Deep learning for survival analysis: a review:

Sharmin Akter
6336641

Contents

1. Introduction

2.1 Targets of estimation

2.2 Data-related aspects

2.3 Estimation in Survival Analysis

3. Deep Learning in Survival Analysis

3.1 Study Selection

3.2 Architectural choices

3.3 Handling of Censoring and Truncation

3.4 Supported Feature-Related Aspects

3.5 Post-Hoc Interpretability Methods

3.6 Model Evaluation

4 Conclusion

1. Introduction

- **Survival Analysis (SA)**, or time-to-event analysis, estimates the distribution of outcome variables that are partially censored, truncated, or both. Common applications include time to death, system failure, or remission.

- **Traditional Methods:**
  - **Non-parametric** approaches like the **Kaplan-Meier estimator** (1958) are foundational and still widely used.
  - **Semi-parametric** models, particularly the **Cox proportional hazards** model (1972), have been central historically.
- **Machine Learning in Survival Analysis:**
  - Since the early 2000s, ML methods like **Random Survival Forests** (2008) and **boosting methods** (2008) have been adapted for survival tasks, offering better predictive performance than traditional models.
- **Neural Networks and Deep Learning:**
  - Shallow neural networks were used in survival tasks as early as the 1990s, but modern Deep Learning (DL) models have seen major developments only since the late 2010s.
  - There's currently no comprehensive systematic review of DL-based survival methods, despite a growing number of proposals.

1. Introduction

- **Existing Reviews and Gaps:**
  - Previous works (e.g., Schwarzer et al. 2000, Lee and Lim 2019, Deepa and Gunavathi 2022) focus narrowly on clinical data, genomics, or specific use cases like cancer prediction, lacking a broad overview of DL methods for time-to-event data.

- **Purpose of the Paper:**
  - The paper provides a comprehensive review of DL-based survival methods, considering both theoretical dimensions (model class, NN architecture) and data-related aspects (outcome types, feature types).

2.1 Targets of estimation

2 ways:

1. Censoring:

- Describes situations where the exact time of the event is unknown for some subjects (e.g., if the study ends before the event occurs).

2. Truncation:

- Refers to cases where subjects are only included in the analysis if the event occurs within a certain time window, leading to biased samples if not properly accounted for.

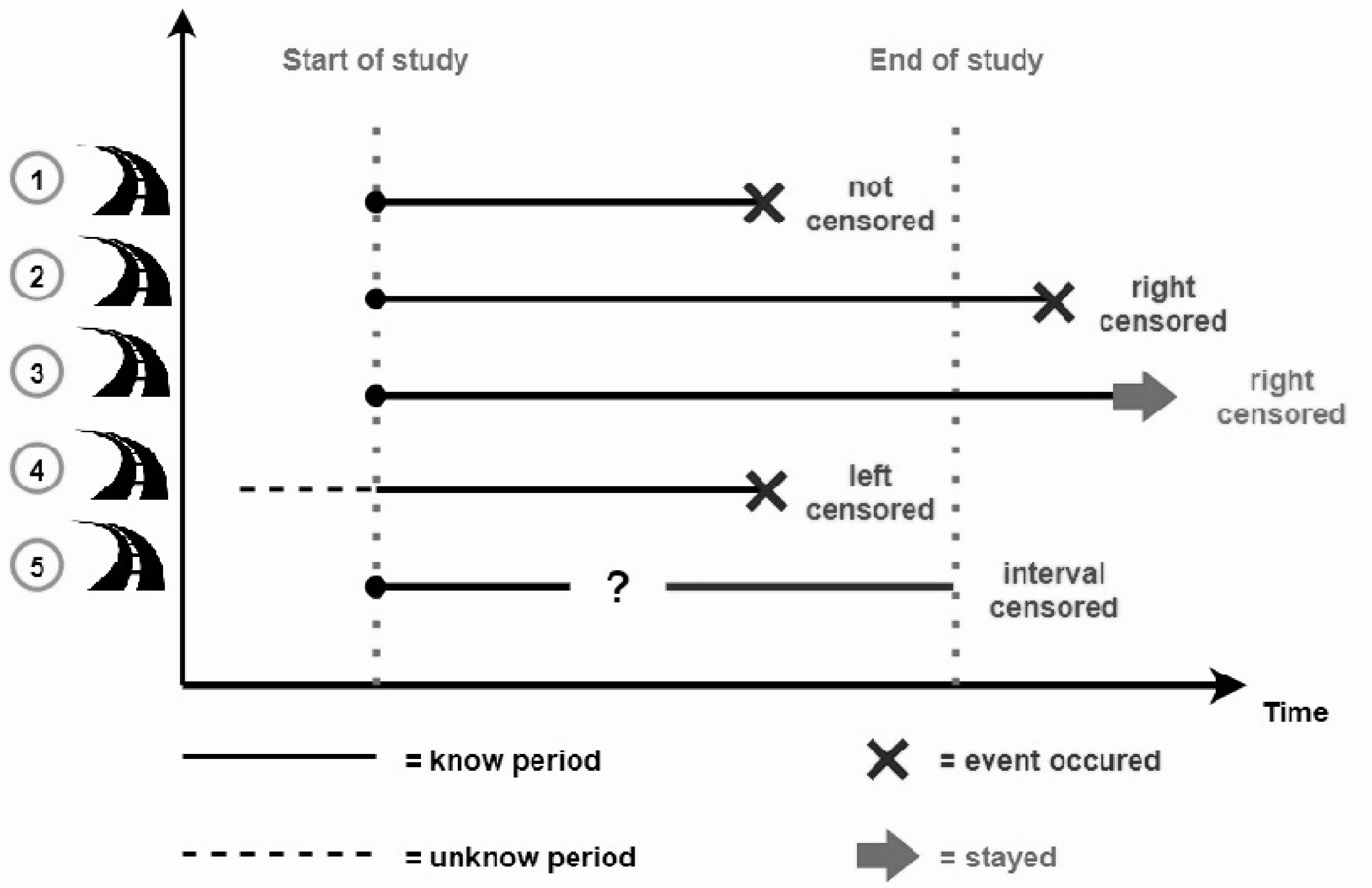

Start of study
End of study
1
not censored
2
right censored
3
right censored
4
left censored
5
?
interval censored
Time
= know period
= unknow period
= event occured
= stayed

Table 2 Overview of different outcome types

Outcome type	Example
Right-censoring	Clinical trials: exact event times are unobserved for some individuals because of dropout
Left-censoring	Age at which children learn a certain task: some children already know the task at the beginning of the study, but it is unknown at which age they learned it
Interval-censoring	Medical study with a periodic follow-up: exact event times are unknown, only the interval between two follow-ups is known
Right-truncation	Transfusion-induced AIDS onset study (Klein and Moeschberger 1997): only patients developing AIDS from transfusion before the registry sampling date are included, while patients with onset after that date are right-truncated
Left-truncation	Coumarin abortion study (Meister and Schaefer 2008): only women conscious of their pregnancy are included; women who had a spontaneous abortion before their pregnancy is recognized never enter the study

Nature of Time Representation

- **Continuous Survival Analysis:**
  - Time is treated as a continuous variable.
  - Events can occur at any moment, making it possible to measure exact time intervals (e.g., survival in days, hours, or seconds).
  - Best suited for scenarios where time can be measured precisely, such as medical survival studies or machine reliability analysis.
- **Discrete Survival Analysis:**
  - Time is divided into intervals or is inherently discrete (e.g., time measured in years, grade levels, or predefined periods).
  - Events are assumed to occur at specific time points (e.g., monthly or yearly intervals
  - Useful when data naturally fits into time intervals or when only interval-based data is available (e.g., annual dropout rates in education).

