UNIVERSITÀ
DI PARMA

Dipartimento di Ingegneria e Architettura

Corso di Laurea in Ingegneria Informatica - Informatica Industriale

---

Machine Learning final project: Companies Status Reconstruction & Classification

Vincenzo Fraello (339641)

ANNO ACCADEMICO 2022/2023

1 Task description

Predicting corporate bankruptcy is one of the fundamental tasks in credit risk assessment [...]. The recent advancements in machine learning (ML) enabled the development of several models for bankruptcy prediction. The most challenging aspect of this task is dealing with the class imbalance due to the rarity of bankruptcy events in the real economy. Furthermore, a fair comparison in the literature is difficult to make because bankruptcy datasets are not publicly available and because studies often restrict their datasets to specific economic sectors and markets and/or time periods [1].

The main objective of this research is to compare the performance of two different architectures employed in predicting corporate bankruptcy, by combining a classification task with an anomaly detection task (specifically: reconstruction task).

2 The Dataset

The entire dataset used for the experiments has been made available to the scientific community for further research and benchmarking purposes. The dataset pertains to 8262 different public companies listed on the American stock market between 1999 and 2018.

For all the companies and for each year, we selected 18 accounting and financial variables. Features were selected according to the most frequently used ratios, and accounting information to which the literature refers [...] [1]. The dataset consists of 26,425 samples grouped in sets of 5, resulting in a total of 5,285 samples. Each sample is composed of the last 5 years of financial statements, starting from the most recent available year. For example, if the latest available year is 2018, the sample would include data from 2014 to 2018. If the latest available year is 2017, the sample would include data from 2013 to 2017. Then the entire dataset is divided into three subsets according to the period of time:

- Training-set: data from 1999 until 2011 for training (2046 samples).
- Validation-set: data from 2012 until 2014 for validation, and model comparison (504 samples).
- Test-set: data from 2015 to 2018 to prove the ability of the models to predict bankruptcy in real never seen cases and macroeconomic conditions (2735 samples).

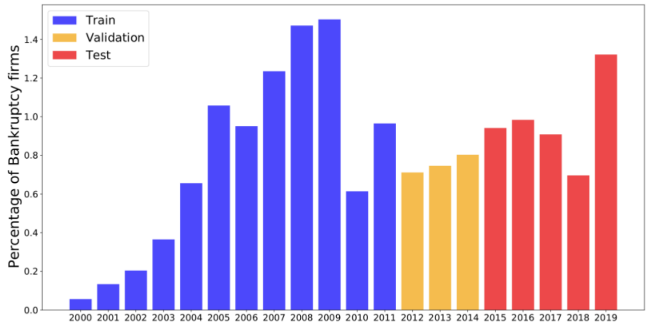


Figure 1. Rate of bankruptcy in the dataset (2000–2019) with financial variables in the period (1999–2018). The next subdivision in training, validation, and test is highlighted with different colors.

Figure 1: Image from [1]

Table 1. The table provides the firm distribution by year in the dataset.

Year	Total Firms	Bankrupt Firms	Year	Total Firms	Bankrupt Firms
2000	5308	3	2010	3743	23
2001	5226	7	2011	3625	35
2002	4897	10	2012	3513	25
2003	4651	17	2013	3485	26
2004	4417	29	2014	3484	28
2005	4348	46	2015	3504	33
2006	4205	40	2016	3354	33
2007	4128	51	2017	3191	29
2008	4009	59	2018	3014	21
2009	3857	58	2019	2723	36

Figure 2: Image from [1]

Table 2. The 18 numerical bankruptcy features we considered in our tests.

