Initial Results

GRU Results for CPU Usage

Service	Test Size	Train RMSE	Train R^2	Test RMSE	Test R^2
Service 1	0.3	0.0024	0.9933	0.0011	0.9983
	0.2	0.0021	0.9943	0.0009	0.9975
	0.1	0.002	0.995	0.0008	0.9929
Service 2	0.3	0.0008	0.9992	0.0006	0.9993
	0.2	0.0009	0.999	0.0005	0.9988
	0.1	0.0008	0.9992	0.0005	0.9959
HashGen	0.3	0.0004	0.9983	0.0005	0.9986
	0.2	0.0004	0.9983	0.0004	0.999
	0.1	0.0004	0.9982	0.0006	0.9973
RandPw	0.3	0.0003	0.9756	0.0002	0.9733
	0.2	0.0003	0.9733	0.0002	0.9714
	0.1	0.0003	0.9763	0.0002	0.9703

GRU Results for Memory Usage

Service	Test Size	Train RMSE	Train R^2	Test RMSE	Test R^2
Service 1	0.3	4,102,713.63	0.8669	5,061,750.44	0.4571
	0.2	5,749,203.37	0.7071	6,654,237.67	0.27
	0.1	3,616,185.57	0.874	8,536,727.26	0.2971
Service 2	0.3	181,455.88	0.5951	183,638.31	0.5057
	0.2	180,699.28	0.6044	183,013.99	0.4918
	0.1	183,227.14	0.5924	171,506.40	0.4293
HashGen	0.3	11,937,950.46	0.6149	15,315,091.39	0.6177
	0.2	10,995,190.73	0.6498	16,927,584.73	0.6405
	0.1	10,400,575.47	0.653	23,956,365.12	0.5581
RandPw	0.3	903,108.78	0.9826	740,339.51	0.9929
	0.2	907,416.74	0.9813	953,970.08	0.9898
	0.1	881,079.65	0.987	351,436.81	0.988

Service 1

CPU Usage:

- Consistently excellent results with R2 values above 0.99 and very low RMSE across all test sizes.

- This indicates the GRU model captures CPU usage patterns very well and generalizes effectively to unseen data.

Memory Usage:

- R2 values are moderate to low (ranging roughly 0.27 to 0.87), and RMSE values are very high (millions).
- The high RMSE despite moderate R2 suggests that memory usage values have a very large range or outliers, inflating RMSE.
- The lower R2 on test sets (especially 0.27-0.46) indicates the model struggles to generalize well on memory data, possibly due to high volatility or noise in memory usage patterns.
- The discrepancy is likely because memory usage data varies on a much larger numeric scale than CPU usage, so RMSE appears large while R2 reflects relative fit.

Service 2

CPU Usage:

- Exceptionally strong results with R2 consistently near 0.999 and very low RMSE.
- Indicates the model fits CPU usage with high accuracy and robustness across data splits.

Memory Usage:

- R2 is moderate (~0.5 to 0.6) with relatively high RMSE compared to CPU.
- Similar to Service 1, the memory data has large numeric values leading to higher RMSE, while moderate R2 indicates partial but imperfect modeling.
- This gap suggests memory usage is inherently noisier or less predictable, or the model architecture/hyperparameters might need tuning for memory.

HashGen Service

CPU Usage:

- Strong performance with R2 around 0.998 and low RMSE, showing the model generalizes well.
- CPU usage has consistent, learnable patterns here as well.

Memory Usage:

- R2 values around 0.6-0.65 with extremely high RMSE (10+ million).
- Again, the high RMSE signals that memory values have a large range or heavy-tailed distribution, making RMSE large despite moderate R2.
- The model captures some variance in memory usage but may not handle spikes or large outliers well.

RandPw Service

CPU Usage:

- Very good R2 (around 0.97) with very low RMSE, consistent with other CPU usage models.
- Predictable CPU usage behavior.

Memory Usage:

- Exceptionally high R2 (0.98-0.99) and comparatively lower RMSE (hundreds of thousands, which is much lower than other services).
- Indicates both relative fit and absolute error are well controlled here, suggesting memory usage is more stable or the model is better suited for this data.
- This service shows the best memory prediction results, likely due to more consistent or normalized memory usage patterns.