Hazard Function

- **Continuous SA:**
  - The hazard function hT(t) represents the instantaneous risk of the event occurring at exactly time t, given that the event hasn’t occurred yet.
- **Discrete SA:**
  - The discrete hazard function hT(t) is the probability of the event occurring in a specific interval t, given survival until the start of that interval.

Survival Function

- **Continuous SA:**
  - The survival function $ST(t)=P(T>t)$ is computed as $ST(t)=e^{-HT(t)}$, where $HT(t)$ is the cumulative hazard.
- **Discrete SA:**
  - The survival function is computed as $ST(t)=\prod\prod j=1t(1-hT(j))$, reflecting the product of the probability of surviving each interval.

2.2 Data-related aspects

1. Outcome types:

- **Competing Risks:**
  - Involves mutually exclusive events where only one can occur (e.g., death vs. hospital discharge).
- **Multi-State Models:**
  - Allows for multiple events (transient and terminal) and transitions between states (e.g., different illness stages leading to death).
- **Recurrent Events:**
  - Models situations where the same event can occur multiple times (e.g., epilepsy seizures or sports injuries).

2.2 Data-related aspects

2. Feature-Related Aspects:

- **Time-Varying Features (TVFs) and Time-Varying Effects (TVEs):**
  - TVFs like weight or lifestyle factors change over time.
  - TVEs are feature effects on the outcome (e.g., hazard rate) that vary over time.
  - Both TVFs and TVEs challenge the proportional hazards (PH) assumption.
- **High-Dimensional Data:**
  - Especially relevant in life sciences (e.g., omics data).
  - Requires models with feature selection or penalization to manage high-dimensional input.
- **Multimodal Features:**
  - Incorporates both structured (e.g., clinical data) and unstructured data (e.g., images, text).
  - Special techniques are needed to extract information from multimodal feature sets.

2.3 Estimation in Survival Analysis

- **Data Representation:**

Event times are represented by tuples that include entry and exit times, censoring indicators, and feature vectors. Simplified notation is often used in cases like single events, no truncation, or single-risk models.

1. Parametric Models
2. Semi-parametric Models

2.3 Estimation in Survival Analysis

1. Parametric Models:

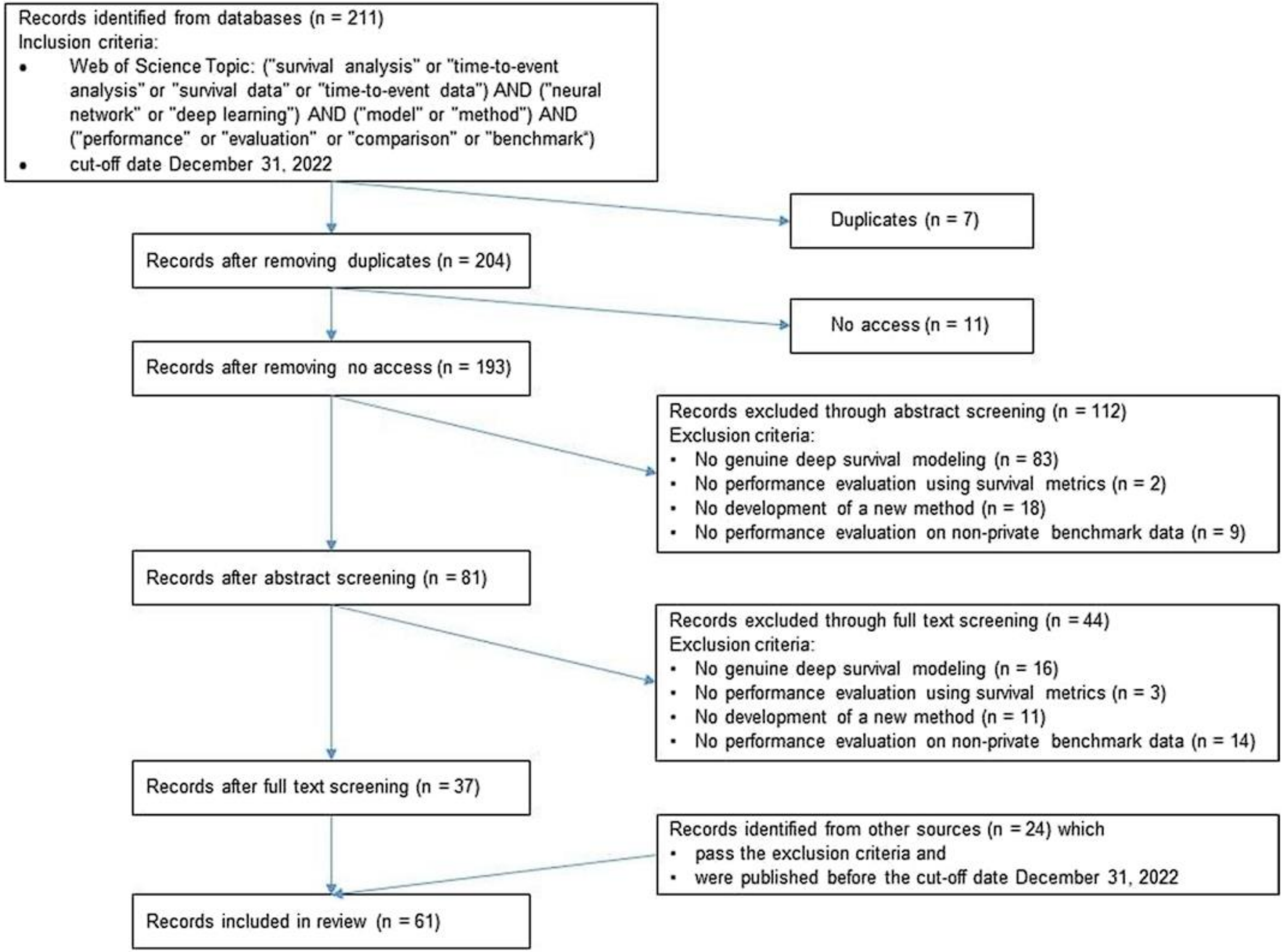
- Examples include the **Accelerated Failure Time (AFT)** model, which assumes event times follow a specific statistical distribution.
- The model estimates distribution parameters as a function of features xxx, using likelihood-based estimation.

- ## Likelihood Function:
  - The likelihood is expressed based on observed event types, considering right-censored, left-censored, and other observation types.
  - Relationships like $S(t)=e^{-\int 0th(u)du}$ allow expressing the likelihood in terms of the hazard rate.