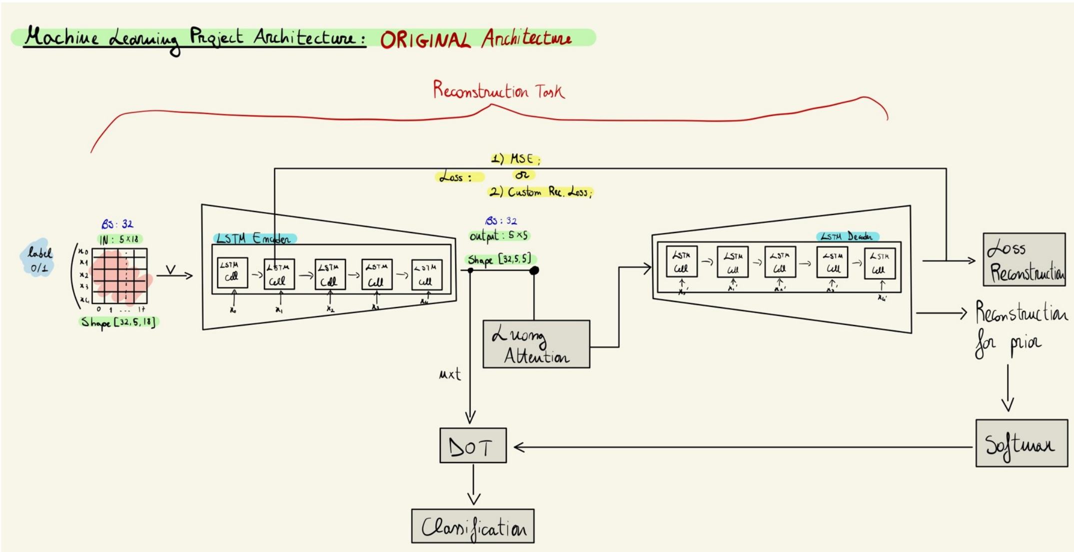
	Variable Name	Description
X1	Current assets	All the assets of a company that are expected to be sold or used as a result of standard business operations over the next year
X2	Cost of goods sold	The total amount a company paid as a cost directly related to the sale of products
X3	Depreciation and amortization	Depreciation refers to the loss of value of a tangible fixed asset over time (such as property. machinery, buildings, and plant). Amortization refers to the loss of value of intangible assets over time.
X4	EBITDA	Earnings before interest,taxes, depreciation and amortization: Measure of a company's overall financial performance alternative to the net income
X5	Inventory	The accounting of items and raw materials that a company either uses in production or sells.
X6	Net Income	The overall profitability of a company after all expenses and costs have been deducted from total revenue.
X7	Total Receivables	The balance of money due to a firm for goods or services delivered or used but not yet paid for by customers.
X8	Market value	The price of an asset in a marketplace. In our dataset it refers to the market capitalization since companies are publicly traded in the stock market
X9	Net sales	The sum of a company's gross sales minus its returns, allowances, and discounts.
X10	Total assets	All the assets, or items of value, a business owns
X11	Total Long term debt	A company's loans and other liabilities that will not become due within one year of the balance sheet date
X12	EBIT	Earnings before interest and taxes
X13	Gross Profit	The profit a business makes after subtracting all the costs that are related to manufacturing and selling its products or services
X14	Total Current Liabilities	It is the sum of accounts payable, accrued liabilities and taxes such as Bonds payable at the end of the year, salaries and commissions remaining
X15	Retained Earnings	The amount of profit a company has left over after paying all its direct costs, indirect costs, income taxes and its dividends to shareholders
X16	Total Revenue	The amount of income that a business has made from all sales before subtracting expenses. It may include interest and dividends from investments
X17	Total Liabilities	The combined debts and obligations that the company owes to outside parties
X18	Total Operating Expenses	The expense a business incurs through its normal business operations

Figure 3: Image from [1]

3 Data manipulation operations

The dataset is imbalanced, but it is not balanced to preserve the anomaly detection task. Otherwise, if an event is not rare, it no longer represents an anomaly. Moreover, the objective is to learn from an imbalanced context. Finally, the provided dataset contains data already normalized for years.

