

パッチベースTransformer及び 時間遅延埋め込みによる 無線リンク品質推定に関する検討

An Investigation into Wireless Link Quality Estimation
Using Transformers

埼玉大学 大学院理工学研究科

高城 友也

渡部 康平

研究背景

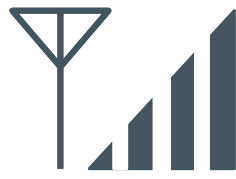
- 5G/IoTの普及により，スマートフォンだけでなく**自動車・医療機器など多様な機器**が無線通信を利用。
- 無線リンクの品質は伝搬環境・干渉・送受信機特性などにより時間的・空間的に変動
- 将来の品質低下を事前に把握できれば，品質低下前にハンドオーバや通信モード切替えを開始できる



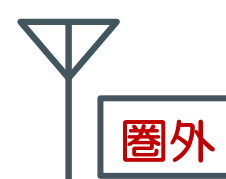
無線リンク品質推定（**WLQE**: Wireless Link Quality Estimation）の高度化が求められている



都心部

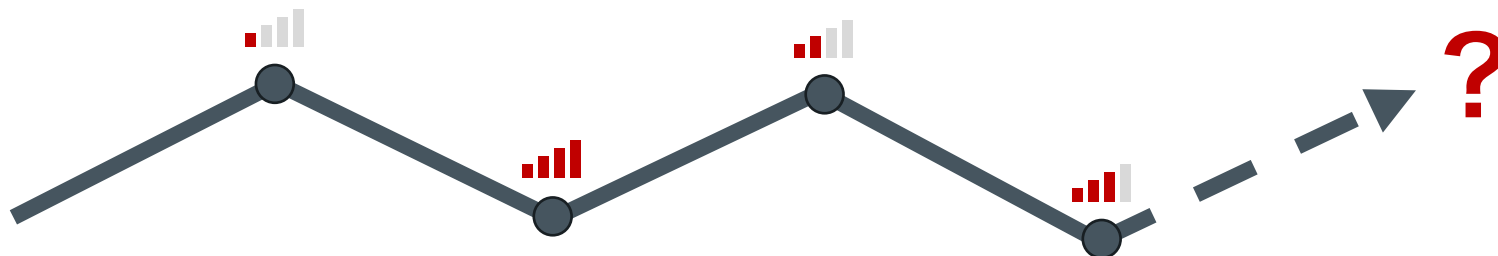


地方山間部・トンネル



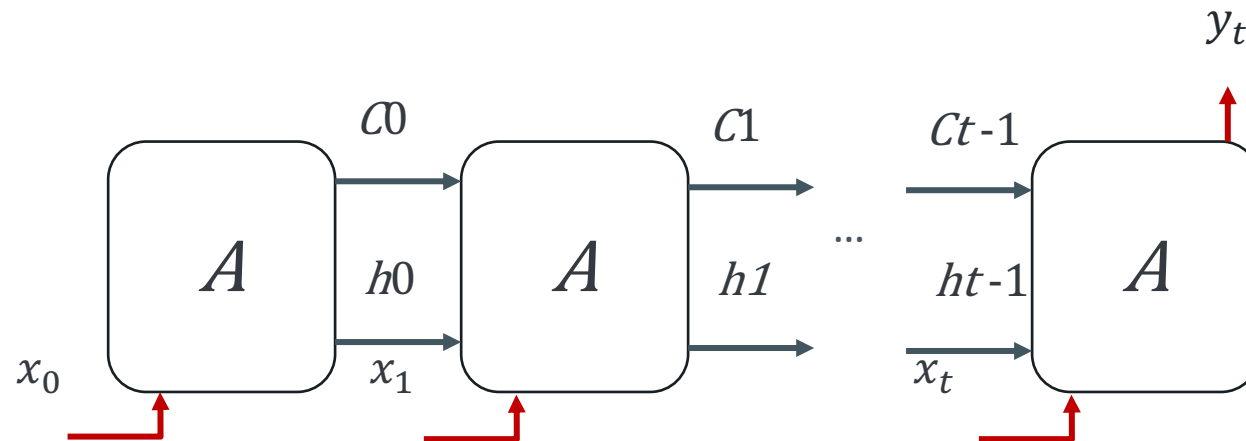
研究目的

- 本研究は**時系列データ**を活用して，未来の通信品質を高精度に推定する
- LSTMを比較基準とし，リンク品質グレードの分類ではなく，**将来複数時点のRSRPを直接出力する回帰問題**を扱う
- Irishデータセットを用い，LSTM・MLP・AE・Vanilla Transformer・TDE Transformer・PatchTST を同一のデータ分割・評価指標で比較
- 移動シナリオおよび予測先に応じて，**各Transformerベース手法のRMSEがどのように変化するかを明らかにする**



