GOETHE UNIVERSITÄT
FRANKFURT AM MAIN

Institut für Informatik, Goethe-Universität Frankfurt am Main

Seminar Künstliche Intelligenz
Sommersemester 2025

Leitung:
Prof. Dr. Manfred Schmidt-Schauß

Large Language Model Guided Knowledge Distillation for Time Series Anomaly Detection*

Roman Christof
Master Informatik
2. Fachsemester

Frankfurt am Main
9. Juni 2025

*Autoren: Chen Liu, Shibo He , Qihang Zhou , Shizhong Li and Wenchao Meng

Inhaltsverzeichnis

1 Einleitung

Das Paper *"Large Language Model Guided Knowledge Distillation for Time Series Anomaly Detection"* [5] beschäftigt sich mit der Erkennung von Anomalien in Zeitreihen. Eine Zeitreihe ist ein geordnete Folge reellwertiger Beobachtungen, die zu festegelegten Zeitpunkten erhoben werden [1]. Im Fall von *einer* beobachteten Variablen spricht man von einer *univariaten* Zeitreihe und im Fall mehrerer Variablen von *multivariaten* Zeitreihen. Eine multivariate Zeitreihe sind zum Beispiel die Vitaldaten eines Patienten, bei denen gleichzeitig Parameter wie Herzfrequenz, Blutdruck, Atemfrequenz und Sauerstoffsättigung erfasst werden. Solche Zeitreihen können nun *Anomalien* aufweisen. Allgemein wird eine Anomalie gefasst als eine Beobachtung, die stark von den übrigen Daten abweicht und damit die Vermutung nahelegt von einem anderen Mechanismus erzeugt worden zu sein [4], also eine Abweichung von der allgemeinen Verteilung der Daten vorliegt.[1] Wichtig ist dabei die Unterscheidung zwischen bloßem Rauschen bzw. Messfehlern und *interessanten*, potenziell problematischen Abweichungen. Allerdings ist die konkrete Defintion, ab wann eine Abweichung eine Anomalie ist, nicht ganz trivial. Ob bspw. ein erhöhter Herzschlag normal oder anormal ist, hängt vom Aktivitätszustand der Person ab. Auch saisonale Effekte oder sich verändernde Nutzergewohnheiten können das Normalverhalten verschieben und erschweren so die Erkennung. Die nicht-stationäre Eigenschaft mancher Zetireihen kann die Bestimmung von Anomalien also erschweren. In Fällen, in denen die Datenverteilung bekannt ist, lassen sich Anomalien mit einfachen Schwellenwerten identifizieren – etwa durch das Drei-Sigma-Kriterium, bei dem Beobachtungen außerhalb von drei Standardabweichungen vom Mittelwert als verdächtig gelten [2]. In der Praxis ist die zugrunde liegende Verteilung jedoch meist unbekannt, was die Entwicklung von Modellen notwendig macht, die diese Strukturen selbstständig lernen. Ein zentrales Problem bei der Entwicklung lernbasierter Verfahren zur Anomalieerkennung ist allerdings der Mangel an Trainingsdaten und insbesondere annotierter Trainingsdaten. Anomalien sind selten, ihre Kennzeichnung erfordert oft Expertenwissen, und die manuelle Annotation ist kosten- und zeitintensiv. Solche Szenarien - geringe Menge an Daten und Annotationen - adressieren die Autoren des Papers mit ihrem Ansatz *AnomalyLLM*. Sie gehen der Frage nach, wie in diesem Fall robuste Repräsentationen von Zeitreihen gelernt werden können, um Anomalien zu erkennen. Dafür kombinieren sie ein für Zeitreihen adaptiertes

[1] Die Autoren gehen nicht näher auf verschiedene Arten von Anomalien ein. Es gibt unterschiedliche Taxonomien, die sich nach ihrer Perspektive unterscheiden. Zum Beispiel mit einem Fokus auf die zu erkennenden Anomalien: *point outliers*, *subsequence outliers* und *outlier time series*. [1]

