numHiddenUnits = \{50,\ 100,\ 150\};\ miniBatchSize = \{10\ 20\ 30\};\ \alpha = \{0.4\ 0.5\ 0.6\};$$
$$K = \{10,\ 15,\ 20\}.$$

We named the new approach (eLh + LSTM + hMuLab + 3 × Tab)_hyperp in Table 2.

The results reported in Table 2 show the usefulness of the ensembles: all the approaches that contain Tab outperform the Fast.AI research group, which up to now had achieved the highest classification score.

5. Discussion

This study proposed an effective ensemble approach to classify novel chemicals/drugs according to first-level ATC classification. The best ensemble proposed in this work is the fusion of the four classifiers EnsATC, which is eLh + LSTM + hMuLab+3×Tab (where 3×Tab indicates a weight equal to the sum of the other three). The performance of EnsATC in terms of absolute true (the most used performance indicator for this problem) is notable compared to other approaches in the literature.

To demonstrate the performance enhancement of EnsATC, we report in Table 3 the results of several state-of-the-art classifiers in terms of the following five performance

Absolute true	DDI	NRAKEL		FRAKEL
LSTM		0.6823		0.6330
hMuLab		0.6991		0.5977
Lh		0.7201		0.6709
eLh		0.7430		0.6853
Tab		0.7667		0.6760
Tab			0.7734	
eLh + LSTM + hMuLab		0.7577		–
eLh + LSTM + hMuLab			0.7762	
eLh + LSTM + hMuLab + Tab			0.7901	
eLh + LSTM + hMuLab + 2 × Tab			0.7919	
eLh + LSTM + hMuLab + 3 × Tab			*0.8009*	
(eLh + LSTM + hMuLab + 3 × Tab)_hyperp			*0.8091*	

Table 2. Combinations of descriptors (absolute true rates) achieved by the ensembles using combinations of features

Table 3.
Comparison of the best ensemble here with the best reported in the literature

Method	Aiming	Coverage	Accuracy	Abs. True	Abs. False
EnsATC	0.9139	0.8432	0.8338	0.8009	0.0131
Chen *et al.* [7]	0.5076	0.7579	0.4938	0.1383	0.0883
EnsANET_LR [13]	0.7536	0.8249	0.7512	0.6668	0.0262
EnsLIFT [11]	0.7818	0.7577	0.7121	0.6330	0.0285
iATC-mISF [9]	0.6783	0.6710	0.6641	0.6098	0.0585
iATC-mHYb [10]	0.7191	0.7146	0.7132	0.6675	0.0243
iATC_Deep-mISF [27]	0.7470	0.7391	0.7157	0.6701	0.0000
NRAKEL [26]	0.7888	0.7936	0.7786	0.7593	0.0363
FRAKEL [16]	0.7851	0.7840	0.7721	0.7511	0.0370
NLSP [15]	0.8135	0.7950	0.7828	0.7497	0.0343
FUS3 [14]	0.8755	0.6973	0.7346	0.6871	0.0238
CGATCPred [17]	0.8194	0.8288	0.8081	0.7658	0.0275

indicators: aiming, coverage, accuracy, absolute true, and absolute false. Examining Table 3, it is clear that our ensemble strongly outperforms the other approaches: EnsATC gains the highest Absolute true and the highest classification accuracy. Note the performance differences reported here compared to those reported in the original papers on NRAKEL [26] and FRAKEL [16]. The main reason for these differences is that the classifiers are not optimized here because we are training on a single data set. Our concern is to avoid any risk of overfitting; thus, we run the approaches using default values.

6. Conclusion

Since ATC classification is a difficult multilabel problem, the goal of this study was to improve performance by generating ensembles trained on three different feature vectors. The original input vectors were fed into a BiLSTM, which functioned (with modification) not only as a multilabel classifier but also as a feature extractor, with features taken from the output layer.

Two other classifiers aside from LSTM were evaluated: one based on multiple linear regression; and another, a deep learning technique for tabular/structured data based on the creation of some embedding layers for categorical variables. To boost the performance of these classifiers, they were trained on the feature sets with results fused by average rule. Comparisons of the best ensembles were made with the standalone classifiers and other notable systems. Results showed that EnsACT, the top-performing ensemble constructed by the method proposed here, obtained superior results for ATC classification using five performance indicators.

Future work will explore the performance of different LSTM and CNN topologies combined with many activation functions. The fusion of other deep learning topologies for extracting features will also be the focus of a future investigation.

References

1. Wouters OJ, McKee M, Luyten J. Estimated research and development investment needed to bring a new medicine to market, 2009-2018. JAMA. 2020; 323(9): 844-53.
2. Pitts RC. Reconsidering the concept of behavioral mechanisms of drug action. J Exp Anal Behav. 2014; 101: 422-41.
3. Wong CH, Siah KW, Lo AW. Estimation of clinical trial success rates and related parameters. Biostatistics. 2019; 20(2): 273-86.