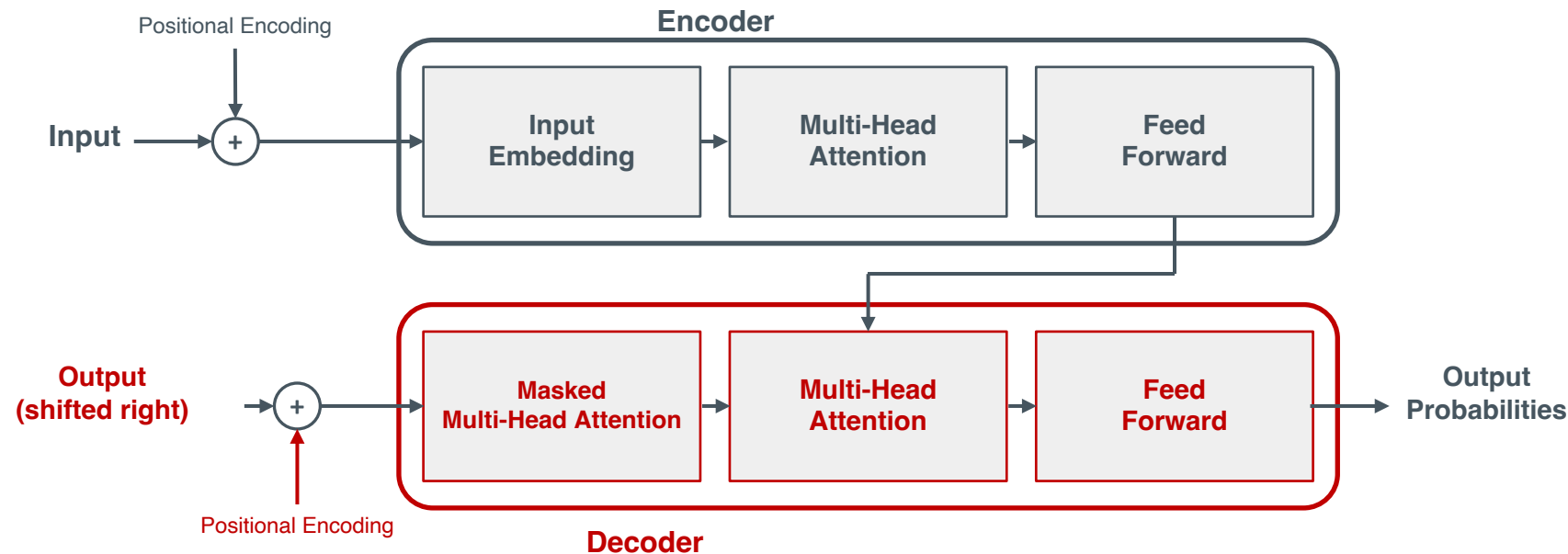
課題：LSTMベース手法の計算上の制約

- 既存のWLQE研究では、**LSTMベース**手法が高い予測精度を示している
- しかし LSTM は系列方向に**逐次処理を行う**ため並列化が難しい
- 入力する観測履歴が長くなるほど**計算時間が増大する**
→**系列全体を並列に処理できる自己注意機構を検討**



Transformer

- Transformerは、従来のRNNや畳み込み（Conv）を排除し、Attention機構を用いたネットワークアーキテクチャである
- Positional Encodingで時系列情報を付与し、Multi-Head Attentionでシーケンス内の依存関係を捉える
- 本研究では、このアーキテクチャをVanilla Transformerと呼ぶ



関連研究：WLQEに関する既存研究

WLQEに関する既存研究

- Millanら[6]：無線コミュニティネットワークで収集されたリンク品質の時系列に複数の予測手法を適用し，遠い予測先ほど精度が低下することと，リンクごとの予測難易度の違いを示した
- Boucettaら[5]：LSTMを用いてRSSIとPDRを予測し，その予測値を5段階のリンク品質グレードへ変換する手法を提案
- Kantoら[2]：実環境で収集された通信品質の時系列を用いたLSTMベースWLQEが，時系列情報を用いない深層学習手法より高い分類精度を達成したことを報告

関連研究：時系列予測分野のTransformer

一般的な時系列予測分野におけるTransformer

- Wuら[4]：インフルエンザ流行データを対象にEncoder-Decoder型Transformerを適用し、観測値から背後の状態空間を表現する時間遅延埋め込みを検討
- Nieら（PatchTST）[3]：長期時系列予測を対象に、連続する観測値を一つのパッチにまとめてトークン数を削減し、より長い観測履歴を利用可能に

