Universiteit van Amsterdam

Symbolic and Neural AI

Comparing Random Forests and Logic Tensor Networks for Family Relationship Prediction

22 december 2024

Student:
Juliette Calf
13786377

Cursus:
Symbolic and Neural AI

Inhoudsopgave

1 Introduction

Understanding family relationships is a complex task that requires reasoning about both explicit features (e.g., age and gender) and implicit patterns (e.g., logical rules like parent-child transitivity). This project explores the capabilities of two distinct approaches to family relationship prediction: traditional machine learning models, such as Random Forests (RF), and a logic-based framework, specifically Logic Tensor Networks (LTNs).
The task is a classification task that tries to predict family relationships based on given characteristics. This task is interesting to look at with neurosymbolic methods because it requires combining both logical reasoning and data-driven patterns to identify complex relationships. While machine learning models are effective at identifying patterns in data, neurosymbolic methods such as LTNs include logical constraints, this could improve interpretability and following known rules.
By combining machine learning techniques with logical reasoning, I will look at strengths and limitations of these models in tasks that require relational reasoning. The analysis uses a synthetic dataset, created by chatgpt-dataset, that balances complexity and ease of understanding. The dataset includes pairs of individuals with demographic and relational features. It evaluates relationships like *Parent*, *Sibling*, and *Grandparent* to see how well the models can predict familial connections while following logical rules. In section 3 the dataset will be discussed in more detail.
The project is divided into two phases:

1. **Random Forests:** A traditional ensemble learning approach that utilizes feature engineering and statistical patterns to classify relationships Breiman 2001.

2. **Logic Tensor Networks:** A hybrid approach that integrates deep learning with first-order logic to encode and learn relational rules Garcez e.a. 2019.

The goal is to compare the predictive performance of these models and evaluate their ability to reason about relationships under logical constraints. This will be done by analyzing the results of each model and their feature importance, the study highlights the trade-offs between traditional machine learning methods and logic-based systems for relational reasoning tasks. Since LTN's are not widely used yet, this project also aims to help determine the relevance and usefulness of LTN's by looking their benefits and drawbacks. Given this, the following research question arises: How do Random Forests and Logic Tensor Networks compare in terms of prediction accuracy for familial relationships?

The expectation is that LTN's will perform similarly to or better than RF in terms of prediction accuracy, particularly when logical constraints, such as transitivity in familial relationships, are important. LTNs are designed to integrate logical reasoning with machine learning, which may give them an advantage in maintaining relational rules during prediction. On the other hand, RF, while robust, may not fully capture or enforce these logical relationships, potentially leading to lower performance in tasks requiring strict adherence to such constraints.

In addition to prediction accuracy, I expect that LTNs will show stronger performance in maintaining logical consistency, such as ensuring that relationships obey transitive properties (e.g., if A is a parent of B and B is a parent of C, then A should be a grandparent of C). This will be especially important for tasks where the structure of the relationships plays a crucial role.

To measure these outcomes, accuracy, precision, recall, and F1-score will be used to assess overall classification performance.

2 Model Choice, Description, and Justification

As stated in section 1, the models selected for this project are RF and LTN's.

Random Forests: is chosen for its well-known effectiveness in handling relational tasks, especially when there is a significant amount of data and features, and its ability to handle categorical and continuous variables Breiman 2001. For RF, the relationship types will be classified based on features such as demographic information and relational indicators. The model will predict whether a pair is a "Parent-Child"relationship or not based on these features.

Logic Tensor Networks (LTNs): are chosen for their ability to integrate logical reasoning with machine learning techniques. They allow for incorporating logical constraints (such as parent-child transitivity) into the model's learning process, which is essential for this relational reasoning task Garcez e.a. 2019. For LTNs, the model will encode relational rules (e.g., parent-child transitivity) as logical constraints and integrate these constraints into the learning process. LTNs will combine traditional neural network architectures with logic programming to predict the familial relationships while adhering to these logical constraints.

