International Journal of Computing and Digital Systems
ISSN (2210-142X)
Int. J. Com. Dig. Sys.13, No.1 (May-23)

http://dx.doi.org/10.12785/ijcds/1301107

Performability of Deep Recurrent Neural Networks for Molecular Sequence data

Roshan R. Kotkondawar[1], Sanjay R. Sutar[2], Arvind W. Kiwelekar[3] and Hansaraj S. Wankhede[4]

[1]*Department of Information Technology, Dr. Babasaheb Ambedkar Technological University Lonere, Raigad 402103, India*
[2]*Department of Information Technology, Dr. Babasaheb Ambedkar Technological University Lonere, Raigad 402103, India*
[3]*Department of Computer Engineering, Dr. Babasaheb Ambedkar Technological University Lonere, Raigad 402103, India*
[4]*Department of Artificial Intelligence, G. H. Raisoni College of Engineering, Nagpur 440016 India*

Received 5 Aug. 2022, Revised 30 Apr. 2023, Accepted 14 May. 2023, Published 30 May. 2023

Abstract: Artificial Intelligence (AI) has appeared as a life-changing innovation in recent years transforming the conventional problem-solving strategies adopted so far. ML and DL-based approaches are making a monumental impact in the fields of life sciences and health care. The tremendous amount of biochemical data has set off leading-edge research in health care and Drug Discovery. Molecular Machine Learning has precisely adopted ML techniques to uncover new insights from biochemical data. Biochemical datasets essentially hold text-based sequential information about molecules in several forms. Simplified Molecular Input Line Entry System (SMILES) is a highly efficient format for representing biochemical data that can be suitably utilized for countless relevant applications. This work presents the SMILES molecular representation in a nutshell and is centered on the major applications of ML and DL in health care especially in the drug discovery process using SMILES. This work utilizes a sequence-to-sequence architecture built on Recurrent Neural Networks (RNNs) for generating small drug-like molecules using the benchmark datasets. The experimental results prove that the Long Short Term Memory (LSTM) based RNNs can be trained to encode the raw SMILES strings with nearly perfect accuracy and to generate similar molecular structures with minimal or no feature engineering. The gradient-based optimization strategy is applied to the network and found distinctly suited to assemble the most stable and proficient sequence model. RNNs can thus be employed in Drug Discovery activities like similarity-based virtual screening, lead compound finding, and hit-to-lead optimization.

Keywords: Healthcare, Artificial Intelligence, Drug Discovery, Deep Learning, SMILES, Recurrent Neural Network, Long Short Term Memory

1. INTRODUCTION

Some phenomenal advances have been marked in fields of *life sciences* and *health care* for a decade concerning global health. *World Health Organization* has already prioritized research in life sciences and global health care. The said research can improve global health serving the world and this potential is evident [1]. The drug discovery process is closely related to human life and holds a significant share in health care.

The huge success of AI-driven techniques like *ML* and *DL* in life sciences and health care has already set up a new technological evolution. AI-based approaches like molecular ML are doing remarkably well for *Computer aided Drug Design* [2] and *De novo Drug Design* [3]. One dominant reason behind this success is the skillful handling and processing of almost all forms of data. Statistical inspection of data gives all insight into patterns and associations in data. It can thus provide an appropriate means to use the ML or DL methods wisely based on analysis to generate more credible and precise results [4].

Text and *sequential* forms of data play a vital role in around all the applications of ML in health care and drug discovery. Some life science disciplines like *Chemistry*, *Biology*, *Genomics*, and *Bio-Chem-informatics* mostly work with biochemical or biomedical data symbolized in *molecular* configuration. Molecular data embeds many decisive aspects of a molecule like an arrangement or sequence of atoms, nature of bonds, topology in the case of molecular structure, and length of a sequence. A thorough interpretation of data should hence be ensured while working with this type of data.

The molecular representation of biochemical data appears in many forms like molecular descriptors, fingerprints, graph-based representation, topological descriptors, etc. Among all, the Simplified Molecular Input Line Entry System (SMILES) representation is most recognized as it covers the most valuable aspects of a molecule.

E-mail address: author 1, author 2, author 3

ML or DL-based models with an ability to handle these complex sequential data can be adopted to resolve the complex tasks of molecule generation [5] [6] [7], molecular property prediction [8] [9], drug-target interaction [10] [11] and many others. In fact, the most decisive usage of the SMILES strings for high-speed machine processing is an ongoing endeavor.

This work aims to confer the applications of ML and DL techniques in health care and drug discovery along with the usefulness of SMILES data to represent the molecules in the context of ML and DL-based approaches. In this study, we investigated novel ML and DL algorithms to deal with molecular data in health care and drug discovery. We further developed a DL-based model for generating small drug-like molecules by allowing the DL model to learn the molecular representation thoroughly. Deep RNNs are used for this purpose and the results are highly appreciable.

The paper is organized into a total of 7 sections. Section 2 briefly explains the language models in the context of Sequence data and presents a novel language to encase molecules called the Simplified Molecular Input Line Entry System (SMILES). Section 3 highlights the concept of Molecular ML concerning sequence data processing. Section 4 deals with the actual applications of molecular ML using SMILES data in the areas of Bio-chem-informatics and Drug discovery. Section 5 illustrates the implementation of the model in a step-wise manner. Section 6 spells out the results and findings from the experimentation. Section 7 concludes the paper by extending some prominent works in the near future.

2. Related Work

A. Language models and Sequence data processing

Language is one of the core aspects of human life without which we cannot communicate. A natural language is simply a collection of signs (symbols), expressions, and sounds accepted as a means of communication. *Natural Language Processing* (NLP), a separate field in AI is committed to developing expert systems for the analysis and manipulation of natural languages using statistical tools of AI.

The language-based data is always *textual* and *sequential*, hence known as *Sequence Data*. As an example a poetry that contains a series of words forming sentences wherein a specific sequence of words can either decide or change the meaning of a sentence. Some other examples can be stock price data, time-series sequence data, a movie review, or even a DNA sequence. Sequence data is continuous data where the arrangement and pattern of data matters for its correct interpretation and usage.

