terazi: AI Fairness Tool for Doubly Imbalanced Data

Asli Umay Ozturk
Middle East Technical University (METU)
Ankara, Türkiye
auozturk@ceng.metu.edu.tr

Yigit Sever
Middle East Technical University (METU)
Ankara, Türkiye
yigit@ceng.metu.edu.tr

Ata Yalcin
Middle East Technical University (METU)
Ankara, Türkiye
ata.yalcin@metu.edu.tr

Viktoria Pauw
Leibniz Supercomputing Centre (LRZ)
Garching b.M., Germany
viktoria.pauw@lrz.de

Stephan Hachinger
Leibniz Supercomputing Centre (LRZ)
Garching b.M., Germany
stephan.hachinger@lrz.de

Ismail Hakki Toroslu
Middle East Technical University (METU)
Ankara, Türkiye
toroslu@ceng.metu.edu.tr

Pinar Karagoz
Middle East Technical University (METU)
Ankara, Türkiye
karagoz@ceng.metu.edu.tr

Abstract

The field of Artificial Intelligence (AI) fairness focuses on developing unbiased approaches for machine learning problems, with many contributions and ready-to-use tools. However, existing solutions fall short when both sensitive attributes and target labels have imbalanced representations in a given dataset. Our proposed algorithm and its tool, `terazi`, aim to propose a fair AI solution for this doubly imbalanced case. The proposed solution is based on finding the optimal distribution within the imbalanced data to balance fairness and classification performance, and the tool facilitates using this solution. In this demonstration, we showcase the capabilities of our algorithm, and the easy-to-use GUI of our web application for data scientists, researchers, and AI practitioners.

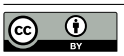
CCS Concepts

• **Software and its engineering → Software libraries and repositories**.

Keywords

Algorithmic fairness, Imbalanced data, Fair AI, Fraud detection, Multi-parameter optimization

ACM Reference Format:
Asli Umay Ozturk, Yigit Sever, Ata Yalcin, Viktoria Pauw, Stephan Hachinger, Ismail Hakki Toroslu, and Pinar Karagoz. 2025. terazi: AI Fairness Tool for Doubly Imbalanced Data. In *Proceedings of the 34th ACM International Conference on Information and Knowledge Management (CIKM '25), November 10–14, 2025, Seoul, Republic of Korea.* ACM, New York, NY, USA, 5 pages. https://doi.org/10.1145/3746252.3761465



CIKM '25, Seoul, Republic of Korea

ACM ISBN 979-8-4007-2040-6/2025/11
https://doi.org/10.1145/3746252.3761465

1 Introduction

Fairness in Artificial Intelligence (AI) and Machine Learning (ML) strives to create models and datasets that do not discriminate against any particular group. As AI tools are increasingly used in classification tasks, unfair models can make biased decisions against unprivileged groups in the data. In the literature, there is a variety of fairness approaches that work well to debias datasets or create fair ML models [2, 6, 7, 9, 12]. There also exist ready-to-use tools for AI practitioners that aim to make proposed methods in the literature easily accessible [1, 3, 13]. However, these approaches were demonstrated on balanced datasets where the classification label is split fairly evenly and the unprivileged population has significant representation in the dataset. As the authors previously claimed [15], existing approaches in the literature, when applied to imbalanced data, tend to ignore the unfavourable target label, causing poor classification performance while trying to increase the fairness of the models. Even though there exist studies on the intersection of fairness and imbalanced data [10, 11, 14], most proposed approaches are either model-dependent or task-specific. Motivated by these shortcomings, we present `terazi`.

`terazi` is a web application that aims to make the recently proposed doubly-balanced fairness approach [15] easily accessible to AI practitioners. Figure 1 summarizes the algorithm behind `terazi`, which searches for the best distribution of imbalanced attributes and labels within the dataset through different samplings to be able to balance classification performance and fairness. In this paper, we demonstrate the web application `terazi` which offers an interactive GUI for AI practitioners so that they will be able to use the proposed fairness approach with their own datasets.