→ いずれもWLQE向けではなく、**急峻な変動**を含む移動体RSRP予測への適用可能性は**未検証**

問題設定：RSRPの複数ステップ予測

過去 h ステップの通信品質を入力とし，将来 p ステップの RSRP(Reference Signal Received Power)を予測する回帰問題

入力窓: $X_{t-h+1:t} = [x_{t-h+1}, \dots, x_t]^\top \in \mathbb{R}^{h \times n}$

正解窓: $y_{t+1:t+p} = [y_{t+1}, y_{t+2}, \dots, y_{t+p}]^\top \in \mathbb{R}^p$

予測: $\hat{y}_{t+1:t+p} = f_\theta(X_{t-h+1:t})$

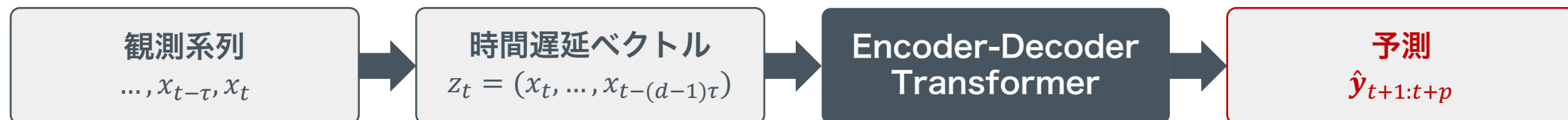
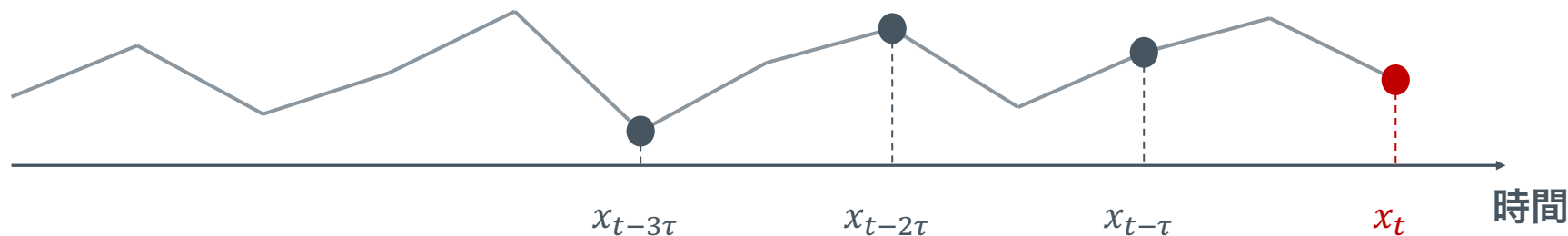
損失: $\mathcal{L}(\theta) = \frac{1}{Np} \sum_{i=1}^N \sum_{k=1}^p \left(\hat{y}_k^{(i)} - y_k^{(i)} \right)^2$

x_t : 時刻 t に観測される特徴量 (通信メトリクスなど)
 y_t : 時刻 t における予測対象



Time Delay Embedding

- RSRP観測系列から時間遅延座標を構成し，Encoder-Decoder型Transformerへ入力
- $TDE_{d,\tau}(x_t) = (x_t, x_{t-\tau}, \dots, x_{t-(d-1)\tau})$ (埋め込み次元 d ，遅延 τ)
- Takensの埋め込み定理に着想[9]：観測値の遅延ベクトルで背後の状態空間を表現



PatchTST

- 時系列を固定長の部分系列（パッチ）に分割し，各パッチをトークンとして扱う
- パッチ長 w ，ストライド s : $s < w$ で隣接パッチが重複し，局所変化を複数トークンで表現
- 点単位トークン化に比べトークン数を削減
→ 長い観測履歴を効率的に処理
- RevIN (Reversible Instance Normalization) で入力窓ごとの分布変化に対応
- Flatten Head で全パッチ表現を連結し， p ステップを一括出力 (Decoderなし)



Irish Dataset

- アイルランドの商用LTE網で収集された通信品質・端末コンテキストの時系列データ[10]
- バス／自動車／歩行者／静止／列車 の5シナリオ，計135系列，約1秒間隔でRSRPなどの通信メトリクス他，位置情報等を記録
- 各シナリオを独立データセットとして扱い，RSRPのみを入力する単変量予測を実施

シナリオ	全体	学習	検証	テスト
バス	16	11	3	2
自動車	53	37	10	6
歩行者	31	21	6	4
静止	15	10	3	2
列車	20	14	4	2
合計	135	93	26	16

実験条件：基本設定

- 比較手法：LSTM・MLP・AE・Vanilla Transformer・TDE Transformer・PatchTST の6手法
- 予測ステップ数 $p=30$ ，入力履歴長 $h=450$
- データ分割シード42，モデルシード42-46（5回学習し平均・標準偏差を評価）
- 評価指標：標準化RSRPに対するRMSE

