Comparative Analysis of ID3 Decision Tree on Mushroom, Tic-Tac-Toe, and Nursery Datasets

1. Performance Comparison

From all the given three datasets, the ID3 decision tree gave different levels of performance depending on the data.

● **Mushroom**: Both PyTorch and scikit-learn reached 100% accuracy, with precision, recall, and F1-score also was perfect. The “odor” attribute almost fully determines the class, making the dataset very easy to separate.

● **Tic-Tac-Toe:** PyTorch achieved 87.30%, while scikit-learn was slightly better at 88.36%. Precision and recall were decent but not as high as mushroom, since there are many board patterns that are harder to distinguish.

● **Nursery:** PyTorch gave 98.67%, and scikit-learn slightly higher at 98.87%. Precision and recall were strong for majority classes, but lower for minority ones.

Overall, the accuracy ranking is: Mushroom (100%) > Nursery (~99%) > Tic-Tac-Toe

2. Tree Characteristics Analysis

The mushroom decision tree was shallow, usually 3–5 levels, with few nodes. The key attribute was odor, which almost completely split the data.

● Tic-Tac-Toe trees were deeper, about 7–9 levels, since all nine board cells matter. The middle position was often selected first because of its importance in deciding the outcome.

● Nursery produced the deepest and largest trees, usually above 10 levels. Features like parents and finance were chosen early, and the tree size reflected the dataset’s complexity and many attribute values.

3. Dataset-Specific Insights

● **Mushroom**: Odor is the decisive feature. The dataset is balanced, and the tree gives very clear decision rules (e.g., “if odor = foul → poisonous”). Overfitting is not an issue.

● **Tic-Tac-Toe:** Middle and diagonal cells matter most. The dataset is balanced. However, the tree sometimes overfits by memorizing exact board states.

● **Nursery:** Parents and finance strongly influence the outcome. The data is imbalanced, with “not_recom” dominating. Trees are large and harder to interpret. Signs of overfitting appear because of the high depth.

4. Comparative Analysis Algorithm performance:

Mushroom was perfect because of its very strong features. Nursery also did very well, showing that with enough examples, trees can handle multiple classes. Tic-Tac-Toe was weakest because of the large number of board states. Larger datasets like nursery created deeper trees, while mushroom stayed shallow due to fewer but stronger attributes.

- Impact of data characteristics: Class imbalance reduced performance in nursery for smaller classes. Binary features like in Tic-Tac-Toe made splits simpler, while multi-valued features in nursery made the tree more complex.

- Practical applications: Mushroom is relevant for food safety where interpretability is important. Tic-Tac-Toe is more for demonstrating AI in simple games. Nursery can be used in decision support for admissions or recommendations. Interpretability was best in mushroom, moderate in Tic-Tac-Toe, and hardest in nursery.

- Improving performance: Mushroom does not need improvement. Tic-Tac-Toe could be improved by pruning or limiting depth to avoid memorization. Nursery could be improved by handling imbalance with resampling, pruning, or using ensemble methods like random forests.