4 Original Networks Architecture



5 Comparison Netork Architectures (1)

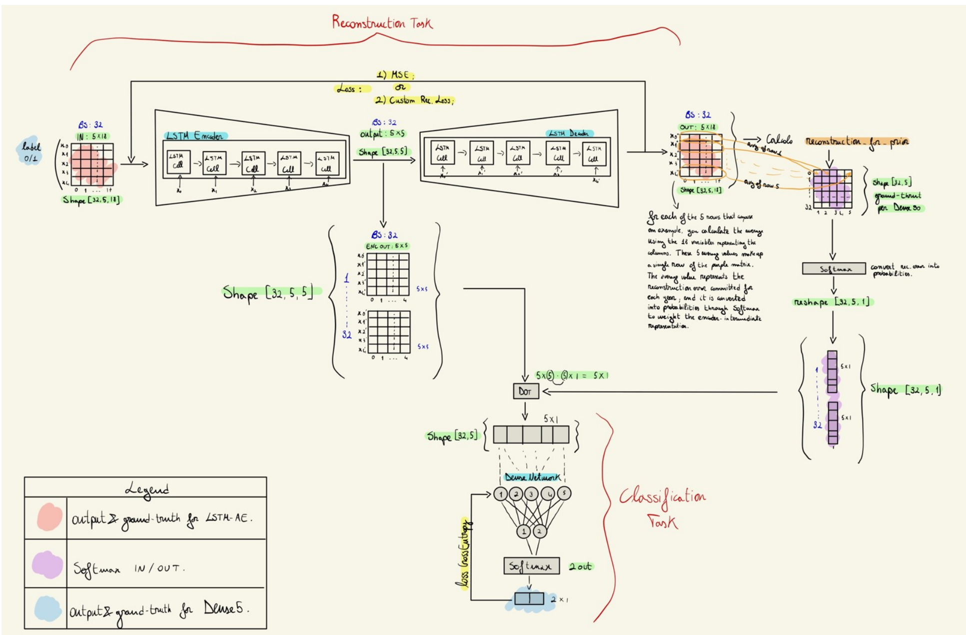


Figure 4: The input of the fc-dense5-net is obtained combining softmax output with encoder output.

6 Comparison Netork Architectures (2)

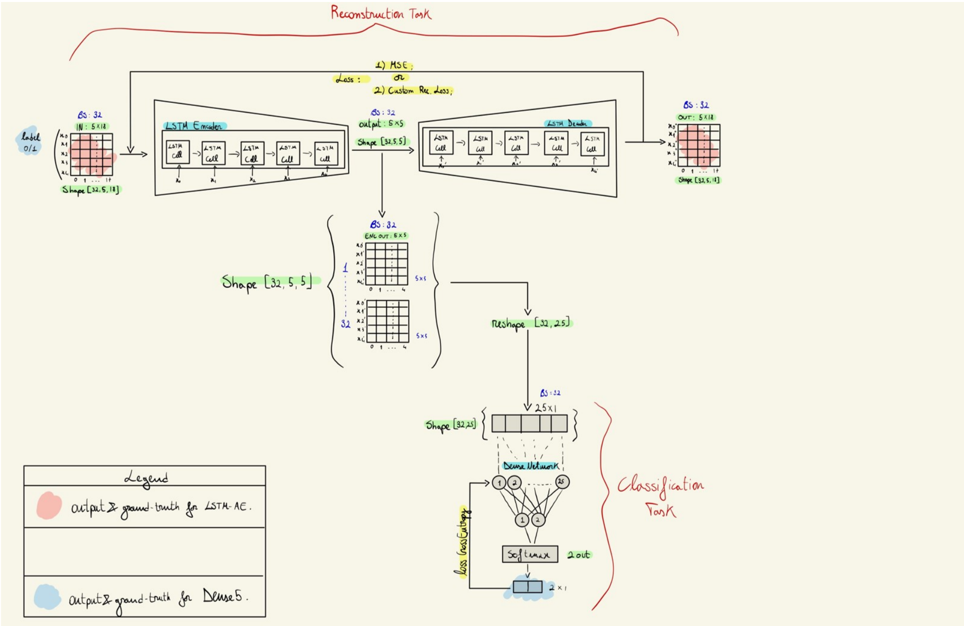


Figure 5: The input of the fc-dense5-net is obtained by flattening the encoder output.