4. MacDonald K, Potvin K. Interprovincial variation in access to publicly funded pharmaceuticals: a review based on the WHO anatomical therapeutic chemical classification system. Can Pharm J/ Revue des Pharmaciens du Canada. 2004; 137(7): 29-34.
5. Dunkel M, Günther S, Ahmed J, Wittig B, Preissner R. SuperPred: update on drug classification and target prediction. Nucleic Acids Res. 2008; 36(May): W55-W9.
6. Hochreiter S, Schmidhuber J. Long short-term memory. Neural Comput. 1997; 9(8): 1735-80.
7. Chen L. Predicting anatomical therapeutic chemical (ATC) classification of drugs by integrating chemical-chemical interactions and similarities. PLoS One. 2012; 7(e35254).
8. Wu L, Ai N, Liu Y, Fan X. Relating anatomical therapeutic indications by the ensemble similarity of drug sets. J Chem Inf Model. 2013; 53(8): 2154-60.
9. Cheng X, Zhao SG, Xiao X, Chou KC. iATC-mISF: a multilabel classifier for predicting the classes of anatomical therapeutic chemicals. Bioinformatics. 2017; 33(3): 341-6.
10. Cheng X, Zhao SG, Xiao X, Chou KC. iATC-mHyb: a hybrid multilabel classifier for predicting the classification of anatomical therapeutic chemicals. Oncotarget. 2017; 8: 58494-503.
11. 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. 2017; 33: 2837-41.
12. Schmidhuber J. Deep learning in neural networks: an overview. Neural Network. 2015; 61: 85-117.
13. Lumini A, Nanni L. Convolutional neural networks for ATC classification. Curr Pharm Des. 2018; 24(34): 4007-12.
14. Nanni L, Brahnam S, Lumini A. Ensemble of deep learning approaches for ATC classification. In: Satapathy SC, Bhateja V, Mohanty JR, Udgata SK (eds). Smart intelligent computing and applications - proceedings of the third international conference on smart computing and informatics. 159. Singapore: Springer; 2020. p. 117-25.
15. Wang X, Wang Y, Xu Z, Xiong Y, Wei DQ. ATC-NLSP: prediction of the classes of anatomical therapeutic chemicals using a network-based label space partition method. Front Pharmacol. 2019; 10: 971.
16. Zhou JP, Chen L, Wang T, Liu M. iATC-FRAKEL: a simple multilabel web server for recognizing anatomical therapeutic chemical classes of drugs with their fingerprints only. Bioinformatics. 2020; 36(11): 3568-9.
17. Zhao H, Li Y, Wang J. A convolutional neural network and graph convolutional network-based method for predicting the classification of anatomical therapeutic chemicals. Bioinformatics. 2021; 37(18): 2841-7.
18. Wang P, Ge R, Xiao X, Zhou M, Zhou F. hMuLab: a biomedical hybrid MUlti-LABel classifier based on multiple linear regression. IEEE ACM Trans Comput Biol Bioinf. 2017; 14(5): 1173-80.
19. Howard J, Fastai GS.. A layered API for deep learning. Information. 2020; 11(2): 108.
20. Ogata H, Goto S, Sato K, Fujibuchi W, Bono H, Kanehisa M. KEGG: kyoto encyclopedia of genes and genomes. Nucleic Acids Res. 1999; 27(1): 29-34.
21. Cho H, Berger B, Peng J. Compact integration of multi-network topology for functional analysis of genes. Cel Syst. 2016; 3(6): 540-8. e5.
22. Chou KC. Some remarks on predicting multilabel attributes in molecular biosystems. Mol Biosyst. 2013; 9: 10922-1100.
23. . Kimura K, Sun L, Kudo M. MLC toolbox: a MATLAB/OCTAVE library for multilabel classification. ArXiv. 2017; arXiv:1704.02592.
24. Zhang ML, Wu L. Lift: multilabel learning with label-specific features. IEEE Trans Pattern Anal Mach Intell. 2015; 37(1): 107-20.
25. Kumar V, Pujari AK, Padmanabhan V, Kagita VR. Group preserving label embedding for multilabel classification. Pattern Recognit. 2019; 90: 23-34.

26. Zhou JP, Chen L, Guo ZH. iATC-NRAKEL: an efficient multilabel classifier for recognizing anatomical therapeutic chemical classes of drugs. Bioinformatics. 2019; 36(5): 1391-6.

27. Lu Z, Chou KC. iATC_Deep-mISF: a multilabel classifier for predicting the classes of anatomical therapeutic chemicals by deep learning. Adv Biosci Biotechnol. 2020; 11: 153-9.

Corresponding author
Sheryl Brahnam can be contacted at: sbrahnam@missouristate.edu