Sequence models signify a class of DL architectures that are highly compatible to work with the *Sequence data*. In a supervised sequence data problem where a labeled dataset (X, Y) is specified, either X or Y or both X and Y can be a sequence. Numerous applied problems like Speech Recognition, Sentiment Analysis, Image captioning, Named Entity Recognition, Machine Translation, Stock price prediction, and DNA sequence analysis come under the umbrella of sequence modeling. One inherent similarity in all these problems is the continuous nature of data. Hence the prior intuition of the data is a necessary aspect to get a precise solution. The analogy between the sequence data and the molecular data can be justified to formulate a sequence model with molecular data.

Significant work is observed recently regarding the use of sequence models with sequence data specifically by utilizing sequence-to-sequence (Seq2Seq) models built on deep *Recurrent Neural Networks (RNNs)*. Zhang et al. recently devised a Sequence (Seq2Seq) model for text-based speaker verification using deep neural networks [12]. Sequence models are found operative in Recommendation Systems (RS) for suggesting short textual conversations with social media dataset [13]. The sentiment analysis problem can be effectively addressed using RNNs and sequence models. Satapathy et al. worked on the task of micro-text analysis using the Seq2Seq DL model and acquired a lot of improved results [14]. A sequence-to-sequence algorithm has even been tested for cognitive Internet of Things to predict the likelihood of future events [15].

B. SMILES: A language of molecules

The inevitable role of language in the interpretation and analysis of data emphasizes a precise mechanism to convey the facts or findings from data. The emergence of *Big-Data* in *Bio-Chem-informatics* coupled with huge biomedical or biochemical data repositories has triggered the use of AI-driven approaches in molecular data processing.

Biochemical data must be precisely represented in the form pertinent to ML or DL algorithms and techniques. These notations are known as a chemical language or language of molecules. Among all chemical languages which are suitably used to represent the molecules Simplified Molecular Input Line Entry System (SMILES) is a highly established language or notation accepted by academia as well as industries for processing molecular data including chemical compounds.

The language typically consists of chemical notations adopted to symbolize and process chemical information. According to David Weininger, It denotes a molecular structure as a two-dimensional valence-oriented graph. This representation accepts postulates of molecular graph theory which utilizes small and natural grammar to characterize the structural details of a molecule or a compound more precisely [16]. There are precise line notations for the molecular structures where the individual example of this language is called a SMILES string. Figure 1 views the representation of Benzene, Isobutyric Acid, and Triethylamine respectively using SMILES showing the molecular graph and SMILES notations as well.

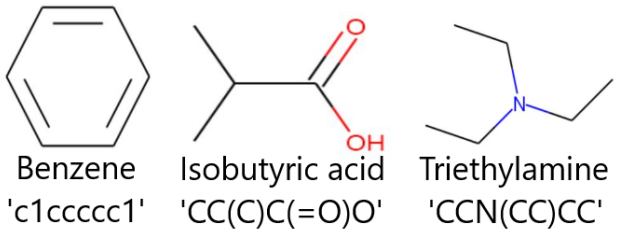

Figure 1. Examples of molecules and their SMILES representation.

1) SMILES: Rules and Specifications

SMILES notation follows a set of rules along with the elementary laws of Chemistry. The notational directives must be conformed while representing a molecule in any computer-recognizable form. Here we focus on the six primitive rules that must be ensured in any SMILES notation.

1) ***Atoms***:
An atom is a fundamental element of a chemical formula relating to a molecule or a compound. An atom is represented by its atomic symbol or alphabet as specified in a periodic table. *Aromatic* atoms are represented by *small letters* while *non-aromatic* ones by *capital letters*. If an atomic symbol contains more than one alphabet then the second letter is always a small alphabet.
As an example, atomic symbols C,O,N represents Carbon, Oxygen, and Nitrogen atoms respectively. Similarly, Cl and Mg represent Chlorine and Magnesium atoms accordingly. Hydrogen atoms are already implied in absence of brackets as well but non-hydrogen atoms are specified independently with their atomic symbol in a square bracket [16].
2) ***Bonds***:
A bond in a chemical compound is a linkage or a connection that holds atoms together. It can be either a single, double, triple, or aromatic bond. A single bond is default and needs not to be shown. The rest of the bonds are structurally shown by the following symbols.
   - Single bond : '-'
   - Double bond : '='
   - Triple bond : '#'
   - Aromatic bond : ':'
   - Disconnected structures : '.'
3) ***Chain of atoms / Strings***:
A sequence or chain in the formula is symbolized by joining atomic symbols and bond symbols together. The conventional notation is followed in the case of linear structures except for the fact that hydrogen atoms are not mentioned explicitly.
4) ***Branches***:
Branches in a chain are stated by confining them within parentheses. Branches can even be nested wherein the default link of expression is to the left of the branch. It clarifies that the expression will be placed directly after the symbol (atom) is connected. If there is a presence of a bond symbol, it is placed just after the left parentheses.
5) ***Rings / Cyclic structures***:
SMILES notation symbolizes cyclic structures by splitting a single bond in every ring [16] and uses numbers to denote the start and end of the ring atom. As an example, Cyclohexane (C1CCCCC1) shows such a cyclic or ring structure where the first carbon atom is linked with the last carbon atom by a single bond. In case a compound has numerous rings, different numbers are used to identify multiple rings.
6) ***Atomic Charge***:
Atomic charges play an important role in the structural specifics of a molecule. The atomic charge is placed in the bracket just next to the atomic symbol. Table I shows some specific examples of such SMILES strings depicting the use of all the above-mentioned rules of SMILES representation.

2) Advantages of SMILES language

The SMILES notation is equipped with prominent methods to represent a molecule or compound. Here we list a few key advantages of SMILES notation.

- The notation is aimed at the effective processing of chemical information employing algorithmic computing.
- It is an easily accessible, flexible, and machine (computer) interactive notation.
- The notation empowers the speedy processing of chemical information utilizing the computing skills of the machine.
- It is a dynamic lexical paradigm and SMILES strings can be used as the vocabulary of a chemical language.
- This representation saves space for stacking a chemical structure with a compact representation compared to other methods.
- An attribute influencing the usage of this representation is that unique SMILES do exists for a molecule since the name of the molecule is identical to its structure.

3. Molecular Machine Learning

Proficient and constructive processing of data is the crux of a ML algorithm. One important class of information is sequential data and molecular data is the best example of this class. The sequence can be a protein or DNA sequence, small molecular data, protein-protein, or protein-ligand interaction data in the context of molecular data. A *molecule* is the least known component of a substance that consists of two or more atoms linked together by chemical bonds. The branch of ML which tackles this special type of data is known as *Molecular Machine Learning*. Molecular ML

TABLE I. SMILES notation for some molecules following rules of notation.