7 Technologies used

The main technologies used for these projects are:

- *Python* programming language;
- *PyTorch* library for deep learning;
- *TensorBoard* for visualization and tracking;
- *Matplotlib* for data visualization;
- *NumPy* for scientific computing;
- *Pandas* for data manipulation and analysis;
- Training on the university cluster.

8 Developed modules

- **main.py:** this module is responsible for launching the simulation. It utilizes other modules, including: (i) Modules for handling training/validation data (dataloader generation). (ii) Modules for training and validation;
- **custom_dataset.py:** this module is responsible for getting data from csv files and generating dataloaders for training, validation and testing;
- **metrics.py:** this module contains various metrics and loss functions used during training, validation and testing;
- **models.py:** this module contains an implementation of the architecture of the networks (LSTM-Autoencoder and Fully-Connected-Network);
- **plotting_utils.py:** this module contains several useful functions for plotting metrics, loss functions, confusion matrix and more;

- **pytorchtools.py:** this module contains an early stopping mechanism implementation;
- **reproducibilit.py:** this module contains some function used to limit the number of sources of nondeterministic behavior in PyTorch;
- **solver.py:** this module includes various methods for training, validation and test models. It also provides functionality for saving and loading models.

9 Operating modes

9.1 Best hyperparameters selection

Initially, I trained the LSTM-Autoencoder by varying some parameters to determine the optimal ones among: number of training epochs, loss function (MSE vs. Custom-Reconstruction), Optimizer (Adam vs. SGD), Early stopping patience and Weights initialization.

At the end, the chosen parameters are: number of training epochs: 3000; Loss function: Custom-Reconstruction; Optimizer: Adam; Early stopping patience: 30; Batch-size: 32; and Weights initialization: True.

9.2 Reconstruction Task: LSTM-AE training and validation

In total, 30 training runs were conducted on the LSTM-Autoencoder using a dataset comprising only of alive companies. The purpose was to observe the network's ability to accurately reconstruct alive companies and identify higher error rates when attempting to reconstruct failed companies. This approach enables determining the state of a company based on the committed error. The training phase yielded reconstruction errors ranging from 0.16 to 0.18, while the validation phase showed errors in the range of 0.25 to 0.28.

The selected model for the classification phase was determined based on its performance during validation. Specifically, the model that exhibited a significant difference between the reconstruction errors of alive companies and failed companies during the validation phase was chosen.

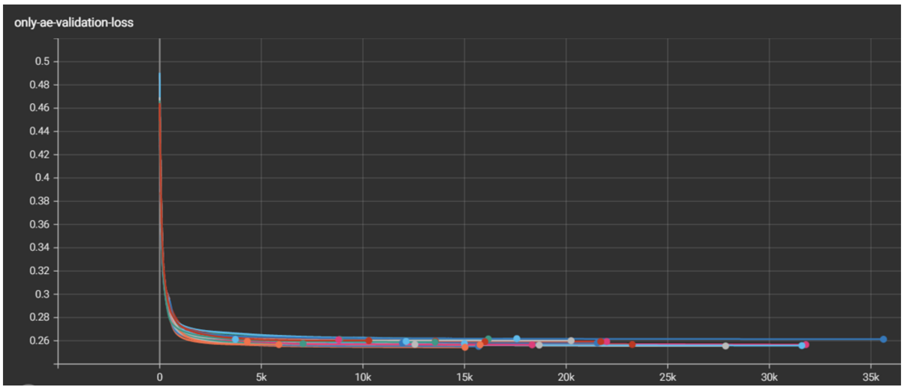


Figure 6: Training loss of the 30 runs of the Autoencoder trained using only alive companies.