Overfitting

Service 1:

- **CPU Usage**:
  - Train R^2: ~0.9998
  - Test R^2: ~0.997–0.9995 → Very close → No signs of overfitting
- **Memory Usage**:
  - Train R^2: ~0.99+
  - Test R^2: ~0.27 to 0.86 → Large gap between train and test performance

Conclusion: Memory usage model is overfitting. The model is not generalizing well to test data.

Service 2:

- **CPU Usage**:
  - Train R^2: ~0.9996+
  - Test R^2: ~0.9985–0.9996 → Generalizes well
- **Memory Usage**:
  - Train R^2: ~0.997+
  - Test R^2: ~0.42 to 0.60 → Moderate to large gap

Conclusion: Moderate overfitting in memory usage prediction.

HashGen (HG):

- **CPU Usage**:
  - Train and Test R^2: both ~0.998+ → No overfitting
- **Memory Usage**:
  - Train R^2: ~0.99+
  - Test R^2: ~0.55 to 0.64 → noticeable drop

Conclusion: Mild overfitting in memory usage prediction.

RandPw (RP):

- **CPU Usage**:
  - Train R^2: ~0.9996
  - Test R^2: ~0.973–0.975 → Slight drop but not extreme
- **Memory Usage**:
  - Train R^2: ~0.998
  - Test R^2: ~0.992 → Very close

Conclusion: No overfitting in memory or CPU. In fact, this service shows the best generalization performance.

Results With Dropout Layer

GRU Training Results for CPU Usage with Dropout

Service	Test Size	Train RMSE	Train R^2	Test RMSE	Test R^2
Service 1	0.3	0.0026	0.9918	0.0014	0.9972
	0.2	0.0027	0.9902	0.0013	0.9946
	0.1	0.0022	0.9938	0.0007	0.9932
Service 2	0.3	0.001	0.9989	0.0007	0.9991
	0.2	0.0011	0.9984	0.0005	0.9987
	0.1	0.0009	0.9989	0.0006	0.9946
HashGen	0.3	0.0005	0.9978	0.0006	0.9977
	0.2	0.0005	0.9975	0.0006	0.998
	0.1	0.0005	0.9976	0.0007	0.9956
RandPw	0.3	0.0003	0.9748	0.0002	0.9713
	0.2	0.0003	0.9745	0.0002	0.9728
	0.1	0.0003	0.9762	0.0002	0.9703

GRU Training Results for Memory Usage with Dropout

Service	Test Size	Train RMSE	Train R^2	Test RMSE	Test R^2
Service 1	0.3	5,137,750.09	0.7913	5,398,322.99	0.3825
	0.2	6,073,312.34	0.6731	6,701,140.79	0.2597
	0.1	5,481,713.59	0.7104	9,358,503.03	0.1553
Service 2	0.3	183,466.04	0.5861	182,642.93	0.511
	0.2	180,878.08	0.6036	183,950.38	0.4866
	0.1	182,414.01	0.596	172,927.69	0.4198
HashGen	0.3	12,979,348.32	0.5447	16,268,326.33	0.5686
	0.2	11,163,386.74	0.639	17,605,453.15	0.6112
	0.1	10,487,173.81	0.6472	24,564,596.36	0.5354
RandPw	0.3	900,340.76	0.9827	724,905.99	0.9932
	0.2	913,290.73	0.9811	912,496.76	0.9907
	0.1	1,388,980.79	0.9678	864,918.91	0.9272

Service 1

CPU Usage

- R^2 values are consistently very high on both train and test sets (around 0.99+), and RMSE values are very low (around 0.001-0.0027).

- The GRU model predicts CPU usage for Service 1 exceptionally well, with no signs of overfitting. The test results slightly improve as test size decreases (test RMSE decreases), which indicates good generalization.

Memory Usage

- R^2 values are quite low on test sets, dropping as low as 0.15 to 0.38, while train R^2 is moderate (~0.67 to 0.79). RMSE values are very high (in millions).