Molecule name	Molecular Formula	SMILES notation
Ethane	CH_3CH_3	**CC**
Ethene	CH_2CH_2	**C=C**
Sodium Chloride	$NaCl$	**Na.Cl**
Hydrogen	H_2	**[H][H]**
Isobutyric Acid	$C_4H_8O_2$	**CC(C)C(=O)O**
Cyclohexane	C_6H_{12}	**C1CCCCC1**
Benzene	C_6H_6	**c1ccccc1**
Propanoic Acid (ionized)	$C_3H_6O_2$	**CCC(=O)O[-1]**

is essentially the art of using ML models to address challenging problems surrounding *molecular data*. Creation or generation of new novel molecules with the preferred profile is the core concern in *Chemistry* and *Material Sciences*. It needs continuous *random experimentation* in laboratories that oblige expert supervision with strong domain knowledge along with infrastructural resources. This process is indeed slow and incurs a huge cost. Molecular ML aims to fill the gap and streamline this traditional approach by utilizing ML predictive capabilities in the finest way to find new molecules with desired properties [17]. Molecular ML has a huge potential to address complex problems in the fields of Chemistry, Bio-Chem-informatics, Life Sciences, and Health Care which involves large integrative attributes.

The complicated structure of molecular data makes its analysis and processing a challenging task. Moreover, the raw molecular data must be translated to a suitable form compatible with ML or DL algorithm by a process called *Featurization* or *Feature Engineering*. Techniques like molecular descriptors or fingerprints look after converting the molecular data into a machine-readable featured representation and this process is termed as *molecular featurization*.

Some basic data structures such as *Molecular Graphs* already exist in literature representing a molecule as a set of nodes and edges wherein each node corresponds to an atom and an edge corresponds to a chemical bond. Some of the well-known molecular featurization techniques are *Extended connectivity Fingerprints (ECFPs)* [18], *Molecular Descriptors* and *Molecular graph convolutions* [19]. Selecting the best form of data representation in the context of the underlying scientific problem as well as the learning algorithm is still an ongoing topic of research for chemical systems [20].

The selection of the appropriate dataset and the featurization procedure pertinent to the problem addressed follows the choice of a learning algorithm. An exact combination of these essentials together will only give the desired results. A highly optimized ML or DL model can commit this task efficiently with desired results, hence optimization of the model is necessary. Gradient-based optimization techniques are mostly preferred and are very efficient as well. In addition to that a few procedures like Bayesian optimization [8] or Genetic algorithms [21] along with some special techniques like transfer learning and fine tuning [5][6][22] are found producing highly acceptable results.

Several such case studies addressing a variety of problems with some outstanding contributions in the area of molecular ML are gaining attention these days. We mention some of such case studies and the applicative areas of molecular ML in the next section.

4. Major applications of sequence modeling using SMILES

The applicative context of molecular ML is not merely restricted to the task of molecule generation and can resolve most complex problems in diverse areas like Chemistry, Genomics, Bioinformatics, Health care, and Drug Discovery. This section focus on the area of Drug Discovery as a major applicative area of molecular ML in health care and comments on some remarkable work witnessed in recent years.

A simple deep neural network or integration of the deep neural network with a learner like Long Short Term Memory (LSTM) is skilled to operate on molecular data and address challenging activities in drug discovery with certainty. *Sequence to Sequence* (Seq2Seq) learning has triggered a new momentum in the molecular ML domain by adopting deep RNNs-based sequence models.

Here, we talk through a few noticeable issues addressed by sequence models using SMILES data.

A. *Molecule generation*

The creation of novel molecules with the desired drug-like properties and positive therapeutic effect is a baseline activity in Drug Discovery. Gupta et al. [22] invented a generative RNNs-based model coupled with LSTM to address the de novo design of novel chemicals with high accuracy which was then fine-tuned to generate molecule libraries, hit to lead optimization and fragment-based drug discovery. In a similar work, Segler et al. proved that RNNs with LSTMs can be adopted for the creation of novel molecules with similar properties and even be used to generate libraries for virtual screening [5].

Recently Jacob Yasonik [6] worked successfully on De novo Drug Design using deep RNNs for the generation and analysis of novel drug-like molecules using transfer learning and fine-tuning. This study took on the issues of scalability and multi-objectivity of the molecules in drug discovery and design. A SMILES based Seq2Seq neural network using bidirectional LSTM is constructed by Sattarov et al. to generate focused molecular libraries [23]. The RNNs-based heteroencoder model is devised in 2018 which not only generates the molecules but precisely predicts similarities among the molecules [24].

B. *Molecular property prediction*

The drug-like molecules created as a result of drug design operation must reflect some properties to be accepted as a *novel drug*. The properties can be related to the *pharmacokinetic* or *pharmacodynamic* profile, safety, toxicity, solubility, and efficacy of the drug. Besides, properties like molecular weight and *lipophilicity* of the compound should be attentively monitored concerning the compound creation adopting new technology trends [25].

Only after validating the presence of *desirable effects* and absence of side effects are the drug-like compounds subjected to further process. For example, the drug-like molecules so formed should be soluble in water or other body fluids for their effective reactions to show desirable effects. At the same time, these molecules should not produce toxic effects after consumption. Such properties hence are first verified thoroughly for the acceptance of drug-like compounds as novel drug.

DL-based sequence models are found superior for predicting molecular properties as well. Zheng Xu et al. put forward a Seq2Seq model comprising deep RNNs and Gated Recurrent Units (GRU) to symbolize SMILES strings in the computer-recognizable form which was further utilized fruitfully for solubility prediction task [9]. In a similar work, Goh et al. invented a SMILES based deep network using RNNs to predict molecular properties like solubility, toxicity, bioactivity, and solvation energy with outshining results [8]. A Convolutional Neural Networks (CNNs) based semi-supervised DL model was recently trained and fine-tuned for property prediction using SMILES datasets [26].

C. *Protein-ligand interaction*

Protein-ligand interaction (PLI) comes under the umbrella of drug-target interaction (DTI). A drug-like compound is termed as a *ligand* before its recognition as a drug. Moreover, most of the drugs target or bind to *proteins* including enzymes and receptors in the human body. This clears the influence of PLI activity during drug discovery. Successful detection of DTI is critical for both drug discovery and drug repositioning.