Theoretical concepts, sampling, and training process used by `terazi` are summarized in Section 2. The system description of the web application with interactive GUI is explained in Section 3. Last but not least, detailed descriptions of features and demonstration scenarios are explained in Section 4.[1]

[1] Demonstration video: https://youtu.be/Es7Ru3BsvaQ

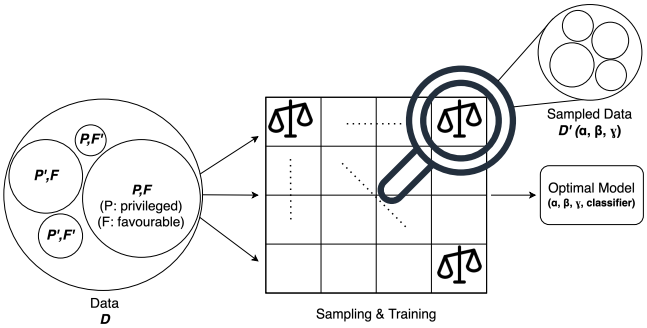


Figure 1: Overview of terazi's algorithm

2 The terazi Framework

From a high-level view, terazi searches for the optimal model by generating samples of doubly imbalanced dataset. The samples generated according to the terazi Sampling Algorithm (Subsection 2.1) are used for training classifiers. The performance of the classification models is evaluated and compared using the MCC-DI Loss function (Subsection 2.2) in order to find the best representation of the imbalanced features to reduce the loss.

2.1 Sampling Algorithm

The goal of the sampling algorithm is to generate a subset of the given dataset D, where the ratios of the four partitions p_f, p_uf, up_f and up_f are controlled by the parameters α, β and γ. The ratios of these partitions denote the distributions of the imbalanced sensitive attribute and the label in the sample. The resulting sample is denoted as D', and the partitions in the sampled set are denoted as p_f', p_uf', up_f' and up_f'. These partitions are defined as follows:

- p_f : privileged instances with favourable label
- p_uf : privileged instances with unfavourable label
- up_f : unprivileged instances with favourable label
- up_uf : unprivileged instances with unfavourable label

When sampling the dataset and generating a new setup with different ratios for partitions, we limit our sample space with some principles. If a partition is overrepresented (i.e., if its ratio is high) in the original dataset, it cannot be more frequent in the sampled dataset. Similarly, if a partition is underrepresented in the original dataset, it cannot be less frequent in the sampled dataset. In other words, partitions should not interchange roles with their counterparts in terms of bias. Additionally, the ideal setup in terms of the balance of the dataset is considered as the equal rates of the four partitions. With these principles, three parameters are defined as follows, where values between 0 and 1 are computed from the linear interpolation of the two endpoints:

- Parameter α: It is the parameter to control the unprivileged group rate within D', where 0 sets the rate to the ratio in D, 1 sets the ratio to be half of the total instances.
- Parameter β: It is the parameter to control the unfavourable labelled instance rate within D', where 0 sets the rate as the same rate of unfavourable labels as in D, and 1 sets it to be half.
- Parameter γ: It is the parameter to control the ratio of the privileged group compared to the unprivileged group in getting assigned the favourable label. With value 0, D' has the same ratio as in D, and with value 1, the ratio of aforementioned groups for getting assigned the favourable label is equal to 1.

Formulating the described principles, definitions of the three parameters, the size of the four demographics p_f', p_uf', up_f' and up_uf' can be determined unambiguously with the equations shown in Equation 1.

$$
\begin{aligned}
&1.\ \frac{|p'|}{|D'|} = \frac{|p|}{|D|}(1-\alpha) + 0.5 * \alpha \\
&2.\ \frac{|f'|}{|D'|} = \frac{|f|}{|D|}(1-\beta) + 0.5 * \beta \\
&3.\ \frac{\frac{|p\_f'|}{|p'|}}{\frac{|up\_f'|}{|up'|}} = \frac{\frac{|p\_f|}{|p|}}{\frac{|up\_f|}{|up|}} * (1-\gamma) + \gamma
\end{aligned}
\tag{1}
$$

Sampling process, with detailed explanations of the equations and walkthroughs, are available in full work [15].