großes Sprachmodell (Large Language Model - LLM) mit einem Wissensdestillation (knowledge distillation) Ansatz. Dabei fungiert ein LLM-basiertes Teacher-Modell als Referenz für normale Datenmuster. Ein Student-Modell wird darauf trainiert, dessen Ausgaben für normale Zeitreihen zu imitieren, jedoch nicht für anomale. Eine Abweichung der Repräsentation beider Modelle wird dann als potenzielle Anomalie gewertet. Um die Repräsentationen gezielt zu steuern und eine Überanpassung zu vermeiden, integrieren die Autoren spezielle Regularisierungstechniken, darunter sogenannte Prototypen und synthetisch erzeugte Anomalien.
In der vorliegenden Zusammenfassung werden zuerst in Abschnitt 2 verwandte Arbeiten vorgestellt, die für die Einordnung des vorliegend Papers relevant sind. Dann werden in Abschnitt 3 die wesentlichen Momente des Ansatzes von AnomalyLLM vorgestellt. Der anschließende Abschnitt 4 stellt die Evaluierung und Resultate der Methode vor, um dann zum Schluss in Abschnitt 5 die Stärken und Schwächen des Ansatzes aufzuzeigen, mögliche zukünfige Forschungsarbeiten und die Anwendbarkeit zu diskutieren.

2 Grundlagen und relevante Vorarbeiten

Die Anomalieerkennung in Zeitreihen wurde bislang vor allem durch *unsupervised* Verfahren dominiert, da annotierte Daten, wie oben beschrieben, oft fehlen. Diese Methoden lernen die Verteilung der Daten ohne Labels und identifizieren Abweichungen vom gelernten Muster. Zwei zentrale Ansätze sind dabei One-Class-Classification-Methoden, bei denen ein Modell ausschließlich auf normalen Daten trainiert wird und dichtebasierte Verfahren, die die Wahrscheinlichkeitsverteilung der Daten schätzen und seltene Ausprägungen als Anomalien werten. Eine Weiterentwicklung stellen *self-supervised* Verfahren dar. Diese generieren Pseudo-Labels aus den Daten selbst, indem sie sogenannte Pretext-Tasks nutzen – wie etwa Rekonstruktion, Vorhersage oder Imputation. Dadurch lässt sich eine Repräsentation der Daten lernen, ohne auf explizite Annotationsdaten angewiesen zu sein.
Das vorliegende Paper mit AnomalyLLM knüpft darüber hinaus an mehrere zentrale Vorarbeiten an. Zhou et al. [10] zeigen, dass vortrainierte LLMs universelle Repräsentationen erzeugen können – auch für Zeitreihen, da sie allgemeine Strukturmuster modellieren, die nicht auf Sprache beschränkt sind. Mit ihrem Ansatz GPT4TS wurde ein LLM erstmals direkt zur Anomalieerkennung in Zeitreihen verwendet. Allerdings führte die starke Generalisierungsfähigkeit des GPT2-Modells dazu, dass auch anomale Daten rekonstruiert wurden, was zu vielen False Negatives führte [10]. Die Autoren versuchen dieses Problem zu lösen, indem sie mehrere bereits entwickelte Methoden der

Wissensdistillierung und Regularisierungstechniken miteinander kombinieren und für die Domaine der Zeitreihen adaptieren. Zentral ist hierbei die sog. *differentielle Wissensdistillation*, die zuvor für die Bildverarbeitung entworfen wurde [9]. Bei der Wissensdistillation lernt ein kleineres Modell (Student) von einem leistungsfähigeren Modell (Teacher). Im Rahmen der *differentiellen* Wissensdistillation lernt das Student-Netzwerk nun nur normale Muster, nicht aber die Repräsentation anomaler Daten. Um auf der einen Seite das Lernen normaler Zeitreihenmuster durch das Student-Netzwerk zu unterstützen, bauen die Autoren zudem auf dem Konzept der Prototypen auf [6], die typische Muster der gegebenen Zeitreihe repräsentieren. Auf der anderen Seite knüpfen sie an Arbeiten der synthetischen Datengenerierung an [7], um synthetische Anomalien zu erzeugen und damit das Teacher-Netwerk zu trainieren. Mit der Zusammenführung dieser Methoden wollen die Autoren die Forschungslücke im Bereich der zeitreihenbasierten Anomalieerkennung im Kontext geringer Verfügbarkeit (gelabelter) Trainingsdaten schließen. In solchen Szenarien kommen bestehende unsupervised und self-supervised Methoden an ihre Grenzen. Durch den speziellen Ansatz zur Wissendistillation führen sie die Forschung zur Anwendbarkeit von LLMs für diese Domaine weiter und stellen einen Ansatz vor, der bisherige Probleme wie die der zu starken Generalisierung von GPT4TS lösen will.