This cryptic activity has been addressed by deep sequence models in recent times. Due to the rapid explosion of biochemical data, researchers have started adopting DL methods for this task. Key factors that have catalyzed the use of DL-based approaches for DTI are highly efficient frameworks like TensorFlow coupled with processing speed offered by GPUs.

A deep RNNs model based on LSTM has been recently devised to monitor and find potent DTIs utilizing protein sequences with notable findings [10]. Similarly, a deep interpretive neural network was proved effective to predict DTIs using SMILES drug data [27]. In recent work, SMILES strings were operated along with protein data using the self-attention mechanism to accurately predict the DTIs [28].

D. *Virtual screening*

In the drug discovery pipeline, *Screening* is a mechanized process that selects compounds having the desired biological activity also known as *lead compounds* or *leads* to make the *admissible drug*. Recently, *virtual screening* has gained attention and is a highly operational phenomenon in computational drug discovery. The virtual screening technique is centered on detecting the leads which are proficient in binding a target molecule specifically a protein [29]. Two categories of virtual screening are (i) Ligand-Based Virtual Screening (LBVS) and (ii) Structure-Based Virtual Screening (SBVS).

DL methods have proved productive for virtual screening because the DL algorithms added with the relevant datasets can detect novel molecules much faster. DNNs have been already employed for the ligand-based approach [30] and structure-based approaches [31]. DL-based model using LSTMs was perfectly formulated to address molecule generation as well as virtual screening tasks in drug discovery. The model proved fruitful for generating the molecules and also to detect the activity of drug-like molecules for a given target [5].

5. Proposed method of molecule generation using RNNs

A. *Overview*

The drug discovery process is centered on the creation of potent drug molecules having desirable activity towards a specific target. The small drug molecules are preferably characterized using SMILES strings. Here, we devise a sequence-to-sequence architecture with RNNs capable of encrypting a sequence of SMILES strings from the dataset to a machine-readable vectorized setup. The theme of this work lies around the usage of raw SMILES strings from the dataset with minimal feature engineering to save the time required for featurizing the SMILES strings into a machine-readable form. The step-wise sequence of the research is shown in figure 2.

The applicability of this network being SMILES strings can be re-engineered to the same sequence. Moreover, similar molecular structures can be generated using the same architecture following the principle of molecular similarity in hidden or latent space. This work aims to employ LSTM-based RNNs to generate a set of similar drug-like molecules

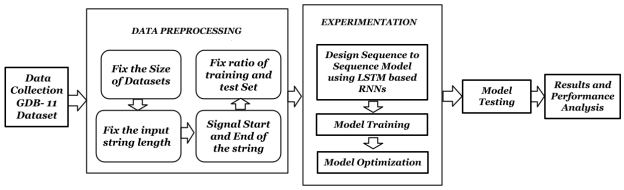

Figure 2. A detailed flowchart showing the research process.

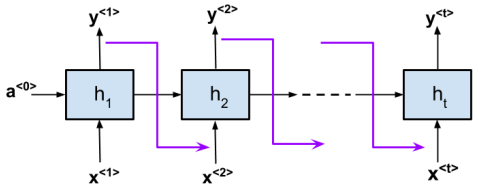

Figure 3. Teacher forcing process in RNNs.

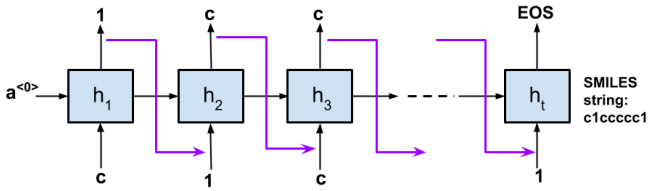

Figure 4. Teacher forcing process for benzene molecule.

Figure 5. Sequence learning in RNNs employing teacher forcing.

using the two datasets from the Generated Databases (GDB) data repository.

B. Sequence Modeling using RNNs and LSTM

RNNs stand strong among the DL architectures technically functional to act on sequence data. RNNs are the class of unsupervised DL algorithms capable of learning the relationship in input sequence fed to the network at specific instances. To master this complex relationship, RNNs uphold a state vector or memory administered by connecting the hidden layers of current input and previous input, as shown in Figure 5. Thus, the output of RNNs at an instance is a function of current input data as well as the input data observed up to that instance. The generalized expression for the output of RNNs at a particular instance can be mathematically represented as follows,

$$Y^t = Activation(dot(W_1, input^t) + dot(W_2, output^{t-1}) + b)$$

Here,Y^t represents the output at particular instance, W_1 and W_2 are weight parameters for inputs at instances **t** ($input^t$) and output for previous instance **t-1** ($output^{t-1}$) respectively. Finally, **b** represents a biased term. A suitable activation function (Activation) is applied based on the problem addressed.

A refined form of RNNs called RNNs with *LSTM* (Long Short Term Memory) is favored for sequence modeling considering the *vanishing gradient* problem in the case of RNNs. LSTM uses a memory unit to remember long-term dependencies between data items in a sequence. A special technique called *Teacher Forcing* [32] is employed during training of the RNNs-based model for faster and more efficient learning. Teacher Forcing in LSTMs is a procedure in which the input at each time step or instance is given as the actual output from the previous instance as depicted in figure 3.

Consider the SMILES string for *Benzene* molecule **c1ccccc1** is fed as an input to the network of LSTMs. Figure 4 views how exactly the network learns relationships in SMILES representation of the Benzene molecule using the teacher forcing method. Teacher forcing strategy helps for faster learning by improving model predictive skills and stabilizing the model for smoother convergence.

C. Strengths of RNNs for sequence modeling

RNNs have appeared as the most preferred deep learning architectures to design a sequence model by both academia and industrial communities in recent times. Some key aspects behind the growing acceptance of RNNs are mentioned below.

- RNNs are one of the most preferred architectures to work with textual or character-based sequence data like SMILES strings.
- RNNs are capable of working with varying lengths of textual data.
- RNNs can learn the representation even with minimal or no feature engineering over the input.
- RNNs can even learn the long-term dependencies in a sequence of input.
- The weights of the network can be effectively shared through time. Sharing weights at different positions might be helpful for a specific position to learn directly from whatever was learned at some other position in a sequence.
- A well-trained and optimized recurrent neural network can learn the most complex relationships among data with excellent accuracy.