- The model struggles to predict memory usage accurately. Low R^2 combined with high RMSE on the test set indicates poor fit and generalization. The model seems to be **overfitting**: better performance on training data but poor test performance, especially at lower test sizes. This could be due to noisy or highly variable memory usage patterns, or insufficient feature representation for memory.

Service 2

CPU Usage

- Excellent R^2 values (above 0.99) for both training and test, with very low RMSE (~0.0005 to 0.001).
- The GRU model predicts CPU usage very accurately with no overfitting signs. Performance remains robust across different test sizes.

Memory Usage

- Moderate R^2 (~0.48 to 0.60) and RMSE around 180,000.
- Similar to Service 1, the model struggles with memory prediction but less drastically. The moderate R^2 and stable RMSE on test sets imply some predictability but room for improvement. The difference between train and test R^2 is smaller than Service 1, so less overfitting but still suboptimal fit.

HashGen

CPU Usage

- **Performance:**
  R^2 ~0.997 on both train and test sets with very low RMSE (~0.0005 to 0.0007).

- **Conclusion:**
  Excellent performance predicting CPU usage, no overfitting detected, strong generalization.

Memory Usage

- Moderate R^2 values (~0.54 to 0.64) with very high RMSE (10M to 17M+).

- Similar pattern to other services—poor memory prediction despite moderate R^2. The model shows signs of overfitting (train R^2 higher than test, test RMSE quite large). Memory usage might be more complex or volatile here, requiring alternative modeling strategies or more features.

RandPw

CPU Usage

- R^2 around 0.97 for both train and test, with very low RMSE (~0.0002 to 0.0003).
- Good CPU usage prediction with consistent performance across test sizes and no overfitting.

Memory Usage

- R^2 very high (above 0.92 on test, 0.96+ on train) and RMSE considerably lower than other services (around 700,000 to 1.3M).
- Memory usage prediction is strong here compared to other services, showing good fit and generalization. No clear signs of overfitting.

Overall Insights

- **CPU Usage Prediction:**
  Across all services, GRU models with dropout perform excellently for CPU usage — high R^2, low RMSE, consistent across train and test sets. This suggests CPU patterns are more predictable and GRU is well suited here.
- **Memory Usage Prediction:** Models struggle with memory usage prediction except for RandPw. Lower R^2 with high RMSE and discrepancy between train and test metrics indicate overfitting and complexity in modeling memory usage.
- **Effect of Test Size:**Smaller test sizes tend to increase overfitting risk, especially visible in memory usage predictions where train metrics improve but test metrics degrade. For CPU usage, test size has less impact on performance stability.

Results With Memory Conversion to MB

GRU Results for CPU Usage

Service	Test Size	Train RMSE	Train R^2	Test RMSE	Test R^2
Service 1	0.3	0.0024	0.9929	0.0009	0.9989
	0.2	0.0021	0.9944	0.0007	0.9984
	0.1	0.0018	0.996	0.0007	0.9944
Service 2	0.3	0.0009	0.9991	0.0006	0.9993
	0.2	0.0009	0.999	0.0005	0.9991
	0.1	0.0008	0.9992	0.0004	0.9973
HashGen	0.3	0.0004	0.9984	0.0005	0.9985
	0.2	0.0004	0.9981	0.0004	0.9989
	0.1	0.0004	0.9983	0.0005	0.9975
RandPw	0.3	0.0003	0.9765	0.0002	0.9747
	0.2	0.0003	0.9766	0.0002	0.9759
	0.1	0.0003	0.9764	0.0002	0.9708

GRU Results for Memory Usage

Service	Test Size	Train RMSE	Train R^2	Test RMSE	Test R^2
Service 1	0.3	3.8082	0.8739	4.8404	0.4542
	0.2	3.4562	0.8836	5.7232	0.4062
	0.1	3.3422	0.8816	7.9458	0.3305
Service 2	0.3	0.1712	0.6036	0.1737	0.5135
	0.2	0.1718	0.6068	0.1734	0.4986
	0.1	0.173	0.6005	0.1635	0.4296
HashGen	0.3	10.7178	0.6587	13.8388	0.6568
	0.2	10.424	0.654	16.0258	0.6458
	0.1	9.8931	0.6548	22.623	0.5667
RandPw	0.3	0.8359	0.9836	0.657	0.9939
	0.2	0.8034	0.9839	0.7553	0.993
	0.1	0.8628	0.9863	0.3491	0.987

Service 1

CPU Usage

- The GRU model performed exceptionally well across all test sizes (0.3, 0.2, and 0.1).
- Test RMSE values were very low (≤ 0.0009), and R^2 scores remained consistently high (≥ 0.9944).