3 Methoden

Im Folgenden werden nun die Kernstücke der Methode detaillierter vorgestellt. Dazu zählen das allgemeine Modell der Wissensdistillation, die Einbindung von Prototypen und synthetisch erzeugten Anomalien sowie der spezifische Trainingsansatz zur Optimierung des Netzwerks.

3.1 Wissensdistillation zur Anomalieerkennung

Für die Anomalieerkennung wird die Zeitreihe zunächst in mehrere feste Zeitfenster unterteilt, die jeweils einzeln analysiert werden. Ziel ist es, für jedes dieser Zeitfenster einen Anomaliewert zu berechnen. Das zugrundeliegende Setting basiert auf einem Wissensdistillierungsansatz. Ein Teacher-Netzwerk, das auf einem vortrainierten LLM basiert, wird um eine Eingabeschicht erweitert, um Zeitreihendaten verarbeiten zu können. Dieses Netzwerk erzeugt für jedes Zeitfenster eine Repräsentation, die als Referenz dient. Parallel dazu verarbeitet ein Student-Netzwerk dasselbe Zeitfenster und wird darauf trainiert, die Repräsentation des Teachers nur für normale Zeitfenster nachzubilden. Der Anomaliewert für ein Zeitfenster ergibt sich aus der Diskrepanz

zwischen den Repräsentationen von Teacher- und Student-Netzwerk. Es wird damit angestrebt für Anomalien ein hohe Differenz zwischen Student und Teacher zu erzeugen. Anomalien werden also nicht direkt klassifiziert, sondern über eine abweichende Repräsentation identifiziert.

3.2 Student-Netzwerk mit Prototypen

Um zu vermeiden, dass das Student-Netzwerk zu generische Repräsentationen lernt, die auch anomale Muster einschließen, werden Prototypen integriert. Diese repräsentieren typische, aus den Trainingsdaten gelernte Zeitfenstersegmente. Dafür wird ein Pool von Prototypen initialisiert. Während des Trainings wird das ähnlichste Prototypensegment über Kosinus-Ähnlichkeit bestimmt. Nach Normalisierung und Patching werden Eingabe und Prototyp jeweils durch einen linearen Layer in Embeddings überführt. Die Integration erfolgt über einen erweiterten Transformer-Encoder, bei dem die Attention-Layer zusätzlich auf die Prototypen zugreifen. Es werden zwei Attention-Werte berechnet - normale Attention und Prototype Attention - die dann kombiniert werden.

3.3 Teacher-Netzwerk und Data Augmentation

Als Teacher-Netzwerk verwenden die Autoren ein auf Textdaten vortrainiertes GPT2-Modell. Um Zeitreihendaten verarbeiten zu können, werden die ursprünglichen Zeitreiehen auch hier zunächst normalisiert, in Patches unterteilt und über eine lineare Embedding-Schicht in ein geeignete Repräsentation überführt. Die vortrainierte Kernarchitektur des GPT2 – insbesondere die Attention-Layer und das Feedforward-Modul – bleibt unverändert. Lediglich das Positions-Embedding sowie die Layer-Normalisierung werden an die Charakteristika von Zeitreihendaten angepasst, d.h. feinabgestimmt. Da wie bereits erwähnt gelabelte Daten für Anomalien rar sind, der Teacher aber auch anomale Zeitreihen lernen soll, nutzen die Autoren einen Ansatz zur Generierung synthetischer Anomalien. Dafür werden zufällig Teile eines Zeitfensters gewählt und mit Jittering, Scaling und Warping modifiziert.[2]

[2]Beim Jittering wird einem Zeitreihensignal zufälliges Rauschen hinzugefügt. Für das Scaling wird das gesamte Zeitfenster mit einem zufälligen Skalierungsfaktor multipliziert und das Warping verändert die zeitliche Struktur der Zeitreihe, indem die zeitliche Abfolge lokal gestaucht oder gedehnt wird.