2.2 MCC-DI Ratio Loss

terazi uses each sampled set D' to train a classifier. Currently, the classifiers are Logistic Regression (LR), Naive Bayes (NB), Random Forest (RF) and Support Vector Machine (SVM). The optimal model is selected according to the loss function described in Equation 2.

$$
Loss = |(1 - DIRatio)| + |(1 - MCC)| \tag{2}
$$

In Equation 2, DI Ratio stands for the *Disparate impact (DI) ratio*, which is a fairness metric derived from the concept of *Disparate Impact (DI)*. It is defined as the systematic favouritism done to a certain group[2], and calculated as shown in Equation 3. In the equation, G denotes group, L denotes label, and $P(X|Y)$ stands for the probability that X is true given Y is true. Since a fair model is expected to behave similarly for both privilege groups, the optimal value for DI ratio is 1, and values between $0.8 - 1.2$ are considered acceptable [5].

$$
\frac{P(\text{L=unfavourable} \mid \text{G=unprivileged})}{P(\text{L=unfavourable} \mid \text{G=privileged})} \tag{3}
$$

On the other hand, Matthews Correlation Coefficient (MCC) in Equation 4, is a classification score. It is a special case of the Phi coefficient in statistics [4]. For cases when a classification task has imbalanced labels, MCC is reported to be a better alternative due to its formula distributing the focus to all types of true and misclassifications. [4, 16]. MCC can have values in the range $[-1, 1]$, where 1 is the best possible value and -1 is the worst possible value. It is calculated as given in Equation 4.

$$
\frac{TP * TN - FP * FN}{\sqrt{(TP + FP) * (TP + FN) * (TN + FP) * (TN + FN)}} \tag{4}
$$

[2] https://www.britannica.com/topic/disparate-impact

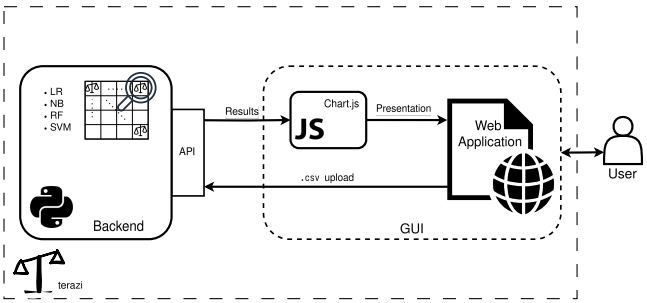


Figure 2: Architecture diagram of `terazi`

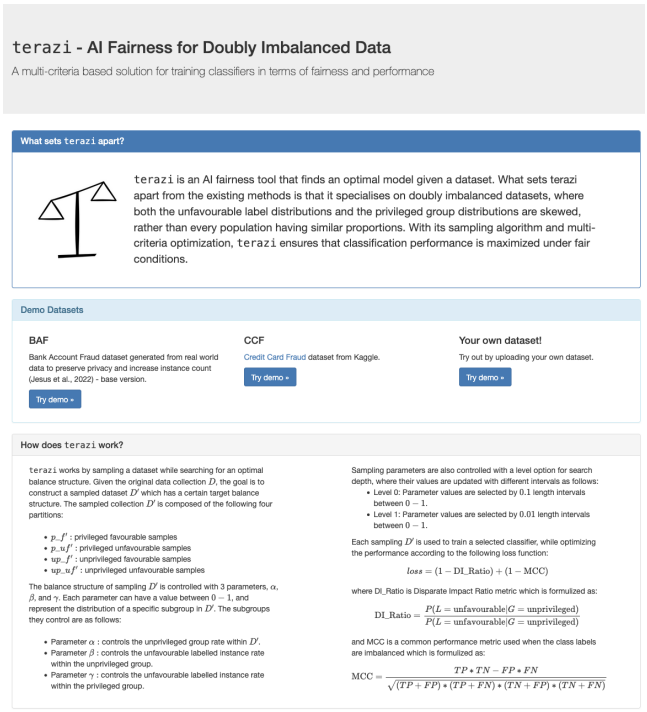


Figure 3: Landing page of `terazi`

3 System Description

Architecture. `terazi` is implemented using JavaScript and Python programming languages. It is designed to be a responsive web application that can work on any modern web browser. Python Flask is used to run the web application and connect the user views with the system backend. The user uploads data as a file, with no additional database needed to run the system. After running the algorithm, generated results are stored as `.pickle` files to be processed by the GUI API to generate related graphs for the user view. GUI, implemented using Javascript, uses Chart.js library to render the demographic graphs, as well as the loss graph for the selected model. The architecture diagram of `terazi` is shown in Figure 2.