3. Deep Learning in Survival Analysis

1. Feedforward Neural Networks (FFNNs)
2. Convolutional Neural Networks (CNNs)
3. Recurrent Neural Networks (RNNs)
4. Autoencoders (AEs)

3.1 Study Selection



Records identified from databases (n = 211)
Inclusion criteria:

- Web of Science Topic: ("survival analysis" or "time-to-event analysis" or "survival data" or "time-to-event data") AND ("neural network" or "deep learning") AND ("model" or "method") AND ("performance" or "evaluation" or "comparison" or "benchmark")
- cut-off date December 31, 2022

Duplicates (n = 7)

Records after removing duplicates (n = 204)

No access (n = 11)

Records after removing no access (n = 193)

Records excluded through abstract screening (n = 112)
Exclusion criteria:

- No genuine deep survival modeling (n = 83)
- No performance evaluation using survival metrics (n = 2)
- No development of a new method (n = 18)
- No performance evaluation on non-private benchmark data (n = 9)

Records after abstract screening (n = 81)

Records excluded through full text screening (n = 44)
Exclusion criteria:

- No genuine deep survival modeling (n = 16)
- No performance evaluation using survival metrics (n = 3)
- No development of a new method (n = 11)
- No performance evaluation on non-private benchmark data (n = 14)

Records after full text screening (n = 37)

Records identified from other sources (n = 24) which

- pass the exclusion criteria and
- were published before the cut-off date December 31, 2022

Records included in review (n = 61)

Fig. 1 PRISMA diagram for literature screening of deep learning-based survival methods

3.2 Architectural choices

1. **Feedforward Neural Networks (FFNNs):**
   - Earliest type of NN architecture (1967).
   - Information flows only forward through layers.
   - Capable of approximating a wide range of functions (universal approximation theorem).
   - Useful for estimating non-linear effects and complex interactions in survival analysis (SA).
   - Limitations include inability to handle multimodal data like images.
   - Serve as a baseline in many early and advanced DL-based survival methods.
2. **Convolutional Neural Networks (CNNs):**
   - Introduced in the late 1980s, mainly used in computer vision.
   - Applied to unstructured data (e.g., images) in time-to-event analysis.
   - Often involve transfer learning, using pre-trained large CNNs like ResNet18 and fine-tuning them on specific datasets.
3. **Recurrent Neural Networks (RNNs):**
   - Invented in the 1980s, can memorize input sequences.
   - Suitable for handling temporal information and time-varying factors (TVFs) in SA.
4. **Autoencoders (AEs):**
   - Learn to reduce input data dimensionality and reconstruct it from latent representations.
   - Variants include stacked AEs (SAEs) and variational AEs (VAEs).

3.2 Architectural choices

1. **Feedforward Neural Networks (FFNNs):**
   - Earliest type of NN architecture (1967).
   - Information flows only forward through layers.
   - Capable of approximating a wide range of functions (universal approximation theorem).
   - Useful for estimating non-linear effects and complex interactions in survival analysis (SA).
   - Limitations include inability to handle multimodal data like images.
   - Serve as a baseline in many early and advanced DL-based survival methods.

3.2 Architectural choices

2. Convolutional Neural Networks (CNNs):

- Introduced in the late 1980s, mainly used in computer vision.
- Applied to unstructured data (e.g., images) in time-to-event analysis.
- Often involve transfer learning, using pre-trained large CNNs like ResNet18 and fine-tuning them on specific datasets.

3. Recurrent Neural Networks (RNNs):

- Invented in the 1980s, can memorize input sequences.
- Suitable for handling temporal information and time-varying factors (TVFs) in SA.

4. Autoencoders (AEs):

- Learn to reduce input data dimensionality and reconstruct it from latent representations.
- Variants include stacked AEs (SAEs) and variational AEs (VAEs).

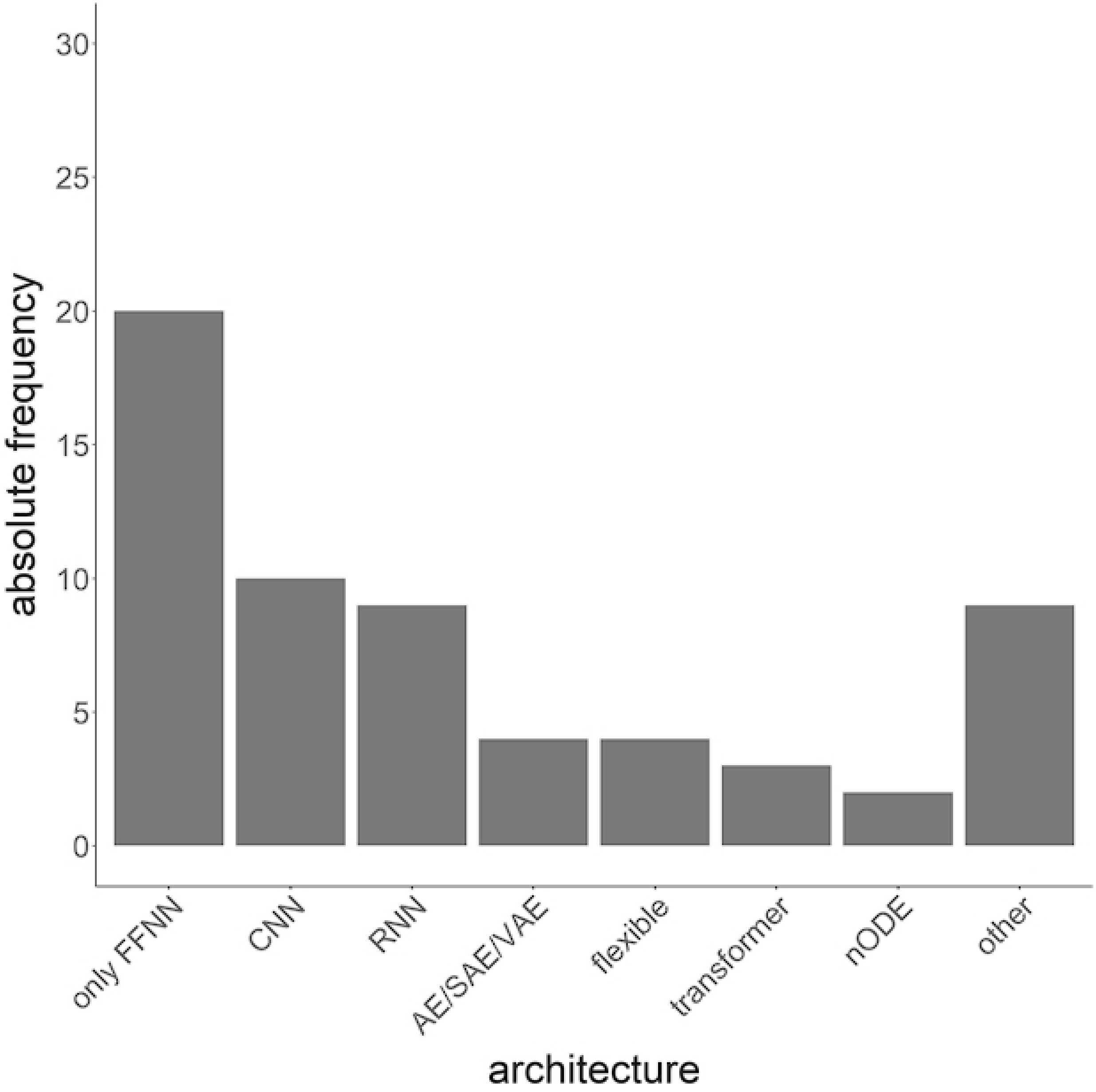


Fig. 5 Absolute frequencies of neural network architectures among all 61 methods reviewed

3.2 Architectural choices

Model Class:

- Methods are categorized by the type of statistical survival technique they are based on (e.g., Cox models, parametric models).
- Loss Functions:
- Often derived from the model class, typically the negative log-likelihood.
- Some methods use multiple loss functions for better performance or multi-task learning.
- Example: Combining a ranking loss with standard survival loss to improve the C-index.
- The final loss is usually a weighted average of all applied losses.