3.4 Training des Teacher- und Student-Netzwerks

Für das Training des Teacher- und Student-Netzwerks werden nun sowohl die Originaldaten als auch die synthetisch generierten Anomalien genutzt. Die Repräsentationspaare der ursprünglichen Zeitfenster werden durch (z_i, c_i) bezeichnet, wobei z_i den vom Student-Netzwerk generierten Feature-Vektor für das i-te Zeitfenster darstellt und resp. c_i die entsprechende Repräsentation des Teacher-Netzwerks. Die hochgestellte Notation z_i^a bzw. c_i^a kennzeichnet die Repräsentationen für die synthetisch augmentierte Zeitfenster. Ziel des Trainings ist es, die Repräsentationen normaler Zeitfenster in beiden Netzwerken anzugleichen, während gleichzeitig die Repräsentationen synthetischer Anomalien voneinander entfernt werden. Dies wird durch die folgende Verlustfunktion umgesetzt:

$$\mathcal{L}_{kd} = \frac{1}{N}\sum_{i=1}^{N} \|z_i - c_i\|_2^2 - \log\left(1 - \exp\left(-\|z_i^a - c_i^a\|_2^2\right)\right) \tag{1}$$

Dabei ist N die Anzahl der Trainingsbeispiele. Kombiert wird diese Verlustfunktion mit einer weiteren sog. kontrastive Verlustfunktion für das Teacher-Netwerk, die darauf fokusiert ist die Repräsentationen des Originals und der synthetischen Variante möglichst ähnlich zu halten:

$$\mathcal{L}_{ce} = \frac{1}{N}\sum_{i=1}^{N} -\frac{c_i}{\|c_i\|_2} \cdot \frac{c_i^a}{\|c_i^a\|_2} \tag{2}$$

Die vollständige Verlustfunktion setzt sich schließlich als gewichtete Summe beider Komponenten zusammen: $\mathcal{L}_{total} = \mathcal{L}_{kd} + \lambda\mathcal{L}_{ce}$, wobei mit λ der Einfluss der kontrastiven Komponente kontrolliert wird.

4 Ergebnisse

Die Autoren testen AnomalyLLM auf 15 realen Datensätzen. Davon sind neun univariate Zeitreihen aus dem UCR Anomaly Archive [8] und sechs multivariate Zeitreihen aus etablierten Benchmarks (u.a. SMD, MSL, SMAP, PSM, WaQ, SWAN). Als Vergleichsmodelle bzw -ansätze werden klassische One-Class-Verfahren , dichtebasierte Modelle sowie self-supervised Ansätze und LLM-basierte Verfahren wie GPT4TS verwendet. Als Metriken dienen Recall, Precision, F1-Score und Accuracy.[3] AnomalyLLM erzielt in fast allen

[3]Genauer, nutzen sie eine angepasste Variante the F1-Scores, den sog. *affiliated F1*, der Anomalien nicht punktweise, sondern auf Ereignisebene zählt.

Domänen den höchsten F1-Score bzw. Accuracy. Auf den UCR-Datensätzen verbessert es deutlich den durchschnittlichen F1-Score um 10% gegenüber dem jeweils zweitbesten Verfahren. AnomalyLLM kann die Ergebnisse besonders verbessern in Szenarien mit wenigen oder gar keinen Trainingsbeispielen. In einem Vergleich auf dem PowerDemand-Datensatz von UCR bleibt erzielt das Modell auch bei nur 10% der Trainingsdaten gute Ergebnisse (mit einem F1-Score 78%), während die andere Methoden deutlich schlechter abschneiden (unter ca. 47%).