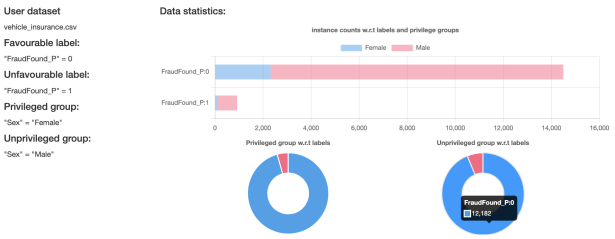


Figure 4: Information screen for the uploaded dataset

Datasets. We showcase `terazi` on three datasets. The first two datasets are embedded in the system for users to explore the capabilities of `terazi`. The third dataset is used to present how users can submit their own dataset to run the algorithm. To demonstrate the effectiveness of `terazi` on imbalanced datasets, all three datasets are chosen to have an imbalance on both the privilege attribute and the classification label. Hence, the following fraud detection datasets, which fit these considerations by their nature, are selected:

- Exploration Data 1 - Bank Account Fraud (BAF) [8]: Released in 2022, BAF is a crafted dataset with 6 variants, each having an induced bias. This dataset is synthesized from real-world data, with the original withheld for privacy reasons. We use the *base* variant, where the unfavourable label is being classified as fraud, and the sensitive attribute is the age of the account holder.
- Exploration Data 2 - Credit Card Fraud (CCF) [3]: Collected by the International Institute of Information Technology Bangalore, this dataset includes credit card applications. The unfavourable label for this dataset is whether the credit card application is fraudulent or not, and the sensitive attribute is Sex, with Female as the privileged label.
- User Supplied Data - Vehicle Insurance Fraud (VIF) [4]: Released by Oracle in 2020, this dataset includes 15420 data points including vehicle accidents and insurance claims. The unfavourable label for the dataset is whether the insurance claim application is found to be fraudulent or not, and the sensitive attribute is Sex, with Female as the privileged label.

4 Features and Demonstration Scenarios

We showcase the system with a demonstration[5]. The captured demonstration was run in Firefox 136.0.3 (aarch64) on an Apple MacBook Pro with an Apple M1 Pro chip and 16 GBs of RAM. There are two sections in the showcase; case studies on existing datasets and the module for user-supplied dataset exploration.

4.1 Features

`terazi` offers two features: data information and model exploration.

Data Information. Given a dataset, `terazi` first analyses the data and shows the distribution of different classes from the cross-product of favourable-unfavourable and privileged-unprivileged

[3] kaggle.com/datasets/mishra5001/credit-card
[4] kaggle.com/datasets/shivamb/vehicle-claim-fraud-detection
[5] https://youtube.com/watch?v=Es7Ru3BsvaQ

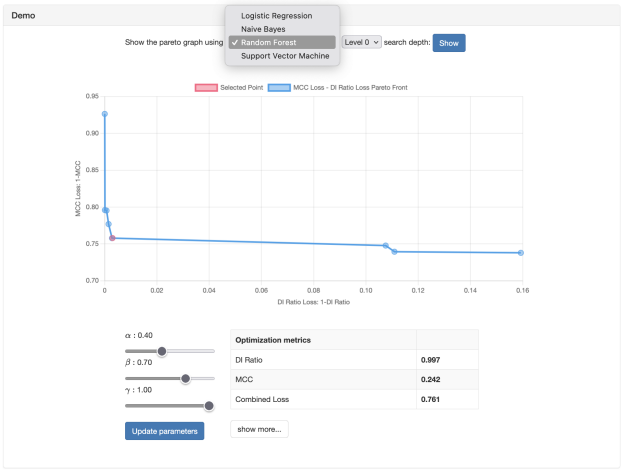


Figure 5: Classifier selection and Pareto graph view for model parameters

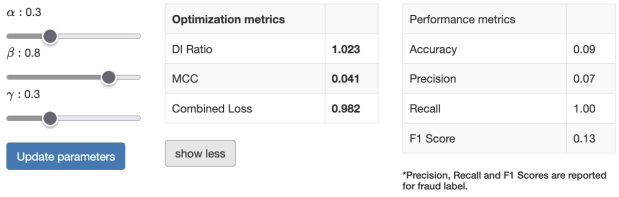


Figure 6: Sliders to choose parameters, table for additional scores

partitions in the dataset. We find that showing the imbalances in the dataset first and foremost before the analysis helps practitioners understand the importance of fair AI approaches. For any dataset that can be explored, three graphs are presented to the user as seen in Figure 4; a graph showing the distribution of privileged and unprivileged instances for both favourable and unfavourable labels, and two graphs for each privilege group showing the distribution of labels for that group. These visualizations make it easier to understand the level of imbalance in a dataset.