Parametrization:

- Defines which model component is parametrized by the neural network.
- Typically aligned with the model class.

Optimization:

- Most methods are optimized using gradient-based techniques.
- Cox-based methods traditionally require full gradient descent due to dependency on the complete risk set, making them computationally intensive.
- Recent advancements (e.g., Kvamme and Borgan 2019) allow Cox-based methods to be optimized using stochastic gradient descent if the batch size is large enough to approximate the risk set, making them more feasible for batch-wise optimization.

Model Reviews

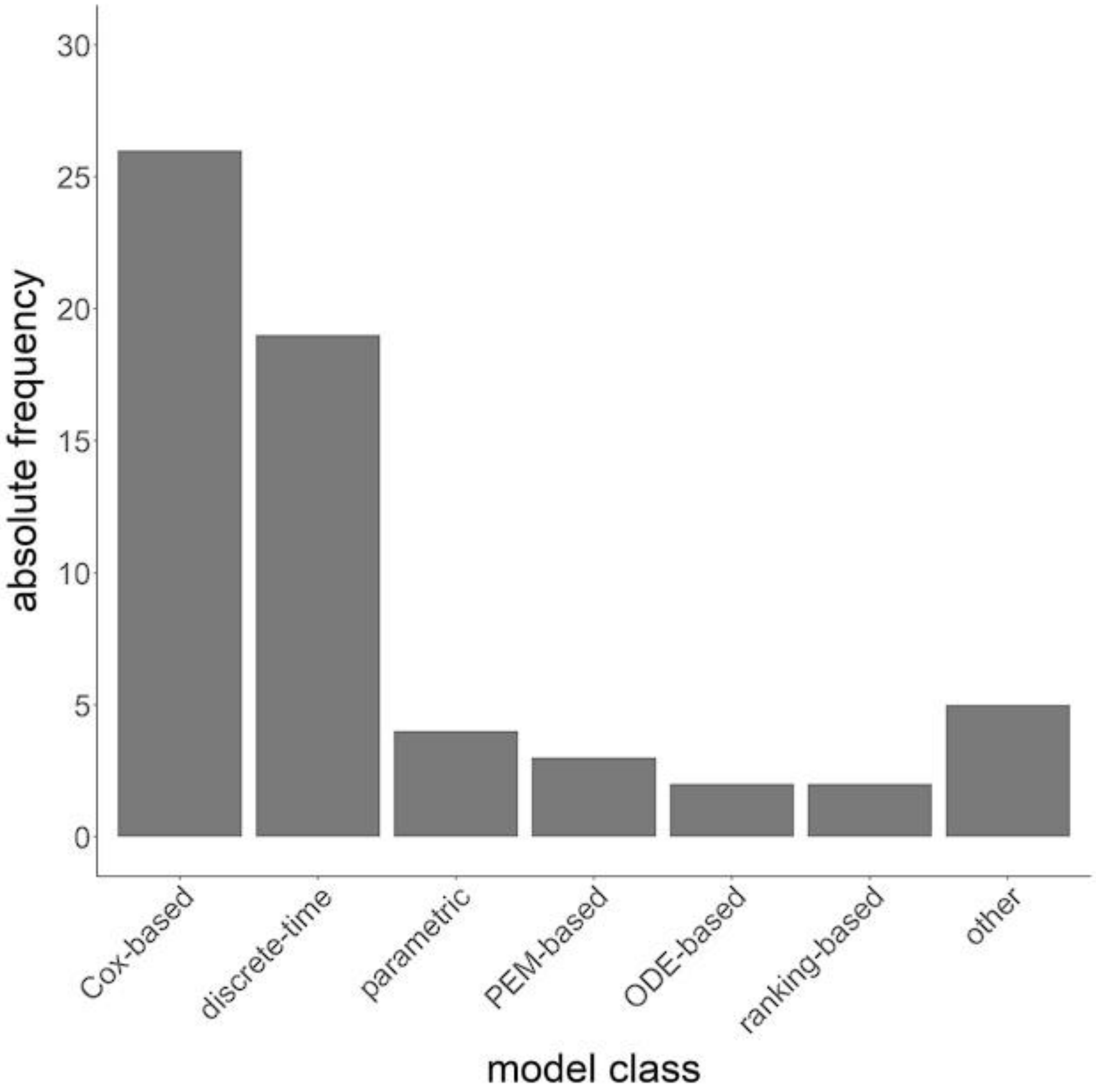


Fig. 4 Absolute frequencies of model classes among all 61 methods reviewed

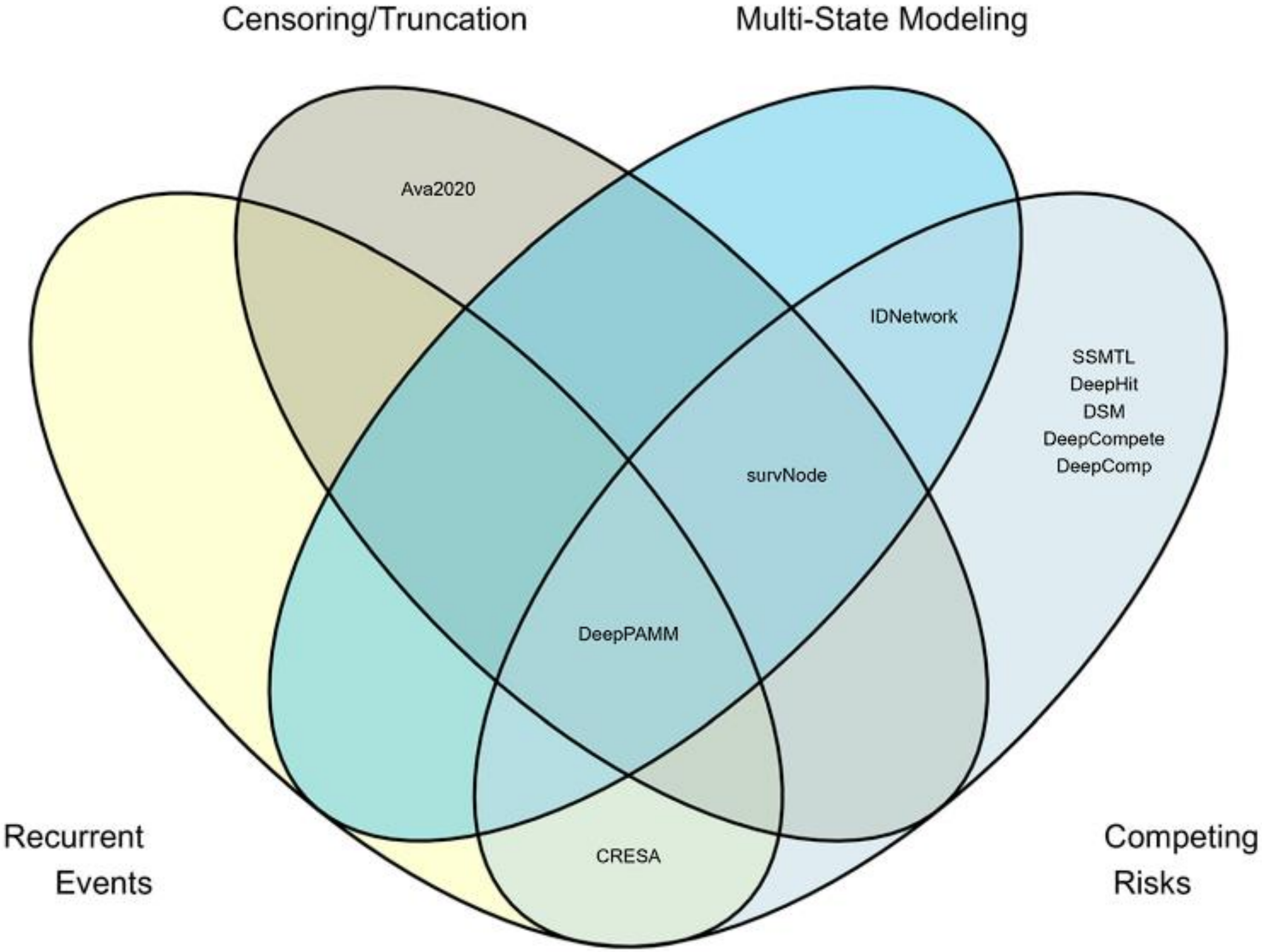


Fig. 6 Venn diagram illustrating which methods can handle the distinct survival outcome types

