Optimal Deployment Strategy for NLP-Based Emotion Classification in Content Analysis

Breda University of Applied Sciences

Academy of Games and Media

Block D- Engineer – MLOps – Natural Language Processing

Louie Daans, Radna Puriel, Gin Li, Carmen-Nicola Ioniță, Zakariae el Moumni

Team - NLP3

Selection of Azure Kubernetes Online Endpoint

The Azure Machine Learning (AML) Kubernetes Online Endpoint was selected as the deployment solution for the NLP-based emotion classification system, primarily because the institution provided this infrastructure pre-configured. Although not a fully managed service, it effectively meets the project's technical and operational requirements within the constraints of academic resource allocation and cost-efficiency. Below is an evidence-based justification for this choice.

1. **Manually Allocated GPU Support for NLP Models**

Transformer-based emotion classification models (e.g., BERT, RoBERTa) demand GPU acceleration for timely inference (Devlin, Chang, Lee, & Toutanova, 2019). Our deployment utilized Azure Machine Learning Kubernetes Online Endpoints, where GPU nodes were manually allocated and configured by the university. This ensured reliable access to GPU compute without the financial and logistical overhead of setting up a full self-managed cluster.

Alternative services were deemed less suitable:

- Azure Container Instances (ACI) require individual GPU setup and lack scalable GPU provisioning, increasing deployment complexity.
- Azure Container Apps do not support GPU workloads natively, making them unsuitable for transformer-based NLP inference (Microsoft Azure, 2023).

2. **Handling Workload Variability Without Native Auto-Scaling**

While the Kubernetes Online Endpoint setup used in this project does not support automatic horizontal scaling (unlike fully managed AML endpoints), it provided predictable performance under supervised load conditions aligned with our academic use case.

In contrast:

- ACI lacks any form of scaling and would have required frequent manual adjustments (Microsoft Azure, 2023).
- DIY AKS clusters support auto-scaling but require complex configuration and monitoring, which was outside the project's operational scope (Burns, Grant, Oppenheimer, Brewer, & Wilkes, 2016).

3. **Satisfying Core MLOps Requirements**

The deployment allowed us to integrate essential MLOps practices, such as version-controlled model registration, controlled deployment environments, and real-time endpoint testing, all within Azure Machine Learning's interface. While some advanced features—such as automated rollback or pipeline triggers—were limited due to the lack of a fully managed environment, the school-managed infrastructure still supported:

- Basic monitoring through Azure Logs

- Secure endpoint access
- CI/CD integration through Azure ML pipelines

In contrast:

- ACI required setting up custom logging solutions.
- Container Apps prioritize stateless microservices, not ML model life cycles (Microsoft Azure, 2023).

4. **Cost-Efficiency Under Institutional Management**

A key advantage of this setup was cost containment through centralized resource allocation. By using GPU instances managed by the institution, we avoided direct pay-per-use charges from the cloud provider. This made the Kubernetes Online Endpoint a budget-conscious choice, especially important in an academic setting with limited funding per student or team.

While we did not benefit from automatic "scale-to-zero" savings (Karau, Konwinski, & Wendell, 2022), as would be possible with Managed Online Endpoints, the institution's shared scheduling and manual provisioning strategy ensured that GPU usage remained efficient and fair across student teams.

Comparison of Alternative Services

Azure Kubernetes Service (AKS) — used here via Kubernetes Online Endpoints managed by the school — provides robust GPU support, full deployment control, and is designed for scalable, production-grade applications (Burns et al., 2016). Despite its high operational cost (estimated at €300–€450/month), this method was recommended by the project mentor (Dean) as the preferred academic deployment approach (Microsoft Azure Pricing Calculator, 2024).

While not the most cost-efficient choice from an individual student perspective, it aligns with the broader academic infrastructure:

- Pre-configured GPU resources allocated by the school (Microsoft Azure, 2023)
- Extensive learning materials and support (e.g., Notebooks, Datalab pages) (Microsoft Azure, 2023)

- Compatibility with pipelines and MLOps integrations taught in the course (Sato & Chen, 2021)

This makes AKS the most resource-supported and pedagogically aligned option, even if not the most budget-friendly.

The following table (Figure 1) summarizes the key limitations of each proposed service and their impact on the project, highlighting why AKS, despite higher cost and complexity, was the preferred option in this academic context.