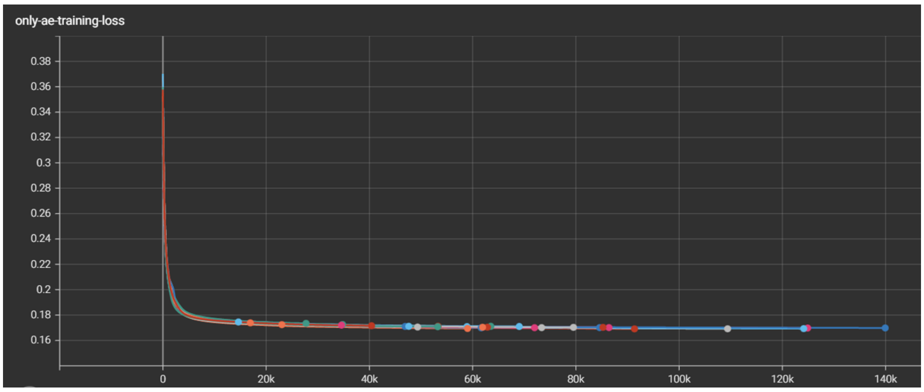


Figure 7: Validation loss of the 30 runs of the Autoencoder trained using only alive companies.

```
"rs15": {
    "validation-loss": 0.2619,
    "mean-reconstruction-error-failed": 0.17382764567931494
},              difference: 0.0881
```

Figure 8: Best model with higher error in validation with alive/failed companies.

9.3 Classification Task: LSTM-AE + FC-Net training and validation

Due to the stochastic nature introduced by ML/DL models, several simulations were performed with varying random seeds to evaluate the performance of different architectures on the data. These random seeds are used to initialize libraries such as torch, numpy, random, and others.

The two network architectures were trained, validated, and tested with the following configurations:

Name	Network Architecture	LSTM-AE freezed	Random seed
config-1	LSTM-AE + FC-Dense-Net + Softmax-Reconstruction	no	from 0 to 2
config-2	LSTM-AE + FC-Dense-Net + Softmax-Reconstruction	yes	from 0 to 2
config-3	LSTM-AE + FC-Dense-Net	no	from 0 to 2
config-4	LSTM-AE + FC-Dense-Net	yes	from 0 to 2

9.3.1 Config-1 results

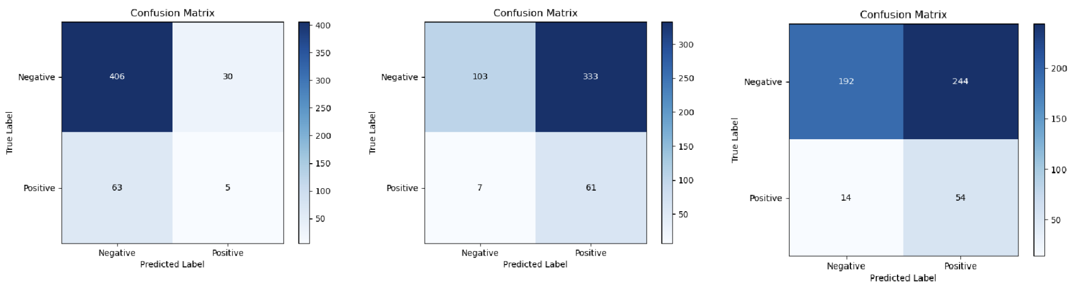


Figure 9: Confusion matrix in validation-set by varying random seed from 0 (left) to 2 (right).

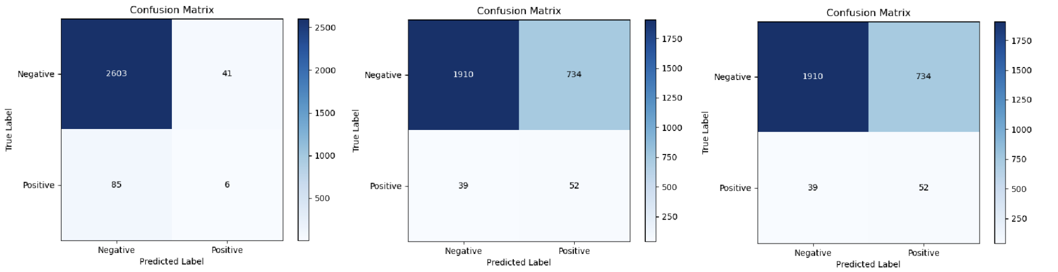


Figure 10: Confusion matrix in test-set by varying random seed from 0 (left) to 2 (right).