5 Diskussion

Insbsondere die zuletzt erwähnten Ergebnisse unter Bedingungen begrenzter Trainingsdaten zeigen, dass der von den Autoren entwickelte Ansatz zur Wissensdistillation mittels LLMs insbesondere unter Bedingungen begrenzter Datenverfügbarkeit deutliche Vorteile gegenüber bisherigen unsupervised Methoden bietet. Die Autoren zeigen damit auch, dass ihr Ansatz vorherige Probleme der übermäßigen Generaliersierung (wie bei GPT4TS) lösen kann. Vor allem Anwendungen in Bereichen, in denen Daten generell als auch speziell gelabelte Anomalien nur eingeschränkt zugänglich sind, könnten besonders von diesem Ansatz profitieren. Ein solcher Fall wäre möglicherweise die medizinische Überwachung von Patienten. U.a. in diesem Bereich ist es wegen der inhärente physiologische Variabilität zwischen Individuen und der großen Spanne dessen, was als "normal"gilt, sehr schwierig Anomalien zu identifizieren. [3] Die *Zero-* bzw. *Few-Shot*-Fähigkeiten von AnomalyLLM könnten hier hilfreich sein. AnomalyLLM würde hier potenziell die Möglichkeit bieten mit wenigen Beispielen ein Vorhersagen zu treffen.
Das Hauptproblem des vorgestellten Ansatzes hingegen scheint die Notwendigkeit des hohen Rechenaufwands und die lange Inferenzzeit, bedingt durch das LLM, zu sein. Eine enstprechende ausführliche Analyse fehlt allerdings in dem Paper. Besonders interessant wäre der Vergleich hinsichtlich dieser Dimension mit herkömmlichen unsupervised Methoden. Zukünftige Arbeiten könnten deshalb erkunden, wie das Teacher-Modell effizienter gestaltet werden könnte, bspw. durch Modellkompression. Eine solche Weiterentwicklung wäre entscheidend für den praktischen Einsatz in ressourcenbeschränkten oder Echtzeitumgebungen. Hinsichtlich der Generierung synthetischer Anomalien könnte zudem neben den klassischen Verfahren (Warping, Jittering und Scaling), die nur begrenzt die Komplexität realer Fehlermuster abbilden, Strategien verwendet werden, die Anomalien realitätsnäher abbilden. Beispielsweise mittels generativer Modelle. Darüber hinaus ist nicht klar, wie gut der Ansatz hinsichtlicht unterschiedlicher Anomaliearten (bspw. Punkt-,

Kontext- oder Subsequenzanomalien) funktioniert, da dies im Paper nicht explizit differenziert oder evaluiert werden. Eine solche Aufschlüsselung nach Anomalytypen könnte auch zukünftige Arbeit ergänzen und möglicherweise lassen sich durch eine solche Fehleranalyse auch Verbesserungen der Architektur ableiten.

Literatur

[1] Blázquez-García, A., Conde, A., Mori, U., and Lozano, J. A. A review on outlier/anomaly detection in time series data. *ACM Comput. Surv. 54*, 3 (Apr. 2021).

[2] Boniol, P., Liu, Q., Huang, M., Palpanas, T., and Paparrizos, J. Dive into time-series anomaly detection: A decade review, 2024.

[3] Fernando, T., Gammulle, H., Denman, S., Sridharan, S., and Fookes, C. Deep learning for medical anomaly detection – a survey. *ACM Comput. Surv. 54*, 7 (July 2021).

[4] Hawkins, D. M. *Identification of Outliers*, 1 ed. Monographs on Statistics and Applied Probability. Springer Dordrecht, 1980.

[5] Liu, C., He, S., Zhou, Q., Li, S., and Meng, W. Large language model guided knowledge distillation for time series anomaly detection. In *Proceedings of the Thirty-Third International Joint Conference on Artificial Intelligence* (2024), IJCAI '24.

[6] Song, J., Kim, K., Oh, J., and Cho, S. Memto: Memory-guided transformer for multivariate time series anomaly detection. In *Advances in Neural Information Processing Systems* (2023), A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine, Eds., vol. 36, Curran Associates, Inc., pp. 57947–57963.

[7] Sun, Y., Pang, G., Ye, G., Chen, T., Hu, X., and Yin, H. Unraveling the 'anomaly' in time series anomaly detection: A self-supervised tri-domain solution. In *2024 IEEE 40th International Conference on Data Engineering (ICDE)* (2024), pp. 981–994.

[8] Wu, R., and Keogh, E. J. Current time series anomaly detection benchmarks are flawed and are creating the illusion of progress (extended abstract). In *2022 IEEE 38th International Conference on Data Engineering (ICDE)* (2022), pp. 1479–1480.

[9] Zhou, Q., He, S., Liu, H., Chen, T., and Chen, J. Pull push: Leveraging differential knowledge distillation for efficient unsupervised anomaly detection and localization. *IEEE Trans. Cir. and Sys. for Video Technol. 33*, 5 (May 2023), 2176–2189.

[10] Zhou, T., Niu, P., Wang, X., Sun, L., and Jin, R. One fits all: power general time series analysis by pretrained lm. In *Proceedings of the 37th International Conference on Neural Information Processing Systems* (Red Hook, NY, USA, 2023), NIPS '23, Curran Associates Inc.