- This indicates that the GRU model is highly capable of capturing temporal dependencies in CPU usage data and generalizes well to unseen data.

Conclusion:
The GRU model for CPU usage in Service 1 is highly reliable and production-ready, demonstrating excellent prediction accuracy and stability across varying train-test splits.

Memory Usage

- The GRU model showed moderate to poor performance in predicting memory usage.
- Although training R^2 was ~0.88, the test R^2 scores dropped drastically (0.4542 → 0.3305) as the test size decreased.
- Test RMSE also increased, indicating poor generalization and potential overfitting or insufficient feature representation.

Conclusion:
The GRU model struggles to capture memory usage patterns for Service 1

Service 2

CPU Usage

- GRU models again showed excellent prediction performance with high consistency.
- All test R^2 scores were above 0.997, and RMSE remained minimal (~0.0005 or lower).

Conclusion:
The GRU model for CPU usage in Service 2 performs with outstanding accuracy and robustness. It is well-suited for deployment in real-time or forecasting scenarios.

Memory Usage

- The GRU model produced low R^2 values (~0.5 or less) and relatively constant RMSE, indicating limited learning from the data.
- Training and testing scores are close, suggesting the model is not overfitting, but the feature space may lack key memory usage patterns.

Conclusion:
The model's performance on memory usage is weak and unstable.

HashGen

CPU Usage

- Very consistent and high R^2 scores across all test sizes (0.9975–0.9989), with very low RMSE values (~0.0004–0.0005).

Conclusion:
The GRU model offers high-fidelity CPU usage predictions for HashGen, with great generalization across test sizes. It can be reliably used in practice.

Memory Usage

- R^2 values remained in the 0.55–0.65 range, and test RMSE increased with smaller test sizes (up to 22.6).
- These signs suggest the model has difficulty generalizing for memory usage and may be sensitive to data distribution.

Conclusion:
The model's performance for memory prediction is suboptimal.

RandPw

CPU Usage

- Solid performance with R^2 around 0.975 and extremely low RMSE (~0.0002–0.0003).
- Even though R^2 is slightly lower than in other services, the model is still highly effective.

Conclusion:
The GRU model demonstrates strong predictive accuracy for RandPw's CPU usage and is appropriate for further application or integration into monitoring systems.

Memory Usage

- Excellent results, with training R^2 of 0.9836 and test R^2 as high as 0.9939.
- RMSE values are also low and decrease on the test set.

Conclusion:
Unlike other services, memory usage for RandPw is well-modeled by the GRU, suggesting clear temporal trends and strong learnability.

Service	CPU Overfitting	Memory Overfitting
Service 1	No	Yes
Service 2	No	Mild
HashGen	No	Moderate
RandPw	No	No

Results with Bidirectional Layer

GRU Results for CPU Usage

Service	Test Size	Train RMSE	Train R²	Test RMSE	Test R²
Service 1	0.3	0.0023	0.9935	0.0009	0.9989
	0.2	0.0021	0.9942	0.0007	0.9984
	0.1	0.0021	0.9946	0.0007	0.9942
Service 2	0.3	0.0009	0.999	0.0006	0.9992
	0.2	0.001	0.9986	0.0005	0.9987
	0.1	0.0009	0.9989	0.0005	0.9964
HashGen	0.3	0.0005	0.9973	0.0006	0.9966
	0.2	0.0005	0.9966	0.0005	0.9968
	0.1	0.0005	0.9973	0.0004	0.9955
RandPw	0.3	0.0008	0.9991	0.0007	0.9988
	0.2	0.0008	0.9986	0.0007	0.9985
	0.1	0.0009	0.9985	0.0007	0.9964