3.3 Handling of Censoring and Truncation:

- Left-censoring and right-truncation are not specifically addressed by any methods reviewed.
  - **Ava2020:** Handles interval-censored data using Survival-CRPS loss.
  - **survNode:** Briefly mentions handling interval-censoring and left-truncation in likelihood computations.
  - **Deep Survival Machines (DSM):** Framework is adaptable for interval-censoring and left-truncation.
  - **DeepPAMM:** Accounts for left-truncation during data preprocessing.

3.3 Competing Risks:

- Nine methods are designed to address competing risks; none are Cox-based.
  1. **Discrete-time methods:** DeepHit, CRESA, and DeepComp use cause-specific subnetworks.
  2. **DeepHit:** Uses feedforward neural networks (FFNNs) to generate a final distribution over all competing causes.
  3. **CRESA and DeepComp:** Utilize recurrent neural networks (RNNs) for similar purposes, with DeepComp specifically outputting cause-specific discrete hazards.
  4. **SSMTL:** Treats competing risk survival analysis as a multiclass problem with a custom loss function.
  5. **Continuous-time method:** DeepCompete uses neural ordinary differential equation (nODE) blocks to output a cumulative hazard function.
  6. **DSM:** Learns a common representation for all risks, treating competing events as censoring.
  7. **Markov-based methods:** survNode and IDNetwork use illness-death and Markov jump processes, respectively.
  8. **PEM-based:** DeepPAMM parametrizes hazard rates and models competing risks and multi-state outcomes.

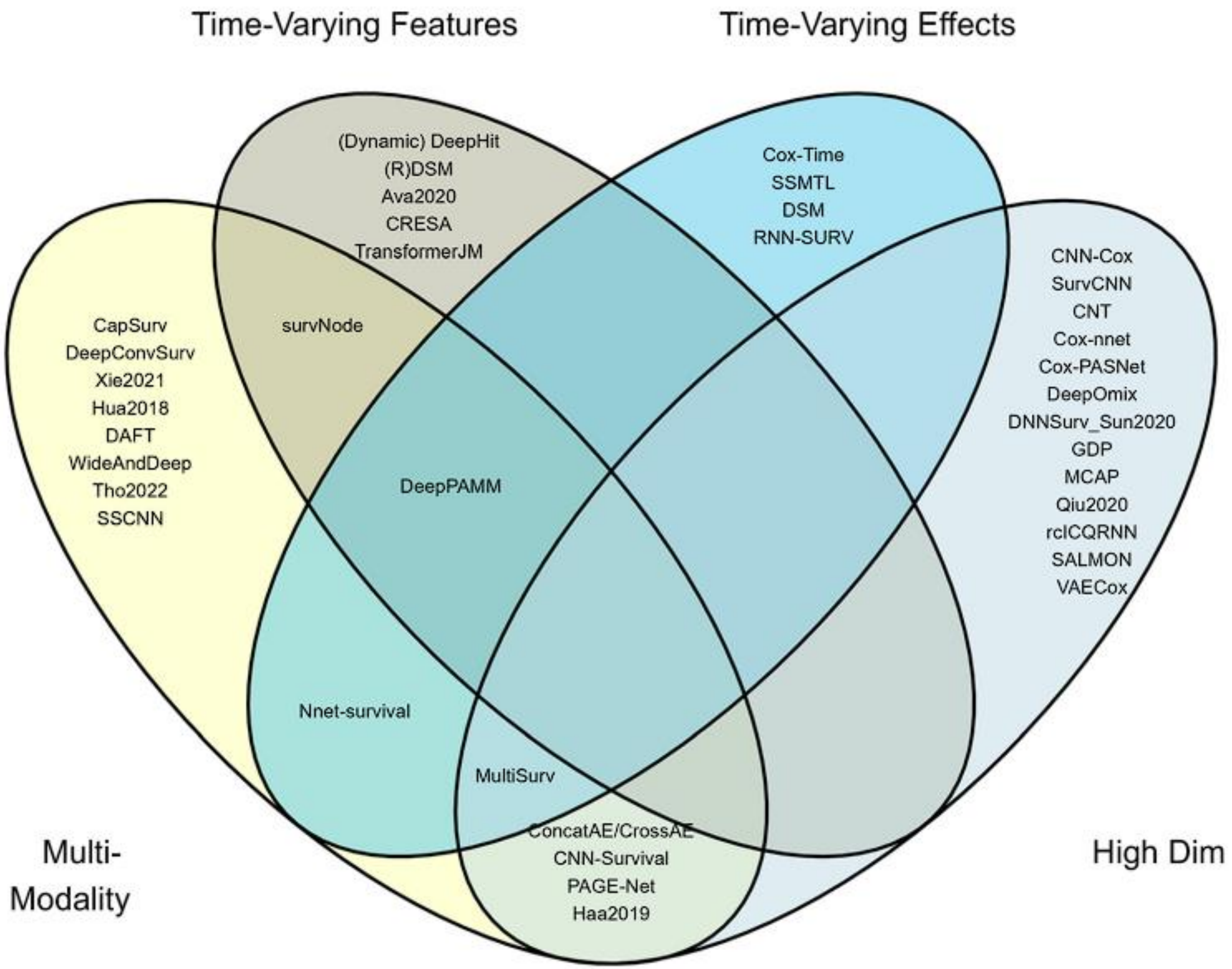


Fig. 7 Venn diagram illustrating which methods can handle the distinct survival feature-related aspects

3.4 Supported Feature-Related Aspects:

- Time-Dependent Features (TVFs):
  - Address deviation from the proportional hazards (PH) assumption in traditional survival models.
  - Techniques:
    - **DeepPAMM:** Converts tabular time-varying feature input into long format.
    - **survNode:** Uses RNNs to process new feature measurements.

- Time-Varying Effects (TVEs):
  - Seven methods model effects that are not constant over time.
  - **Discrete-time approaches:** Nnet-survival, MultiSurv, RNN-Surv.
    - **Nnet-survival and MultiSurv:** Use a fully connected neural network (NN) to model TVEs.
    - **RNN-Surv:** Captures TVEs with its RNN architecture.