Model Exploration. After `terazi`'s algorithm is run, users can explore different sampling setups according to three parameters. Every possible setup of the three parameters is used to sample the given dataset, and four classification models are used for each combination of parameters to train a classifier. In the model exploration view shown in Figure 5, users can select a classifier, and the granularity level for parameter value precision to explore the metrics and performance scores of the trained models under different sampling parameter setups. The loss graph in this view is interactive, where users can see the parameter setups for selected point on hover. Under the loss graph, the parameter setup resulting in the smallest loss value is shown with a table of metrics for the trained model. As seen in Figure 6, sliders for the parameters are responsive, and clicking "update parameters" updates the table and the loss graph

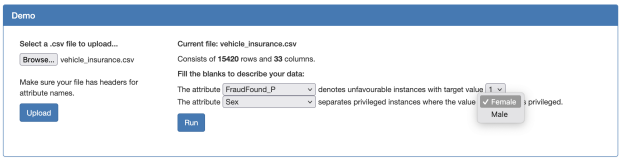


Figure 7: Data upload screen for users

to represent the information of the model trained with the newly selected parameters. Performance metrics of the model can be seen by clicking "show more".

4.2 Demonstration Scenarios

We start the demonstration on the landing page and explore two main scenarios. The landing page (Figure 3) offers users two datasets to explore the system with, as well as the option to upload their own data to explore the system. This page also has a panel for explaining the tool on a surface level, as well as a panel with detailed theoretical information on the algorithm of `terazi`.

Scenario 1: Uploading a user dataset to run `terazi`*'s algorithm and explore the results.* In the first scenario, we demonstrate how easy it is for users to upload their own data to the system. Using the upload view shown in 7, the VIF dataset is uploaded, relevant fields are chosen to define sensitive attributes and target labels, and the algorithm is run. When the results are ready, users are directed to the data information and model exploration view shown in Figures 4, 5. Here, different classifiers, levels and parameter setups are explored in an interactive manner to understand the performance of different models and select the setup according to the user's needs.

Scenario 2: Exploring two different ready-to-explore fraud detection datasets. In this scenario, users can explore the BAF and CCF datasets, where the results of the sampling and training of `terazi` have already been performed and are ready to be explored in the system. BAF and CCF datasets are selected to showcase the performance of `terazi` with datasets of different sizes and distributions. Pre-computed nature of this scenario aims to highlight the exploration functionality of `terazi` with all models trained with many parameter setups that can be compared in an interactive manner.

Acknowledgments

This research received the support of the EXA4MIND project, funded by the European Union´s Horizon Europe Research and Innovation Programme, under Grant Agreement N° 101092944. Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union or the European Commission. Neither the European Union nor the granting authority can be held responsible for them. The work is also supported by EuroHPC Development Access Call with Project ID DD-23-122. The work is also partially supported by METU with Grant No. ADEP-312-2024-11484.

GenAI Usage Disclosure

No GenAI assistance was used in the creation of this work. This includes the research and implementation of the concepts presented in this paper, as well as the writing, proofreading and editing of the paper itself.

References

[1] Rachel K. E. Bellamy, Kuntal Dey, Michael Hind, Samuel C. Hoffman, Stephanie Houde, Kalapriya Kannan, Pranay Lohia, Jacquelyn Martino, Sameep Mehta, Aleksandra Mojsilovic, Seema Nagar, Karthikeyan Natesan Ramamurthy, John Richards, Diptikalyan Saha, Prasanna Sattigeri, Moninder Singh, Kush R. Varshney, and Yunfeng Zhang. 2018. AI Fairness 360: An Extensible Toolkit for Detecting, Understanding, and Mitigating Unwanted Algorithmic Bias. arXiv:1810.01943 [cs]