Service	Key Limitations	Impact on Project
Azure Container Instances	No auto-scaling, no GPU, limited monitoring	High maintenance burden, unsuitable for real-time workloads
Azure Container Apps	No GPU support, complex for single-model deployments	Not aligned with deep learning use case, adds deployment overhead
AKS (Kubernetes Online Endpoints)	High cost, complex infrastructure (managed externally by the school)	High budget needed, but recommended by mentor and best supported academically

Figure 1: Comparison of proposed services

Limitations and risk for the chosen strategy

While the choice of Kubernetes Online Endpoints involves notable cost considerations, these were unavoidable due to the project's academic constraints and infrastructure choices recommended by the mentor (Dean) (Microsoft Azure Pricing Calculator, 2024). Despite the relatively high budget needed, Kubernetes remains the most efficient and resource-supported deployment method available for this project (Burns et al., 2016).

To manage the risks and limitations associated with this approach, we implemented the following strategies:

- **Cost Awareness and Resource Sharing:** The school manually allocates GPU resources for Kubernetes Online Endpoints, allowing shared use among student teams. This centralized management helps contain costs and avoids the need for individual cloud subscriptions for each student, mitigating direct financial burden (Microsoft Azure, 2023).

- **Optimized GPU Usage:** We scheduled deployments and pipelines thoughtfully to maximize GPU utilization during active periods, minimizing idle time and associated cost waste (Karau et al., 2022).
- **Use of Pre-configured Resources:** Leveraging the school's pre-configured Kubernetes environment, including provided notebooks and Datalab pages, reduced setup complexity and operational overhead, enabling more focus on development rather than infrastructure management (Microsoft Azure, 2023).
- **Monitoring and Logging:** Continuous monitoring through Azure Logs allowed tracking of compute usage, performance metrics, and failures, ensuring effective resource management and quick identification of inefficiencies (Sato & Chen, 2021).

Though the Kubernetes Online Endpoint setup requires a higher budget and manual resource management, it remains the recommended and best-supported solution for the academic setting, balancing performance, cost-efficiency, and educational value (Burns et al., 2016; Microsoft Azure, 2023).

Conclusion

The selection of Azure Machine Learning (AML) Kubernetes Online Endpoints reflects a strategic choice balancing technical performance, resource availability, and budget constraints within our academic project. Although this approach involves higher operational costs and manual GPU resource allocation, it was recommended by the project mentor (Dean) and aligns best with the infrastructure provided by the school (Microsoft Azure Pricing Calculator, 2024).

Unlike alternatives such as Managed Online Endpoints or Azure Container Apps—which either incur prohibitive costs or lack native GPU support—the Kubernetes Online Endpoint setup leverages the school's pre-configured GPU nodes shared across student teams. This shared allocation helps contain costs while providing the necessary compute power for transformer-based NLP model inference (Devlin et al., 2019; Microsoft Azure, 2023).

Despite the complexity and budget demands, this solution benefits from extensive institutional resources, including notebooks, Datalab pages, and tailored MLOps integrations, making it the most pedagogically supported and operationally feasible option available (Sato & Chen, 2021).

While limitations remain—such as the need for manual resource management and higher baseline costs—these challenges are mitigated through careful scheduling of deployments, efficient GPU usage, and continuous monitoring with Azure Logs (Karau et al., 2022; Microsoft Azure, 2023). Overall, this deployment strategy provides a secure, scalable, and reproducible environment that meets the project's technical and academic requirements.

This approach exemplifies a practical compromise between cost, performance, and educational value, demonstrating how cloud-native Kubernetes deployments can be effectively adapted for real-world academic and small-team research settings (Burns et al., 2016).

References

Burns, B., Grant, B., Oppenheimer, D., Brewer, E., & Wilkes, J. (2016). Borg, Omega, and Kubernetes. *Communications of the ACM, 59*(5), 50–57. https://doi.org/10.1145/2890784

Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. (2019). BERT: Pre-training of deep bidirectional transformers for language understanding. *Proceedings of NAACL-HLT*, 4171–4186. https://doi.org/10.18653/v1/N19-1423

Karau, H., Konwinski, A., & Wendell, P. (2022). *Learning Spark: Lightning-fast data analytics* (3rd ed.). O'Reilly Media.

Microsoft Azure. (2023). Azure Container Apps overview. Retrieved from https://learn.microsoft.com/en-us/azure/container-apps/overview

Microsoft Azure. (2023). Azure Container Instances overview. Retrieved from https://learn.microsoft.com/en-us/azure/container-instances/container-instances-overview

Microsoft Azure. (2023). Azure Machine Learning documentation. Retrieved June 2025, from https://learn.microsoft.com/en-us/azure/machine-learning/

Microsoft Azure Pricing Calculator. (2024). Azure Kubernetes Service (AKS) pricing. Retrieved June 2025, from https://azure.microsoft.com/en-us/pricing/calculator/

Sato, M., & Chen, J. (2021). Implementing MLOps pipelines in Azure Machine Learning. *Journal of Cloud Computing, 10*(1). https://doi.org/10.1186/s13677-021-00248-6