D. Sequence to Sequence architecture

The sequence model adopted in this work comes under the umbrella of *many to many* RNNs which are also referred to as *Encoder-Decoder* architecture or *Autoencoders* [33]. Figure 6 shows off the schematic of the sequence-to-sequence model employing the Encoder-Decoder architecture.

1) Encoder network

An encoder network is typically built as RNNs which could either be LSTM or GRU (Gated Recurrent Unit) network. In this work, an encoder network is built as a

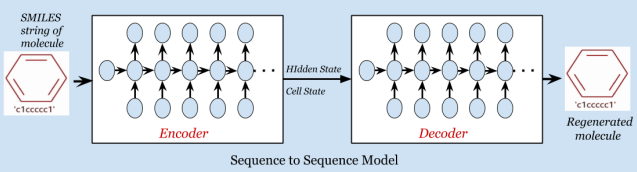

Figure 6. Architecture of the Sequence model.

series of LSTM modules. The input SMILES strings are sequentially fed to the encoder network one string at a time. After browsing the input sequence, the encoder network then outputs the vector known as the *internal state* vector which includes *hidden state* ($h_{<t>}$) and *cell state* ($c_{<t>}$) in the case of LSTMs.

The vectors ($h_{<t>}$ and $c_{<t>}$) generated by the encoder at the last instance after reading entire input sequences summarize or encode the input and are referred to as *Encoding* for the input. The input encodings are kept up discarding the outputs of the encoder network since the output sequence is formed tracking the input encodings only.

2) Decoder network

Similar to the encoder network, the decoder network is also built as a series of LSTM units that receive encodings from the encoder as input. Hence the initial decoder states ($h_{<0>}$ and $c_{<0>}$) are actually set to the final states of the encoder network. This network is then trained to generate the output strings calling upon the initial states. The decoder network processes each sequence in two stages called *Training* and *Inference* operating diversely in both stages.

Teacher forcing technique is followed during training for learning the long-term dependencies in the given strings faster. This method not only helps the decoder to efficiently learn the exact alliance in the strings but also accelerates the learning. A variant of the Back-propagation algorithm known as *Back-propagation Through Time* or BPTT is utilized for updating the weights of the network through the instances.

The decoder network after training is then used for predicting the output sequences in the inference stage. During Inference, the decoder operates recursively and generates one output string at a time operating through instances. Here, the predicted output at each instance is given as input to the next instance in the decoder network.

E. Dataset

The dataset selected for this work belongs to huge Generated Databases (GDB) repository called the GDB-11 dataset [34][35]. GDB database is an open-access database designed by *Reymond Research Group* at the University of Bern in Switzerland. It is a universal chemical dataset that contains nearly 977 million chemical compounds with at most 13 atoms of Carbon (C), Nitrogen (N), Oxygen (O), Chlorine (Cl), and Sulphur (S). The GDB datasets enumerate the small chemical compounds considering all the essential properties such as the possible number of molecules allowed, configuration, and functional groups [36].

In our work, we have experimented with two GDB-11 datasets to justify the utility of RNNs. GDB-11 dataset comes in the category of a small molecular dataset and contains molecules with a fairly small value of molecular weight [34]. The dataset is potentially relevant to be utilized in drug discovery activities since molecules exhibit few properties to be called *drug-like* or *lead-like* compounds [35]. In fact, both these datasets are very useful in small molecular design as well as synthesis [36].

It lists small organic compounds and molecules containing at most 11 atoms specifically Carbon (C), Nitrogen (N), Oxygen(O), and Fluoride (F) represented in canonized SMILES notation. All the molecules in the GDB-11 dataset follow the notion of valency,chemical stability, and synthetic feasibility. Following two GDB-11 small molecular datasets are utilized for implementation in this work,

1) GDB-11 dataset having number of atoms = 8
2) GDB-11 dataset having number of atoms = 9

The terms dataset-1 and dataset-2 will refer to the above-listed datasets in the coming sections for ease of writing.

F. Experiments

1) Data Preprocessing

One highest perks of employing RNNs for sequence data is that preprocessing is minimal since RNNs are well suited for such data. The experiments are carried out with two variations of the GDB-11 SMILES dataset. The variations in the number of atoms of two GDB-11 datasets are purposely picked to supervise the efficiency of the sequence model and to generalize the model.

There are a total of 66,706 input strings in dataset-1. To keep the data size nearly equal, the first 70,000 strings from dataset-2 are selected. 80% of data items are operated in training while 20% in testing. Hence, 53,364 strings from dataset-1 were utilized during training and 13,342 in testing. Likewise, 56,000 strings from dataset-2 were utilized during training and 14,000 in testing.

The input sequences are also used to find the molecular structure of a particular compound typically shown by a *structural graph*. Inter-conversion between the string and the molecular structure is performed using the python RDKit library. *RDKit* is a python library designed especially to develop DL applications concerning Bio-Chem-informatics [37]. The proposed sequence-to-sequence model is indeed trained in a specific environment activated using Rdkit called python *rdkit environment*.

Both datasets are first studied to note all the *textual*, *numeric*, and *special symbols* present in these strings. All

these symbols denote a Character Set which will be put through further processing. It is necessary to signal the beginning and end of each input sequence during the encoding and decoding operation of the sequence model. Hence, the symbols I and X is added to indicate the beginning and end of strings.

Finally, before encoding the input data into the vectorized form, the length of all strings is set to a value greater than the maximum string length. The maximum length for a string is fixed to 28 for dataset-1 and 34 in the case of dataset-2. This setting will indeed generate the encoding vectors of equal length. Above mentioned are the only preprocessing steps done before training the sequence model.

2) Model building

The sequence-to-sequence model is implemented using the *Keras* high-end functional application programming interface (API) that works on top of a system specially designed for DL applications named *TensorFlow* [38].

The Encoder network is built by deploying a chain of LSTM layers followed by the dense layer. The size of LSTM used during training is 64. Only the hidden states of the encoder network are saved using the return_state argument of LSTM and concatenated together using the dense layer activated by *ReLU* activation function. This encoded information is then analyzed by the decoder network to predict the output sequences.