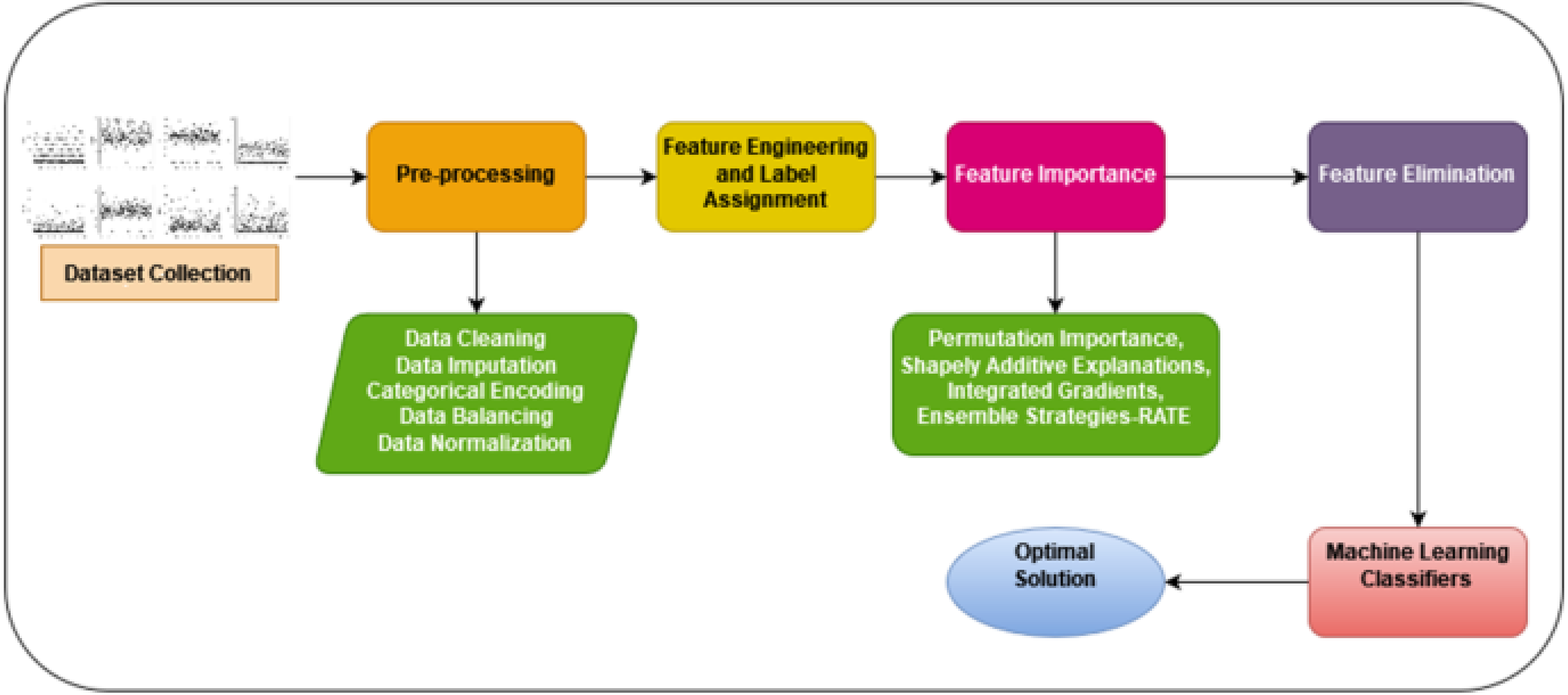
3.4 Supported Feature-Related Aspects:

- Other methods:
  - Cox-Time: Uses a time-dependent NN-parametrized relative risk.
  - DeepPAMM: Models TVEs through interaction between follow-up time and other features.
  - DSM and SSMTL: Mention TVEs but do not provide detailed methodology.

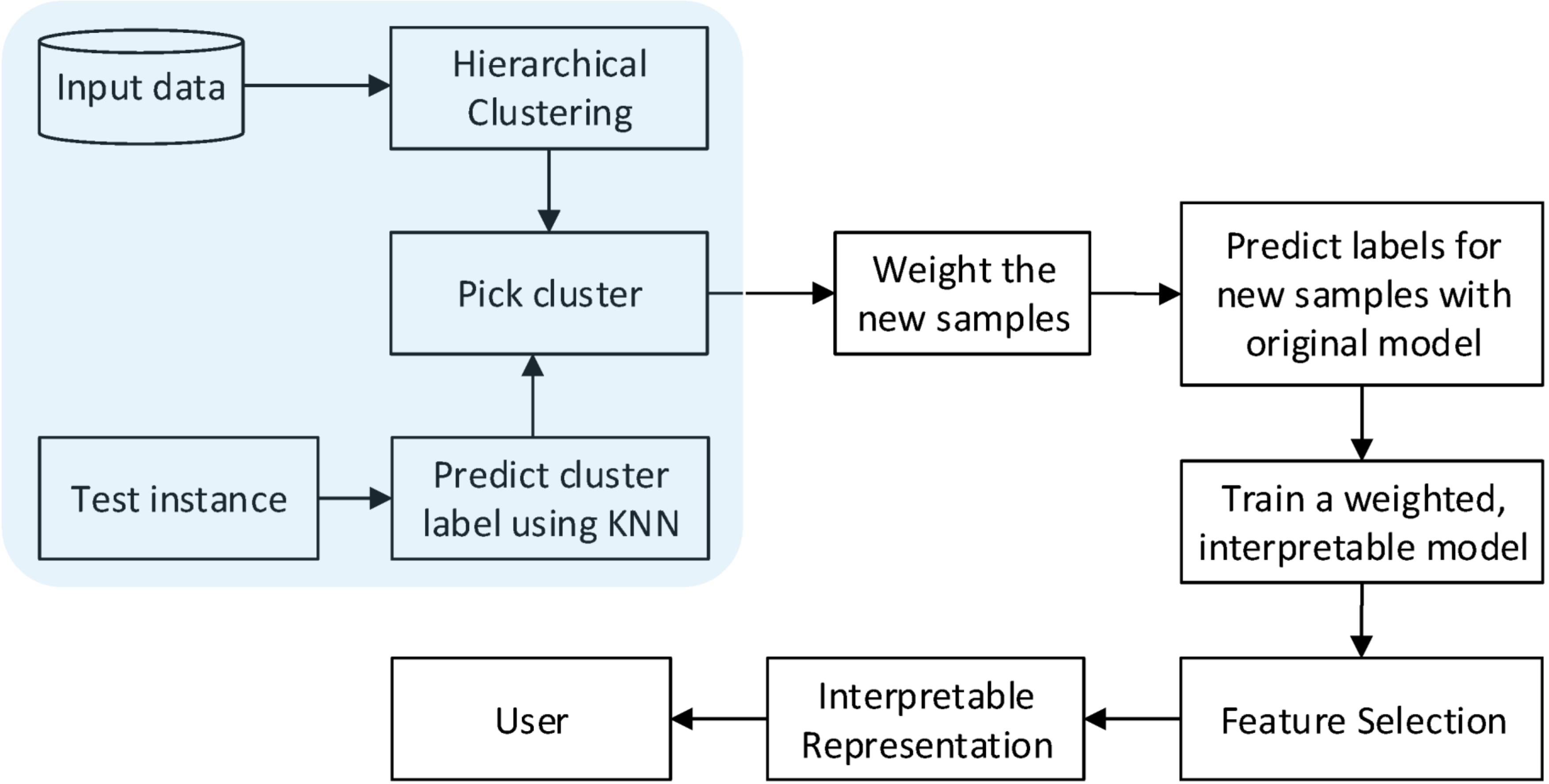
3.5 Post-Hoc Interpretability Methods:

- Several machine learning methods can be applied to DL survival models:
  - Permutation Feature Importance
  - Local Interpretable Model-agnostic Explanations (LIME)
  - Shapley Additive ExPlanations (SHAP)
  - Attention Maps
  - Layer-Wise Relevance Propagation (LRP)

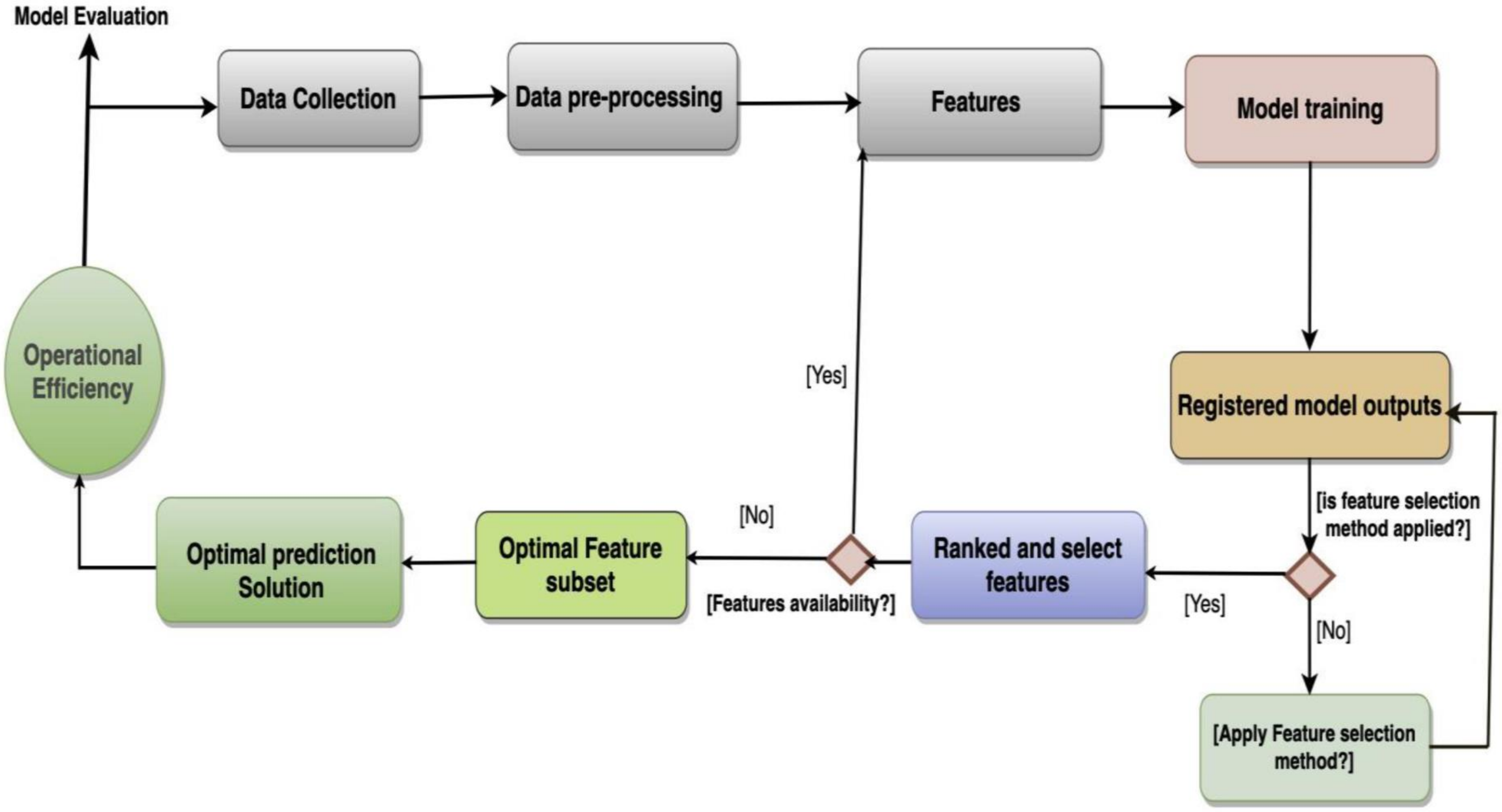
Permutation Feature Importance



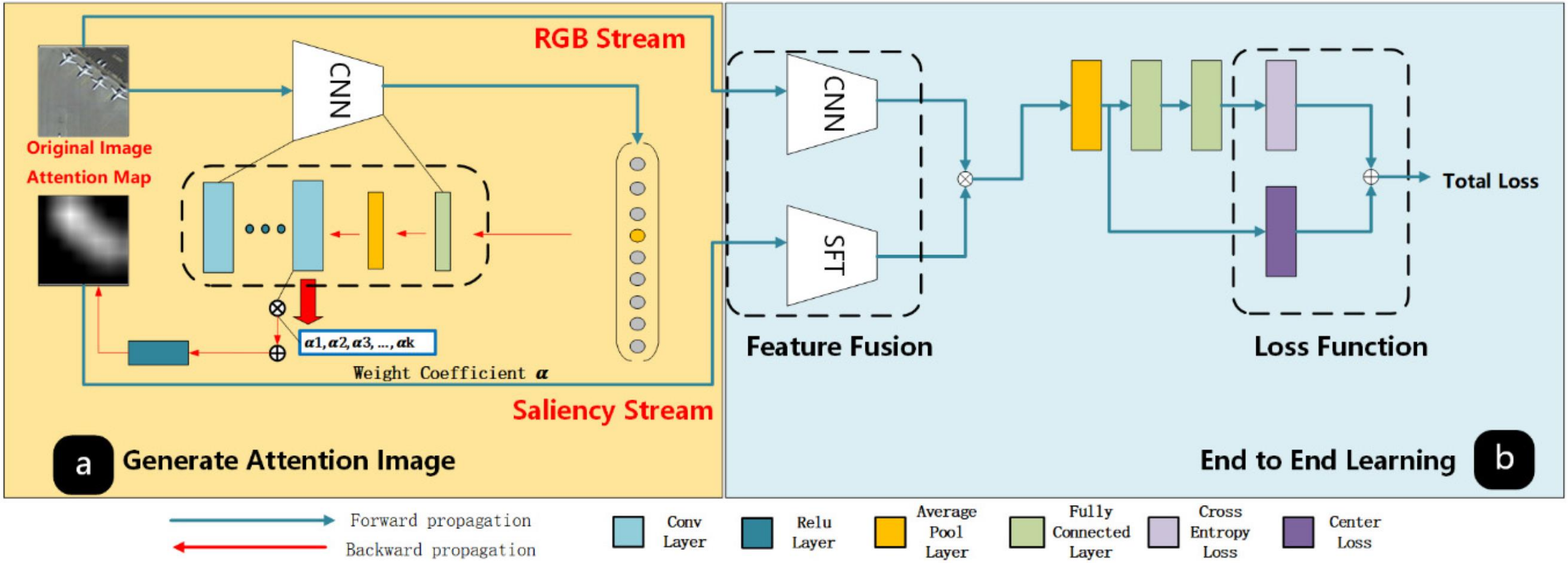
Local Interpretable Model-agnostic Explanations (LIME)