GRU Results for Memory Usage

Service	Test Size	Train RMSE	Train R²	Test RMSE	Test R²
Service 1	0.3	3.8598	0.8705	4.7396	0.4767
	0.2	3.3685	0.8894	5.6279	0.4258
	0.1	4.2809	0.8058	8.2267	0.2823
Service 2	0.3	0.1719	0.6006	0.1751	0.5056
	0.2	0.1714	0.6085	0.1749	0.4896
	0.1	0.1736	0.5975	0.1651	0.4183
HashGen	0.3	0.0689	0.7576	0.0817	0.6452
	0.2	0.0707	0.7662	0.0817	0.6556
	0.1	0.071	0.7645	0.0895	0.569
RandPw	0.3	0.053	0.9477	0.0538	0.9207
	0.2	0.0554	0.9332	0.0613	0.8813
	0.1	0.061	0.9138	0.0557	0.8915

Service 1

CPU Usage

- **Performance:**
  - Very low RMSE on both train and test sets (train RMSE ~0.0021, test RMSE ~0.0007).

    - High R^2 values (train R^2 ~0.994 - 0.995, test R^2 ~0.994 - 0.999), indicating excellent fit and generalization.

- **Conclusion:**
    - The model predicts CPU usage extremely well, with no visible signs of overfitting.
    - Performance is consistent across test sizes (0.1 to 0.3), confirming robustness.

Memory Usage

- **Performance:**
    - Relatively high RMSE values (train RMSE around 3.3 to 4.3, test RMSE increasing from ~4.7 to ~8.2).
    - Lower R^2 values, especially on test sets (test R^2 dropping from 0.48 to 0.28), indicating poor generalization.

- **Signs of Overfitting:**
    - Train R^2 (around 0.8 to 0.89) is significantly higher than test R^2 (dropping below 0.5).
    - Test RMSE increases as test size decreases, implying overfitting is more pronounced with less training data.

- **Conclusion:**
    - The GRU struggles to generalize for memory usage in Service 1 and is overfitting the training data.
    - Consider regularization, more training data, or feature engineering to improve memory usage predictions.

Service 2

CPU Usage

- **Performance:**
    - Very low RMSE (~0.0005 to 0.0009) and very high R^2 (~0.998 to 0.999) for both train and test.

- **Conclusion:**
    - Excellent prediction quality with no overfitting. The GRU models CPU usage of Service 2 very accurately and consistently across different test sizes.

Memory Usage

- **Performance:**
    - RMSE around 0.17 on train and test, R^2 around 0.5-0.6 on train, slightly lower on test.

- **Signs of Overfitting:**
    - Train R^2 higher than test R^2 but difference is small (train ~0.6, test ~0.49-0.5).
    - Test RMSE and R^2 are fairly stable, no drastic performance drop.

- **Conclusion:**
    - Some overfitting, but less severe than Service 1 memory usage.
    - Memory predictions could be improved with tuning or more features, but model generalizes moderately well.

HashGen

CPU Usage

- **Performance:**
  - Very low RMSE (~0.0004 to 0.0006) and very high R^2 (~0.997 to 0.998) on train and test.

- **Conclusion:**
  - Strong CPU usage prediction without overfitting.

Memory Usage

- **Performance:**
  - Very high RMSE values (~9.6 to 16.2) and moderate R^2 (~0.55 to 0.67).

- **Signs of Overfitting:**
  - Noticeable difference between train and test RMSE (test RMSE up to 22.8) and a moderate drop in R^2 on test sets.

- **Conclusion:**
  - Memory usage predictions are weak and likely overfitting the training data.
  - Further investigation needed; possibly data quality issues or model complexity mismatch.

RandPw

CPU Usage

- **Performance:**
  - Extremely low RMSE (~0.0002 to 0.0003) and high R^2 (~0.97) on both train and test.

- **Conclusion:**
  - CPU predictions are solid and consistent, no overfitting observed.