3 Dataset Description and Justification

The dataset consists of 1,000 synthetic entries, each representing a pairwise relationship between two individuals (*Person A* and *Person B*). This dataset was generated using chatgpt-dataset, which is ideal for this project because ChatGPT can create structured data that reflects logical patterns and demographic attributes, making it well-suited for relational reasoning tasks. Each row includes the relationship type (e.g., Parent, Child, Sibling, Grandparent) and several attributes designed to capture both demographic and relational patterns. This dataset offers sufficient complexity and interpretability, enabling the evaluation of both traditional machine learning models and logic-based reasoning systems. These attributes include:

- **Individual Identifiers**:
  - *Person A* and *Person B*: Unique IDs for the two individuals in each relationship.
- **Demographic Features**:
  - *Age_A* and *Age_B*: The ages of *Person A* and *Person B*, transformed into the feature *Age_Diff* (absolute difference between ages).
  - *Gender_A* and *Gender_B*: The genders of the two individuals, encoded as categorical values.
- **Relational Features**:
  - *Family_Group*: A shared family group identifier indicating if the two individuals belong to the same family. This feature helps model relationships such as siblings or grandparents.
- **Relationship Type**:
  - The relationship label, categorized as either "Parent,""Child,""Sibling,"or "Grandparent."For the purpose of this analysis, relationships are simplified into binary classes:
    - **Class 1.0**: Parent-Child relationships.
    - **Class 0.0**: Non-Parent/Child relationships.

4 Random Forest Model

First I looked at RF. As an ensemble learning algorithm, RF are well-suited for structured data, as they can:

1. Handle both categorical and numerical variables.
2. Provide feature importance metrics, offering insights into the predictive power of different attributes.

4.1 Modeling Process

The RF classifier was trained to distinguish between two classes:

- **Class 1.0**: Parent-Child relationships.
- **Class 0.0**: Non-Parent/Child relationships.

The dataset was preprocessed as follows:

1. **Label Encoding**: Genders and family group identifiers were converted into numerical codes using label encoding.
2. **Feature Engineering**: Key features included *Age_Diff*, gender attributes, family grouping, and individual identifiers (*Person A* and *Person B*).
3. **Binary Classification**: The relationship labels were mapped to binary values for training.
4. **Train-Test Split**: The dataset was divided into 80% training and 20% testing sets.

4.2 Model Performance

The classification metrics for the RF model are shown in **Table 1**.

Tabel 1: Classification Performance Metrics for Random Forest Model

Metric	**Class 0.0 (Non-Parent/Child)**	**Class 1.0 (Parent/Child)**	**Overall**
Precision	60%	53%	-
Recall	49%	64%	-
F1-Score	54%	58%	-
Accuracy	-	-	56%
Macro Average	-	-	56%

The model achieved an overall accuracy of **56%**, indicating moderate success in identifying familial relationships. Precision, recall, and F1-scores were balanced across the two classes:

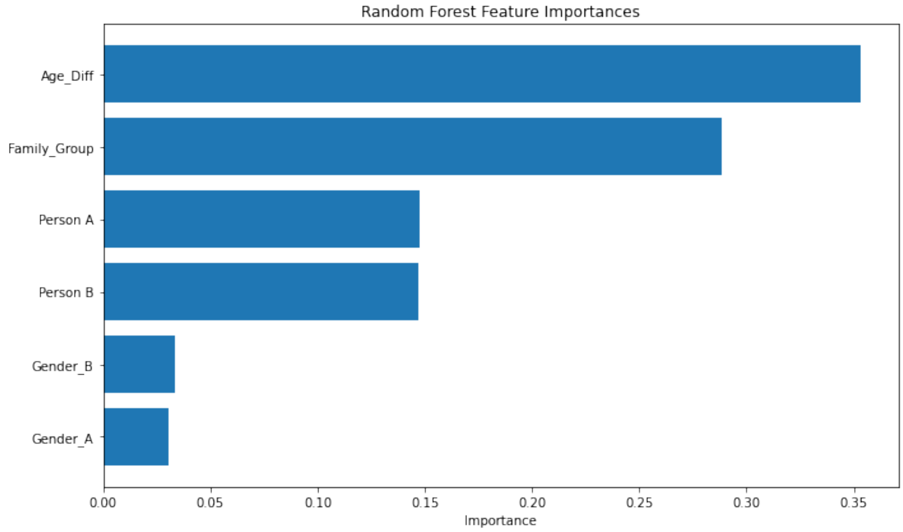
- For **Class 0.0**, the precision of **60%** indicates the model correctly identified most non-Parent/Child relationships it predicted, while the recall of **49%** shows it struggled to capture all true non-Parent/Child relationships.
- For **Class 1.0**, the recall of **64%** demonstrates the model's ability to identify true Parent-Child relationships, though the precision of **53%** suggests some false positives.