Input data
Hierarchical Clustering
Pick cluster
Test instance
Predict cluster label using KNN
Weight the new samples
Predict labels for new samples with original model
Train a weighted, interpretable model
Feature Selection
Interpretable Representation
User

Shapley Additive ExPlanations (SHAP)



Attention Maps



Layer-Wise Relevance Propagation (LRP)

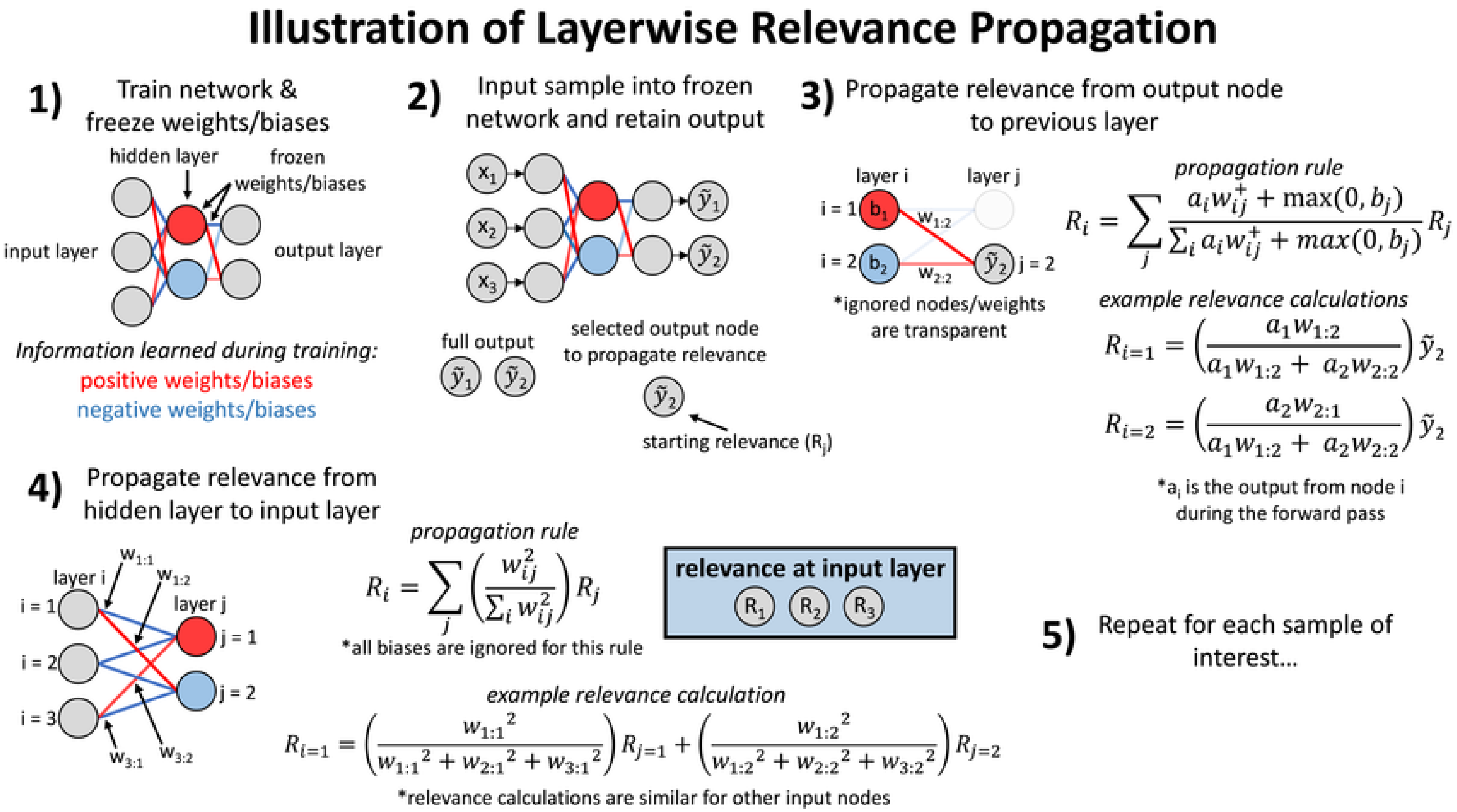


Illustration of Layerwise Relevance Propagation

1) Train network & freeze weights/biases

Information learned during training:
positive weights/biases
negative weights/biases

2) Input sample into frozen network and retain output

x_1
x_2
x_3
$\tilde{y}_1$
$\tilde{y}_2$
full output
$\tilde{y}_1$ $\tilde{y}_2$
selected output node to propagate relevance
$\tilde{y}_2$
starting relevance (R_j)

3) Propagate relevance from output node to previous layer

*ignored nodes/weights are transparent

propagation rule

$$R_i = \sum_j \frac{a_i w_{ij}^+ + \max(0, b_j)}{\sum_i a_i w_{ij}^+ + max(0, b_j)} R_j$$

example relevance calculations

$$R_{i=1} = \left(\frac{a_1 w_{1:2}}{a_1 w_{1:2} + a_2 w_{2:2}}\right) \tilde{y}_2$$

$$R_{i=2} = \left(\frac{a_2 w_{2:1}}{a_1 w_{1:2} + a_2 w_{2:2}}\right) \tilde{y}_2$$

*a_i is the output from node i during the forward pass

4) Propagate relevance from hidden layer to input layer

propagation rule

$$R_i = \sum_j \left(\frac{w_{ij}^2}{\sum_i w_{ij}^2}\right) R_j$$

*all biases are ignored for this rule

relevance at input layer

example relevance calculation

$$R_{i=1} = \left(\frac{{w_{1:1}}^2}{{w_{1:1}}^2 + {w_{2:1}}^2 + {w_{3:1}}^2}\right) R_{j=1} + \left(\frac{{w_{1:2}}^2}{{w_{1:2}}^2 + {w_{2:2}}^2 + {w_{3:2}}^2}\right) R_{j=2}$$

*relevance calculations are similar for other input nodes

5) Repeat for each sample of interest...

3.6 Model Evaluation

- Common Evaluation Metrics:
  - C-index: The most popular metric for assessing risk predictions in survival analysis, typically using Harrell’s.
  - Brier/Graf Score: Used for evaluating distribution predictions.

4 Conclusion

- **Overall Conclusion:**
  - Deep survival methods have made significant progress recently and will likely see further advances as new ML/DL methodologies emerge.
- **Observations on DL-based Survival Methods**:
  - Many DL-based survival methods are adaptations of techniques developed in other areas of DL, like computer vision or NLP.
  - These adaptations focus on flexibly estimating associations between features and outcomes, rather than addressing unique challenges of time-to-event data not covered by traditional statistical approaches.
  - Outcome types beyond right-censoring and competing risks are rarely considered, likely due to limited applications.

4 Conclusion

- **Challenges in DL-based Survival Analysis**:
  - Limited focus on optimization (e.g., choice of optimizers, hyperparameter tuning, or neural architecture search) in the reviewed methods. The Adam optimizer is the most commonly used.
  - Common survival analysis losses, such as log-likelihood-based losses, can be poorly conditioned, posing optimization challenges.
  - Batching strategies may need to be adapted based on the specific survival analysis task, such as recurrent events.

- **Future Directions**:
  - DL-based survival analysis will continue to benefit from advances in machine learning and deep learning.
  - Generative DL techniques, such as diffusion models, are promising for adaptation to survival tasks.
  - The research community is encouraged to contribute to the ongoing development of resources like the open-source interactive table to keep the field up-to-date.