Memory Usage

- **Performance:**
  - RMSE ranging from 0.5 to 1.0, with high R^2 values (train R^2 ~0.98, test R^2 ~0.97 to 0.99).

- **Conclusion:**
  - Memory usage modeling is very good, with minimal signs of overfitting.
  - Test results are strong, indicating good generalization.

Results with New Features

GRU Results for CPU Usage

Service	Test Size	Train RMSE	Train R^2	Test RMSE	Test R^2
Service 1	0.3	0.0022	0.9939	0.001	0.9987
	0.2	0.0022	0.9936	0.0009	0.9978
	0.1	0.0018	0.9959	0.0007	0.9938
Service 2	0.3	0.0008	0.9993	0.0006	0.9993
	0.2	0.0008	0.9993	0.0004	0.9992
	0.1	0.0008	0.9992	0.0004	0.9973
HashGen	0.3	0.0005	0.9977	0.0006	0.9979
	0.2	0.0004	0.9983	0.0004	0.9989
	0.1	0.0004	0.9982	0.0005	0.9974
RandPw	0.3	0.0003	0.9747	0.0002	0.9712
	0.2	0.0003	0.9755	0.0002	0.9738
	0.1	0.0003	0.976	0.0002	0.9692

GRU Results for Memory Usage

Service	Test Size	Train RMSE	Train R^2	Test RMSE	Test R^2
Service 1	0.3	4.3666	0.8344	4.7875	0.4664
	0.2	3.9567	0.8475	5.7319	0.4053
	0.1	3.4153	0.8764	8.1086	0.3039
Service 2	0.3	0.1745	0.5881	0.1734	0.5156
	0.2	0.1722	0.6047	0.175	0.4883
	0.1	0.1735	0.598	0.1631	0.4335
HashGen	0.3	10.8748	0.6488	14.6393	0.6078
	0.2	10.8837	0.623	16.9586	0.5939
	0.1	10.235	0.6308	24.3435	0.4859
RandPw	0.3	0.8936	0.9812	0.7478	0.9921
	0.2	0.8245	0.983	0.785	0.9924
	0.1	0.9212	0.9844	0.3735	0.985

Service 1

CPU Usage:

- Excellent performance across all test sizes.
- Test R^2 scores consistently above 0.99, indicating highly accurate predictions.
- RMSE remains very low (≈0.0007–0.0010).

- Model is reliable for CPU usage prediction.

Memory Usage:

- Performance is moderate.
- Train R^2 improves slightly with smaller test sizes (up to 0.87), but test R^2 drops (as low as 0.30).
- RMSE remains high (≈4.7–8.1).
- Model is underperforming for memory usage; possibly due to:
    - Higher variance or noise
    - Incomplete feature representation
    - Need for tuning GRU architecture or features

Service 2

CPU Usage:

- Very high accuracy with R^2 scores ~0.999 consistently.
- RMSE extremely low (≈0.0004–0.0006).
- Model generalizes well for CPU usage.

Memory Usage:

- Test R^2 is low (≈0.43–0.51), though better than Service 1.
- Train R^2 ranges from 0.59–0.60, indicating the model is not fitting well even on training data.
- Needs improvement: consider trying additional features, or a different architecture (e.g., attention mechanisms or CNN+GRU hybrids).

HashGen

CPU Usage:

- Strong generalization with R^2 consistently above 0.997.
- RMSE very low (≈0.0004–0.0006).
- Very reliable for CPU usage prediction.

Memory Usage:

- R^2 fluctuates between 0.48 and 0.65, not very high.
- Test RMSE is high (≈14–24), especially with smaller test sizes.
- Indicates poor generalization and high prediction error; possibly due to:
    - Higher memory usage variability
    - Model underfitting

RandPw

CPU Usage:

- R^2 around 0.97, which is slightly lower than other services but still acceptable.
- RMSE ~0.0002–0.0003, still very low.
- Model performs well for CPU usage.

Memory Usage:

- Outstanding performance.

- Test R^2 consistently above 0.99, with very low RMSE (~0.37–0.78).
- This is your best-performing memory model across all services.