To develop the sequence model the *Model* utility of Keras is utilized. The Model utility takes a combination of the source or input string as well as the target or output string (shifted by one time-step) together as an input of the model. Hence, the input is specified as a string and shifted form of the same string during training the model to accurately predict the output. The encoded information is passed through the series of Dense layers in the decoder network with *ReLU* activation function which decodes the hidden states of the encoder network. This information then flows through the chain of LSTM layers followed by a Dense layer with a *Softmax* activation function. The LSTM layer coupled with the Dense layer predicts the output data string one character at an instance.

3) Model training and optimization

This work employs the gradient-based optimization policy to train the Encoder-Decoder network for both datasets. The overall model loss (cross-entropy loss function) is regularly monitored in parallel with the SMILES string reconstruction accuracy during the whole training process. Hyper-parameters including the learning rate is adjusted at random in training till the global minimum is reached.

Numerous configurations of hyper-parameters especially learning rate, batch size, LSTM size, and the number of epochs is verified to acquire an optimized model that minimizes loss attaining maximum reconstruction accuracy. The dataset is shuffled through training since it helps in faster training and prevents overfitting by avoiding bias as well. Small batch sizes are preferred for the mini-batch gradient descent algorithm to decrease the error in model generalization.

The model is trained for 200 epochs using Adam optimizer wherein the learning rate is altered from an initial value of 0.001 to optimize the hyper-parameters. The value of the learning rate is subsequently lowered in case the training hibernates by constantly monitoring the validation loss. The model is analyzed after each epoch using the holdout cross-validation set. The best model with the lowest loss and superior molecule reconstruction accuracy is accepted.

Two distinct models are trained following this procedure intended for the above-mentioned datasets. The conclusive set of optimized hyper-parameter values concerning both datasets are mentioned in the table II. In a similar context, Figure 9 articulates the estimate of loss all along the training process over dataset-1 as depicted in Figure 7 and dataset-2 as depicted in Figure 8.

TABLE II. Hyper-parameters for optimized model

Hyper-parameter	**Dataset**	
	Dataset-1	Dataset-2
Learning rate	0.0025	0.002
Batch size	196	128
Optimizer	Adam	Adam
Number of epochs	200	200

The *Transfer Learning* technique with *weight initialization* strategy is then adopted to generate the molecules in hidden (latent) space. The *latent* or *hidden* space in the simplest terms is the space that holds SMILES strings in the vectorized form. The decoder network is defined by inheriting the learned weights from the trained model and used to generate the molecules using the test sets.

6. Results and Findings

The prime outcome of this study strongly validates the efficacy of RNNs and LSTM to encase the association in SMILES strings from datasets. The sequence model adopted in the study performs exceptionally well to encode the information stored in the input strings and to reconstruct the same. Every valuable detail like atomic position, bonding style, string length, and the atomic charge in the SMILES string is well-preserved by the sequence model.

The trained sequence model is verified to check the accuracy in encoding the SMILES strings on the test set and the results are highly optimistic. Only nine among 13342 SMILES strings are incorrectly regenerated for dataset-1 achieving nearly 99% accuracy on the test set. In a similar sense, only 15 among 14000 SMILES strings are incorrectly regenerated for dataset-2 achieving nearly 99% accuracy on the test set. These results are cross-checked for model overfitting but as the accuracy is related to molecule regener-

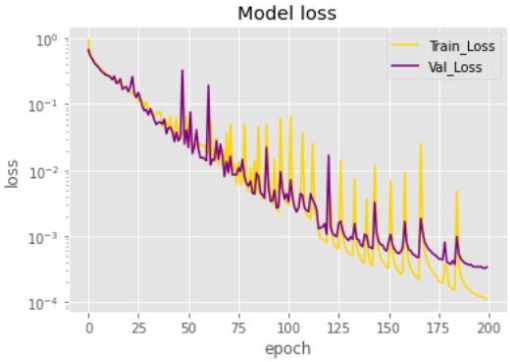

Figure 7. Dataset-1

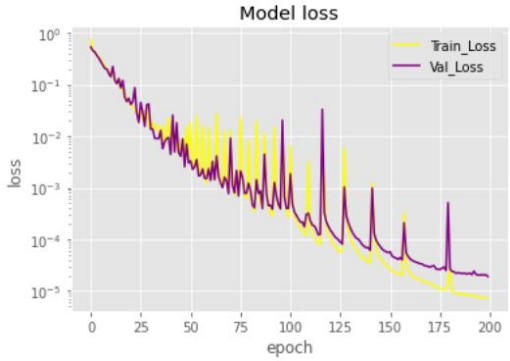

Figure 8. Dataset-2

Figure 9. Estimate of Model Loss

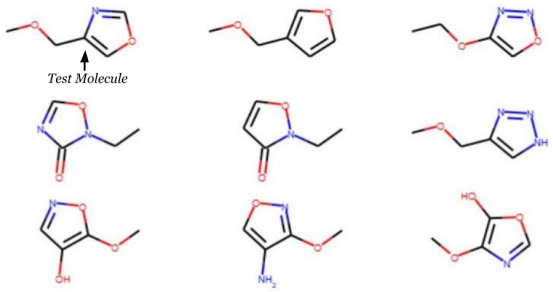

Figure 10. Dataset-1

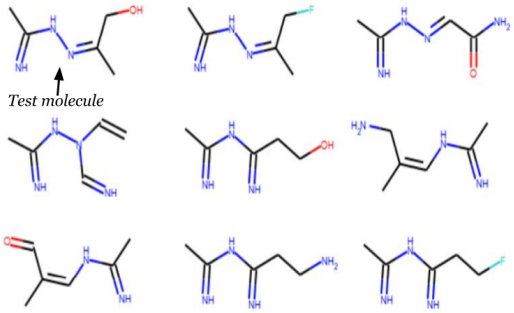

Figure 11. Dataset-2

Figure 12. Molecular and substructure similarity for GDB-11 datasets

ation only, the findings confirm the expertise of RNNs to restructure the mapping among individual SMILES strings.

The second outcome is associated with molecular and substructure similarity that has a direct impact on drug discovery activities. According to the principle of molecular similarity, compounds (molecules) having similar structures do exhibit similar *physiochemical* and *biological* properties [39].

The molecules generated using the sequence model show the similarity in the context of substructures, atomic positions, and bonding patterns in their hidden or latent space. Figure 12 displays the similarity in the generated molecular structures concerning the test molecule and adjoining molecular structures in said context for dataset-1 (figure 10) and dataset-2 (figure 11). This validates that similar molecules hold similar representations in the hidden space.