9.3.2 Config-2 results

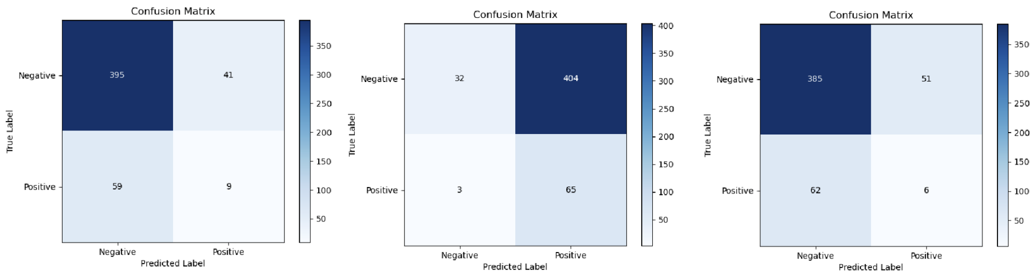


Figure 11: Confusion matrix in validation-set by varying random seed from 0 (left) to 2 (right).

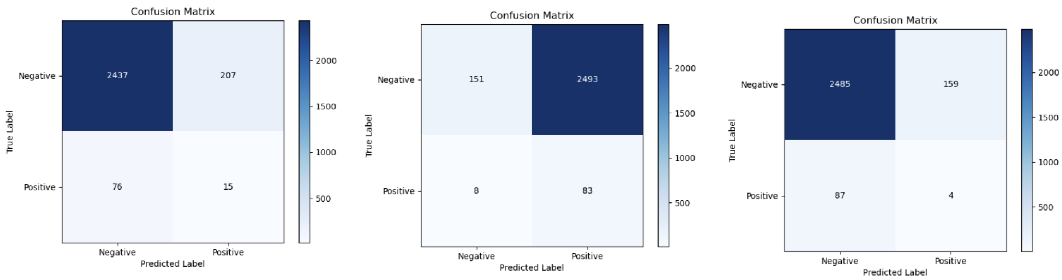


Figure 12: Confusion matrix in test-set by varying random seed from 0 (left) to 2 (right).

9.3.3 Config-3 results

Figure 13: Confusion matrix in validation-set by varying random seed from 0 (left) to 2 (right).

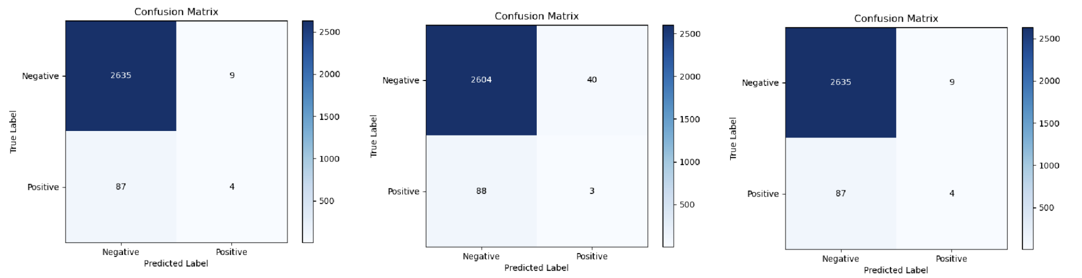


Figure 14: Confusion matrix in test-set by varying random seed from 0 (left) to 2 (right).