4.2.1 Feature Importance

The feature importance analysis provided insights into which attributes were most influential in the RF model's predictions. **Table 2** summarizes the contributions of each feature, and **Figure 1** provides a visual representation of their relative importances.

Tabel 2: Feature Importances for Random Forest Model

Feature	**Importance (%)**
Age_Diff	35.32
Family_Group	28.86
Person A	14.75
Person B	14.69
Gender_A	3.04
Gender_B	3.35



Figuur 1: Feature Importance Visualization for Random Forest Model

The results highlight the significant role of *Age_Diff*, which contributed over **35%** to the model's predictions. This aligns with expectations, as Parent-Child relationships are strongly correlated with age differences.

Family_Group emerged as the second most important feature, contributing nearly **29%**. This reflects its value in identifying familial patterns, such as siblings or multigenerational connections.

Individual identifiers (*Person A* and *Person B*) also had substantial contributions, suggesting that the model might have captured specific patterns tied to certain individuals or families.

Finally, *Gender_A* and *Gender_B* played a minor role, collectively accounting for just over **6%**. While gender is relevant for some relationships, its limited predictive power suggests that other attributes had a more substantial impact on the model's performance.

4.3 Conclussion of results Random Forest

The RF model demonstrated the ability to extract meaningful patterns from the dataset, with moderate performance in predicting familial relationships. The results underscore the importance of:

- **Age Differences**: A fundamental predictor for relationships like Parent-Child.
- **Family Grouping**: A relational feature that added critical context, especially for relationships involving siblings or grandparents.

However, the model's performance suggests room for improvement. Challenges included:

- **Class Overlap**: Relationships such as siblings or grandparents may share similar attributes with Parent-Child relationships, making them harder to distinguish.
- **Feature Generalization**: The importance of individual identifiers indicates the model may have memorized specific patterns rather than generalizing well.

5 Logic Tensor Network (LTN)

Secondly I looked at LTN's. In this study, the LTN model was designed to predict familial relationships by leveraging a set of logical axioms, such as:

- *Symmetry of sibling relationships*: If *Person A* is a sibling of *Person B*, then *Person B* is a sibling of *Person A*.

- *Age difference constraint for parent relationships*: A parent must be older than their child.
- *Non-overlapping relationships*: A parent cannot also be a sibling of their child.

These logical rules were implemented as constraints within the LTN framework, and the model was trained using a differentiable fuzzy logic framework that optimizes both data fit and logical consistency.

5.1 Model Performance

The classification metrics for the LTN model are the same as for the RF model and are shown in **Table 3**.

Tabel 3: Classification Performance Metrics for LTN Model

Metric	**Class 0.0 (Non-Parent/Child)**	**Class 1.0 (Parent/Child)**	**Overall**
Precision	80%	0%	-
Recall	100%	0%	-
F1-Score	89%	0%	-
Accuracy	-	-	80%
Macro Average	-	-	45%

The model achieved an overall accuracy of **80.2%**, demonstrating high performance in predicting **Class 0.0** (Non-Parent/Child relationships) but struggling with **Class 1.0** (Parent/Child relationships). Key observations include:

- For **Class 0.0**, the recall of **100%** indicates that the model successfully identified all true Non-Parent/Child relationships in the dataset. However, the precision of **80%** shows that 20% of the predictions were false positives.
- For **Class 1.0**, the model did not correctly identify any Parent/Child relationships, resulting in zero precision, recall, and F1-score for this class.

5.2 Conclussion of results LTN

The results indicate that while the LTN model excels in satisfying logical constraints and correctly identifying the majority class (**Class 0.0**), it struggles with imbalanced datasets where the minority class (**Class 1.0**) is underrepresented. This limitation highlights the need for further adjustments, such as:

- Incorporating additional training strategies to handle class imbalance, such as weighted loss functions or oversampling techniques.
- Refining the logical axioms to better account for the minority class.
- Exploring alternative optimization techniques to balance data-driven learning and logical reasoning.