7. Conclusion and Future Scope

The initial part of this work investigates the SMILES notation to characterize the biochemical sequence data and highlights the applicative areas of DL in the drug discovery domain with the biochemical data. The SMILES language is reliable to encase the information of the molecule or compound considering all the aspects like atoms, bonds, bonding patterns, atomic charges, rings or cycles, and branches. The Sequence to Sequence models designed on the top of RNNs and LSTMs are skilled enough to take molecular ML a step away with the help of biochemical Big-data.

In the subsequent part of this work, a sequence model is successfully employed on the small molecular dataset to encode and reconstruct the molecules. The accuracy with which these small molecules are reconstructed proves the usefulness of RNNs in processing molecular data. Since the drug molecules are mostly represented by SMILES notation, the sequence modeling will surely assist the drug discovery process to encode and generate drug-like molecules. This work validates the utility of SMILES notation along with RNNs and LSTMs to design a prototype that can work with drug-like molecules at ease.

The model is also functional to generate similar molecular substructures in hidden space concerning atomic positions and bonding patterns in the molecule. Similar molecules exhibit similar bio-physical activities. A molecule or a group of molecules can be easily examined for an explicit pattern to predict the activity of that molecule toward a specific target.

It will surely help to find the molecules active towards a specific molecule or to generate such molecules with less experimentation and can surely benefit drug discovery in the areas of lead structure finding, hit-to-lead optimizations, and even generating lead molecules or libraries. Mechanization of the process is often acknowledged in fields of Bio-Chem-

informatics and drug discovery as it simplifies the tedious task of testing an individual compound by an expert. RNNs-based deep learning models have the potential to mechanize and accelerate the complex tasks in drug discovery which will surely save cost and time.

8. Acknowledgements

The following blogs from Keras documentation have been referred to implement the sequence-to-sequence model in Keras.

- https://keras.io/examples/nlp/lstm_seq2seq/
- https://blog.keras.io/a-ten-minute-introduction-to-sequence-to-sequence-learning-in-keras.html

A. Abbreviations and Acronyms

Artificial Intelligence: AI, Machine Learning: ML, Deep Learning: DL, Recurrent Neural Network: RNN, Long-Short Term Memory: LSTM, Simplified Molecular Input Line Entry System: SMILES, Sequence to Sequence: Seq2Seq

References

[1] S. R. Hanney and M. A. González-Block, "Health research improves healthcare: now we have the evidence and the chance to help the who spread such benefits globally," pp. 1–4, 2015.

[2] S. Surabhi and B. Singh, "Computer aided drug design: an overview," *Journal of Drug Delivery and Therapeutics*, vol. 8, no. 5, pp. 504–509, 2018.

[3] M. Hartenfeller and G. Schneider, "De novo drug design," in *Chemoinformatics and computational chemical biology*. Springer, 2010, pp. 299–323.

[4] B. Ratner, *Statistical and Machine-Learning Data Mining:: Techniques for Better Predictive Modeling and Analysis of Big Data*. CRC Press, 2017.

[5] M. H. Segler, T. Kogej, C. Tyrchan, and M. P. Waller, "Generating focused molecule libraries for drug discovery with recurrent neural networks," *ACS central science*, vol. 4, no. 1, pp. 120–131, 2018.

[6] J. Yasonik, "Multiobjective de novo drug design with recurrent neural networks and nondominated sorting," *Journal of Cheminformatics*, vol. 12, no. 1, pp. 1–9, 2020.

[7] S. Zheng, X. Yan, Q. Gu, Y. Yang, Y. Du, Y. Lu, and J. Xu, "Qbmg: quasi-biogenic molecule generator with deep recurrent neural network," *Journal of cheminformatics*, vol. 11, no. 1, p. 5, 2019.

[8] G. B. Goh, N. O. Hodas, C. Siegel, and A. Vishnu, "Smiles2vec: An interpretable general-purpose deep neural network for predicting chemical properties," *arXiv preprint arXiv:1712.02034*, 2017.

[9] Z. Xu, S. Wang, F. Zhu, and J. Huang, "Seq2seq fingerprint: An unsupervised deep molecular embedding for drug discovery," in *Proceedings of the 8th ACM international conference on bioinformatics, computational biology, and health informatics*, 2017, pp. 285–294.

[10] Y.-B. Wang, Z.-H. You, S. Yang, H.-C. Yi, Z.-H. Chen, and K. Zheng, "A deep learning-based method for drug-target interaction prediction based on long short-term memory neural network," *BMC Medical Informatics and Decision Making*, vol. 20, no. 2, pp. 1–9, 2020.

[11] M. Wen, Z. Zhang, S. Niu, H. Sha, R. Yang, Y. Yun, and H. Lu, "Deep-learning-based drug–target interaction prediction," *Journal of proteome research*, vol. 16, no. 4, pp. 1401–1409, 2017.

[12] Y. Zhang, M. Yu, N. Li, C. Yu, J. Cui, and D. Yu, "Seq2seq attentional siamese neural networks for text-dependent speaker verification," in *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2019, pp. 6131–6135.

[13] J. Torres, C. Vaca, L. Terán, and C. L. Abad, "Seq2seq models for recommending short text conversations," *Expert Systems with Applications*, vol. 150, p. 113270, 2020. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0957417420300956

[14] R. Satapathy, Y. Li, S. Cavallari, and E. Cambria, "Seq2seq deep learning models for microtext normalization," in *2019 International Joint Conference on Neural Networks (IJCNN)*. IEEE, 2019, pp. 1–8.

[15] P. Wu, Z. Lu, Q. Zhou, Z. Lei, X. Li, M. Qiu, and P. C. Hung, "Bigdata logs analysis based on seq2seq networks for cognitive internet of things," *Future Generation Computer Systems*, vol. 90, pp. 477–488, 2019.

[16] D. Weininger, "Smiles, a chemical language and information system. 1. introduction to methodology and encoding rules," *Journal of chemical information and computer sciences*, vol. 28, no. 1, pp. 31–36, 1988.

[17] B. Ramsundar, P. Eastman, P. Walters, and V. Pande, *Deep learning for the life sciences: applying deep learning to genomics, microscopy, drug discovery, and more*. " O'Reilly Media, Inc.", 2019.

[18] D. Rogers and M. Hahn, "Extended-connectivity fingerprints," *Journal of chemical information and modeling*, vol. 50, no. 5, pp. 742–754, 2010.