9.3.4 Config-4 results

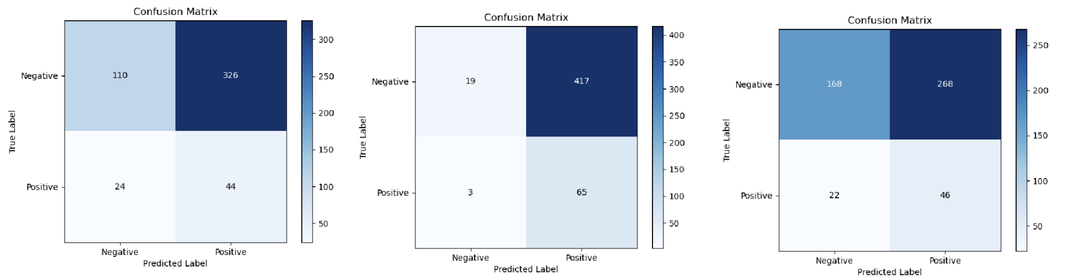


Figure 15: Confusion matrix in validation-set by varying random seed from 0 (left) to 2 (right).

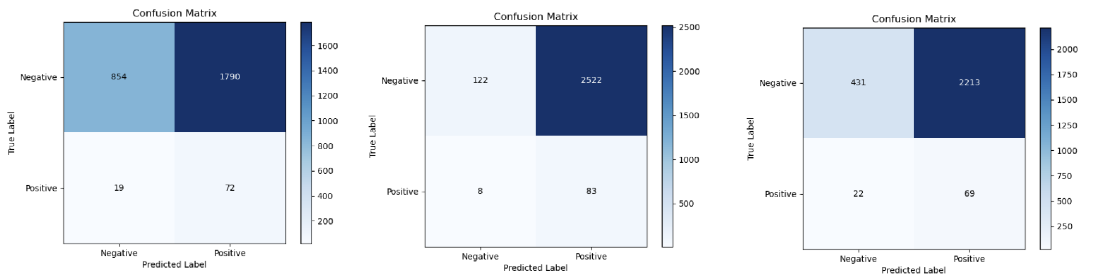


Figure 16: Confusion matrix in test-set by varying random seed from 0 (left) to 2 (right).

9.4 Precision-Recall on positive class

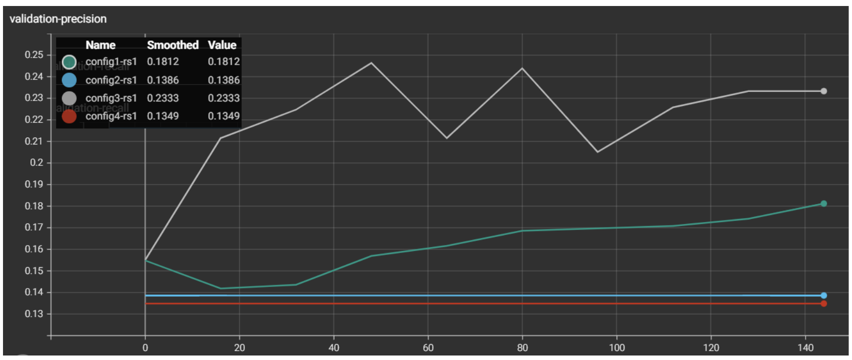


Figure 17: Precision on validation-set with random-seed 1 (increase in false positive rate in configurations where LSTM-AE is trained).

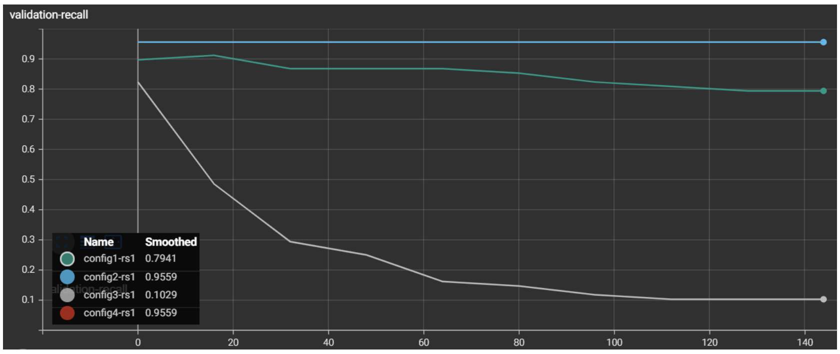


Figure 18: Recall on validation-set with random-seed 1.