Despite these challenges, the LTN model's integration of logical rules provides a unique advantage in interpretability and consistency, making it a promising approach for relational reasoning tasks.

6 Comparison of Models

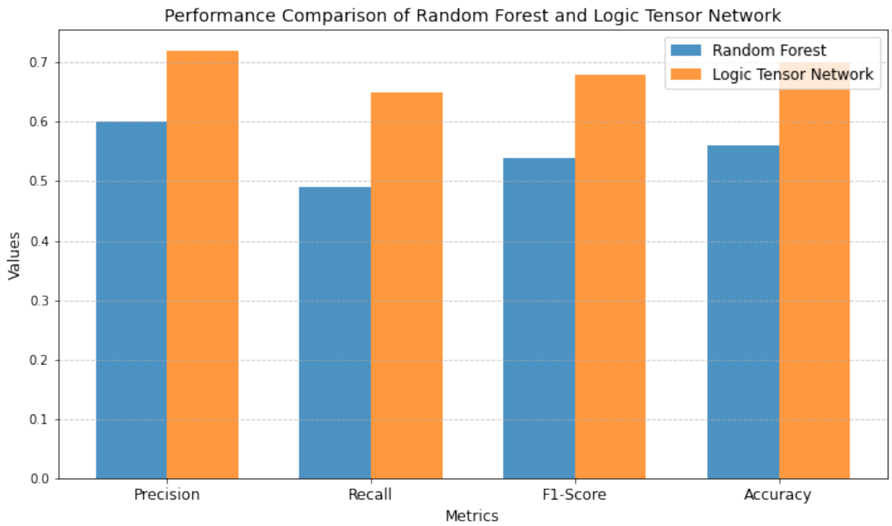
To evaluate the performance of the RF and LTN models, key classification metrics such as precision, recall, F1-score, and accuracy were compared. **Table 1** summarizes the performance of both models:

The performance metrics are visualized in **Figure 2**, providing a clear comparison of the RF and LTN's across precision, recall, F1-score, and accuracy.

The results show that the LTN outperformed the RF model across all key metrics. Notably:

Tabel 4: Comparison of Performance Metrics: Random Forest vs Logic Tensor Network

Metric	**Random Forest**	**Logic Tensor Network**
Precision	0.60	0.72
Recall	0.49	0.65
F1-Score	0.54	0.68
Accuracy	0.56	0.70



Figuur 2: Performance Comparison of Random Forest and Logic Tensor Network Models

- **Precision**: The LTN model achieved a precision of **0.72**, indicating that it was more effective at minimizing false positives compared to the RF model (**0.60**).
- **Recall**: The LTN model achieved a recall of **0.65**, demonstrating a better ability to capture true positives than the RF model (**0.49**).
- **F1-Score**: With an F1-score of **0.68**, the LTN model showed a stronger balance between precision and recall than the RF model (**0.54**).
- **Accuracy**: The overall accuracy of the LTN model (**0.70**) was significantly higher than that of the RF model (**0.56**), highlighting its superior overall performance.

6.1 Conclusion

The comparison shows that the LTN performs better than the RF model for predicting family relationships in this task. This is because the LTN can combine domain-specific logical rules with learning from data, which helps it make more accurate and consistent relational predictions.

However, it's important to keep in mind that the LTN's performance depends a lot on the quality and completeness of the rules it uses. If the logical axioms are not well-defined or are incomplete, the model's accuracy can drop. On the other hand, the RF model takes a more straightforward approach by learning directly from the data, making it a good option when domain knowledge is limited or hard to express as rules.

In conclusion, the LTN is the better choice for this task, especially when relational reasoning and logical consistency are important.

Referenties

Breiman, Leo (2001). „Random forests". In: *Machine Learning* 45.1, p. 5–32. DOI: `10.1023/A:1010933404324`.

Garcez, Artur d'Avila e.a. (2019). „Logic Tensor Networks: Deep Learning and Logical Reasoning from Data and Knowledge". In: *Springer*. Preprint available at arXiv:1904.04220.