[19] S. Kearnes, K. McCloskey, M. Berndl, V. Pande, and P. Riley, "Molecular graph convolutions: moving beyond fingerprints," *Journal of computer-aided molecular design*, vol. 30, no. 8, pp. 595–608, 2016.

[20] F. A. Faber, L. Hutchison, B. Huang, J. Gilmer, S. S. Schoenholz, G. E. Dahl, O. Vinyals, S. Kearnes, P. F. Riley, and O. A. Von Lilienfeld, "Prediction errors of molecular machine learning models lower than hybrid dft error," *Journal of chemical theory and computation*, vol. 13, no. 11, pp. 5255–5264, 2017.

[21] N. J. Browning, R. Ramakrishnan, O. A. Von Lilienfeld, and U. Roethlisberger, "Genetic optimization of training sets for improved machine learning models of molecular properties," *The Journal of Physical Chemistry Letters*, vol. 8, no. 7, pp. 1351–1359, 2017.

[22] A. Gupta, A. T. Müller, B. J. Huisman, J. A. Fuchs, P. Schneider, and G. Schneider, "Generative recurrent networks for de novo drug design," *Molecular informatics*, vol. 37, no. 1-2, p. 1700111, 2018.

[23] B. Sattarov, I. I. Baskin, D. Horvath, G. Marcou, E. J. Bjerrum, and A. Varnek, "De novo molecular design by combining deep

autoencoder recurrent neural networks with generative topographic mapping," *Journal of chemical information and modeling*, vol. 59, no. 3, pp. 1182–1196, 2019.

[24] E. J. Bjerrum and B. Sattarov, "Improving chemical autoencoder latent space and molecular de novo generation diversity with heteroencoders," *Biomolecules*, vol. 8, no. 4, p. 131, 2018.

[25] T. Vallianatou, C. Giaginis, and A. Tsantili-Kakoulidou, "The impact of physicochemical and molecular properties in drug design: navigation in the "drug-like" chemical space," in *GeNeDis 2014*. Springer, 2015, pp. 187–194.

[26] S. Wang, Y. Guo, Y. Wang, H. Sun, and J. Huang, "Smiles-bert: large scale unsupervised pre-training for molecular property prediction," in *Proceedings of the 10th ACM International Conference on Bioinformatics, Computational Biology and Health Informatics*, 2019, pp. 429–436.

[27] K. Y. Gao, A. Fokoue, H. Luo, A. Iyengar, S. Dey, and P. Zhang, "Interpretable drug target prediction using deep neural representation." in *IJCAI*, vol. 2018, 2018, pp. 3371–3377.

[28] B. Shin, S. Park, K. Kang, and J. C. Ho, "Self-attention based molecule representation for predicting drug-target interaction," *arXiv preprint arXiv:1908.06760*, 2019.

[29] A. Gonczarek, J. M. Tomczak, S. Zareba, J. Kaczmar, P. Dabrowski, and M. J. Walczak, "Interaction prediction in structure-based virtual screening using deep learning," *Computers in biology and medicine*, vol. 100, pp. 253–258, 2018.

[30] J. Ma, R. P. Sheridan, A. Liaw, G. E. Dahl, and V. Svetnik, "Deep neural nets as a method for quantitative structure–activity relationships," *Journal of chemical information and modeling*, vol. 55, no. 2, pp. 263–274, 2015.

[31] K. Tian, M. Shao, Y. Wang, J. Guan, and S. Zhou, "Boosting compound-protein interaction prediction by deep learning," *Methods*, vol. 110, pp. 64–72, 2016.

[32] R. J. Williams and D. Zipser, "A learning algorithm for continually running fully recurrent neural networks," *Neural computation*, vol. 1, no. 2, pp. 270–280, 1989.

[33] I. Sutskever, O. Vinyals, and Q. V. Le, "Sequence to sequence learning with neural networks," in *Advances in neural information processing systems*, 2014, pp. 3104–3112.

[34] T. Fink, H. Bruggesser, and J.-L. Reymond, "Virtual exploration of the small-molecule chemical universe below 160 daltons," *Angewandte Chemie International Edition*, vol. 44, no. 10, pp. 1504–1508, 2005.

[35] T. Fink and J.-L. Reymond, "Virtual exploration of the chemical universe up to 11 atoms of c, n, o, f: assembly of 26.4 million structures (110.9 million stereoisomers) and analysis for new ring systems, stereochemistry, physicochemical properties, compound classes, and drug discovery," *Journal of chemical information and modeling*, vol. 47, no. 2, pp. 342–353, 2007.

[36] J.-L. Reymond, L. C. Blum, and R. van Deursen, "Exploring the chemical space of known and unknown organic small molecules at www. gdb. unibe. ch," *CHIMIA International Journal for Chemistry*, vol. 65, no. 11, pp. 863–867, 2011.

[37] G. Landrum, "Rdkit: A software suite for cheminformatics, computational chemistry, and predictive modeling," 2013.

[38] M. Abadi, P. Barham, J. Chen, Z. Chen, A. Davis, J. Dean, M. Devin, S. Ghemawat, G. Irving, M. Isard *et al.*, "Tensorflow: A system for large-scale machine learning," in *12th {USENIX} symposium on operating systems design and implementation ({OSDI} 16)*, 2016, pp. 265–283.

[39] M. A. Johnson and G. M. Maggiora, *Concepts and applications of molecular similarity*. Wiley, 1990.

Roshan R Kotkondawar is a research scholar in the Department of Information Technology, Dr. Babasaheb Ambedkar Technological University, Lonere-Raigad (MS), India. His research interests include Machine Learning, Deep Learning, bioinformatics, Optimization and Data-structures.

Sanjay R. Sutar is Professor and Head, Department of Information Technology, Dr. Babasaheb Ambedkar Technological University, Lonere-Raigad (MS), India. His research interests include Evolutionary algorithms, Scheduling and Data-structures.

Arvind W. Kiwelekar is Professor, Department of Computer Engineering, Dr. Babasaheb Ambedkar Technological University, Lonere-Raigad (MS), India. His research interests include Software Architecture, Artificial Intelligence and Machine Learning.

Hansaraj S. Wankhede is an Associate Professor, Department of Artificial intelligence, G. H. Raisoni College of Engineering, Nagpur- 440016 (MS), India. His research interests include Software Engineering, Cognitive Science and Machine Learning.