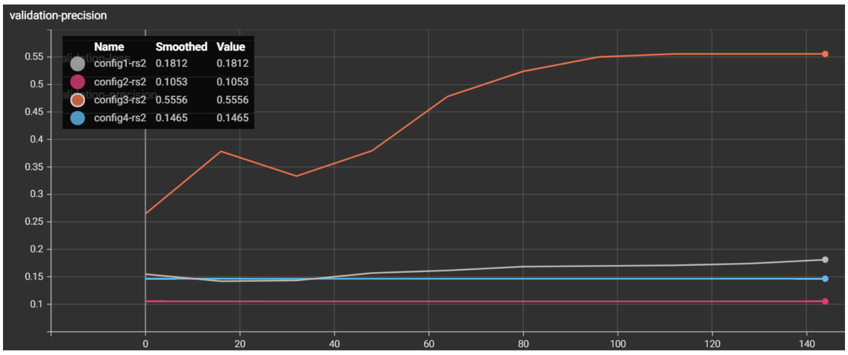


Figure 19: Precision on validation-set with random-seed 2.

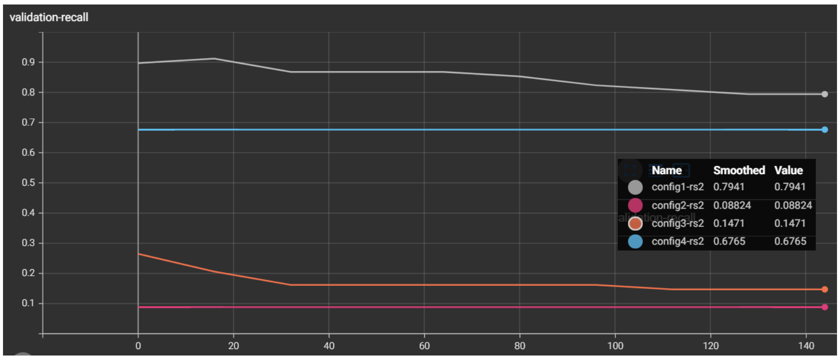


Figure 20: Recall on validation-set with random-seed 2.

10 Conclusions

After conducting various experiments, valuable results were obtained to compare the architectures presented in the preceding sections. The confusion matrices revealed that, due to the stochastic nature of the utilized tools, different outcomes can be observed even when all parameters remain unchanged, solely by varying the random seed. The performance of the configuration using Attention (Reconstruction-Softmax) sometimes surpasses the one using the LSTM-AE encoder's learned embedding, depending on the random seed.

Additionally, it was observed that training the LSTM-AE network during the classification task leads to an increase in false positives. The network learns to classify alive companies as failed, resulting in a higher number of misclassifications. Conversely, when the network remains frozen, the knowledge acquired in reconstructing alive companies remains intact. However, this frozen network struggles to learn how to accurately classify failed companies, resulting in a lower true positive rate.

In summary, the experiments demonstrated the impact of stochasticity in the results and highlighted the trade-off between false positives and true positives when training or freezing the LSTM-AE network during the classification task.

References

[1] Gianfranco Lombardo; Mattia Pellegrino; George Adosoglou; Stefano Cagnoni; Panos M Pardalos; Agostino Poggi. *Machine Learning for Bankruptcy Prediction in the American Stock Market: Dataset and Benchmarks*. 2022. URL: https://scholar.google.com/citations?view_op=view_citation&hl=it&user=XaGrJdQAAAAJ&sortby=pubdate&citation_for_view=XaGrJdQAAAAJ:qxL8FJ1GzNcC.