[2] Sumon Biswas and Hridesh Rajan. 2021. Fair Preprocessing: Towards Understanding Compositional Fairness of Data Transformers in Machine Learning Pipeline. In *Proceedings of the 29th ACM Joint Meeting on European Software Engineering Conference and Symposium on the Foundations of Software Engineering*. ACM, Athens Greece, 981–993. doi:10.1145/3468264.3468536

[3] Kathleen Cachel and Elke Rundensteiner. 2024. FairRankTune: A Python Toolkit for Fair Ranking Tasks. In *Proceedings of the 33rd ACM International Conference on Information and Knowledge Management (CIKM '24)*. Association for Computing Machinery, New York, NY, USA, 5195–5199. doi:10.1145/3627673.3679238

[4] Davide Chicco and Giuseppe Jurman. 2020. The advantages of the Matthews correlation coefficient (MCC) over F1 score and accuracy in binary classification evaluation. *BMC genomics* 21 (2020), 1–13.

[5] Michael Feldman, Sorelle Friedler, John Moeller, Carlos Scheidegger, and Suresh Venkatasubramanian. 2015. Certifying and Removing Disparate Impact. arXiv:1412.3756 [cs, stat]

[6] Rubén González-Sendino, Emilio Serrano, and Javier Bajo. 2024. Mitigating Bias in Artificial Intelligence: Fair Data Generation via Causal Models for Transparent and Explainable Decision-Making. *Future Generation Computer Systems* 155 (June 2024), 384–401. doi:10.1016/j.future.2024.02.023

[7] Vasileios Iosifidis and Eirini Ntoutsi. 2019. AdaFair: Cumulative Fairness Adaptive Boosting. In *Proceedings of the 28th ACM International Conference on Information and Knowledge Management*. ACM, Beijing China, 781–790. doi:10.1145/3357384.3357974

[8] Sérgio Jesus, José Pombal, Duarte Alves, André Cruz, Pedro Saleiro, Rita P. Ribeiro, João Gama, and Pedro Bizarro. 2022. Turning the Tables: Biased, Imbalanced, Dynamic Tabular Datasets for ML Evaluation. *Advances in Neural Information Processing Systems* (2022).

[9] Patrik Joslin Kenfack, Adil Mehmood Khan, S.M. Ahsan Kazmi, Rasheed Hussain, Alma Oracevic, and Asad Masood Khattak. 2021. Impact of Model Ensemble On the Fairness of Classifiers in Machine Learning. In *2021 International Conference on Applied Artificial Intelligence (ICAPAI)*. IEEE, Halden, Norway, 1–6. doi:10.1109/ICAPAI49758.2021.9462068

[10] Ana Lavalle, Alejandro Mate, Juan Trujillo, and Miguel A Teruel. 2023. A data analytics methodology to visually analyze the impact of bias and rebalancing. *IEEE Access* 11 (2023), 56691–56702.

[11] Shruti Nagpal, Maneet Singh, Richa Singh, and Mayank Vatsa. 2022. Detox Loss: Fairness Constraints for Learning With Imbalanced Data. *IEEE Transactions on Biometrics, Behavior, and Identity Science* 5, 2 (2022), 244–254.

[12] Brianna Richardson and Juan E. Gilbert. 2021. A Framework for Fairness: A Systematic Review of Existing Fair AI Solutions. arXiv:2112.05700 [cs]

[13] Pedro Saleiro, Benedict Kuester, Loren Hinkson, Jesse London, Abby Stevens, Ari Anisfeld, Kit T. Rodolfa, and Rayid Ghani. 2019. Aequitas: A Bias and Fairness Audit Toolkit. arXiv:1811.05577 [cs]

[14] Lele Sha, Dragan Gašević, and Guanliang Chen. 2023. Lessons from debiasing data for fair and accurate predictive modeling in education. *Expert Systems with Applications* 228 (2023), 120323.

[15] Ata Yalcin, Asli Umay Ozturk, Yigit Sever, Viktoria Pauw, Stephan Hachinger, Ismail Hakki Toroslu, and Pinar Karagoz. 2025. Fair for a Few: Improving Fairness in Doubly Imbalanced Datasets. arXiv:2506.14306 [cs] doi:10.48550/arXiv.2506.14306

[16] Qiuming Zhu. 2020. On the performance of Matthews correlation coefficient (MCC) for imbalanced dataset. *Pattern Recognition Letters* 136 (2020), 71–80.