実験条件：各手法のハイパーパラメータ

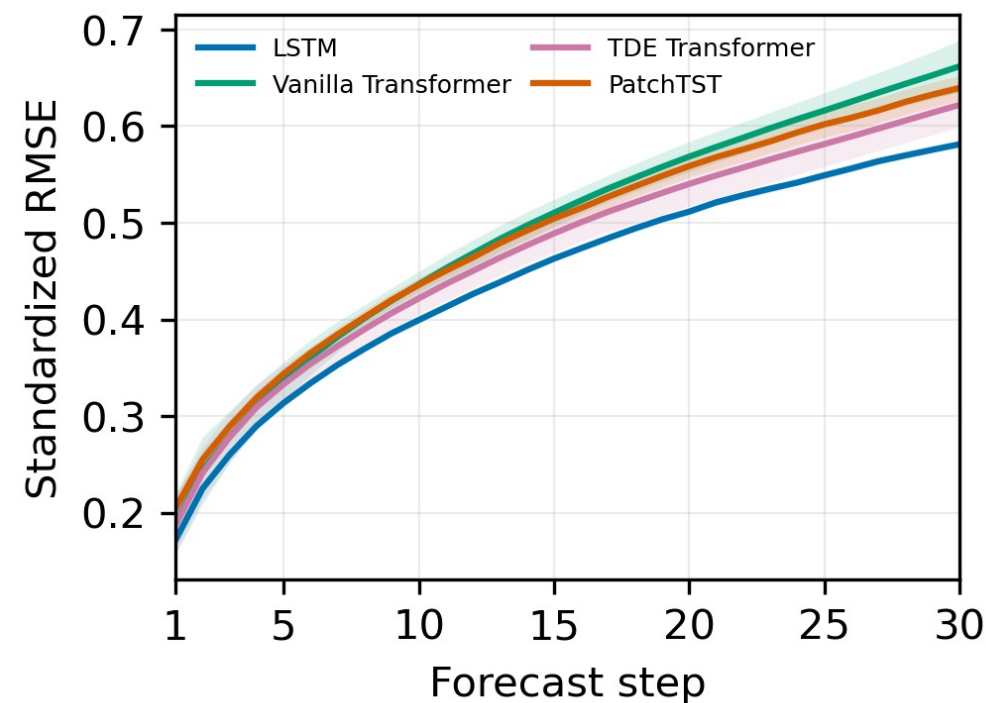
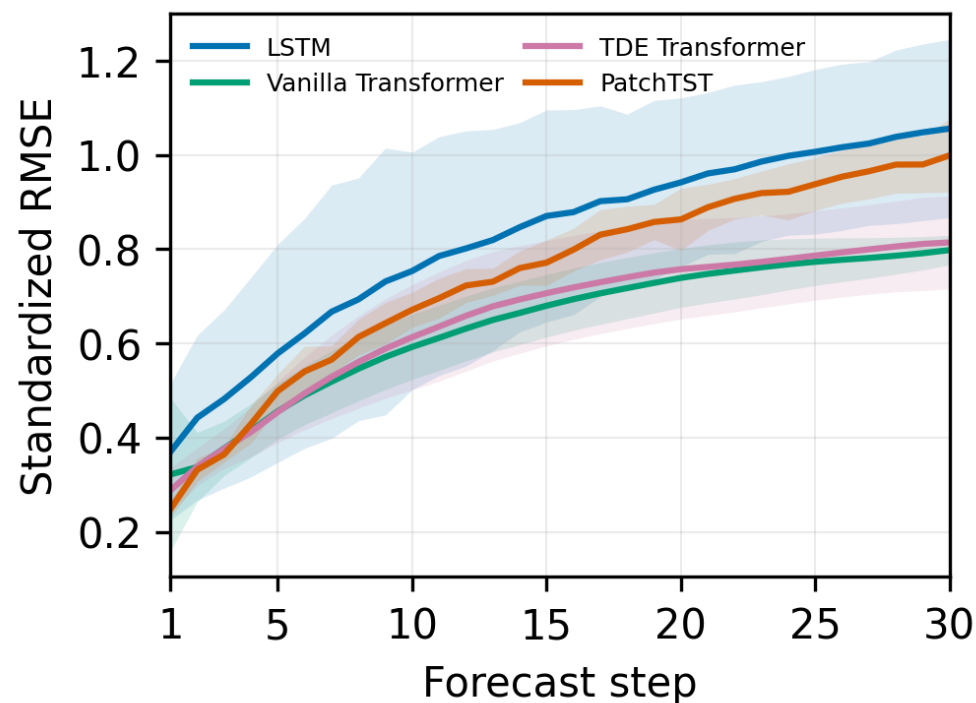
手法	主要パラメータ設定	補足
LSTM	層数4, 隠れ次元128, 学習率 $1e-4$ 等	先行研究(Kanto2024)の設定を参考
TDE Transformer	埋め込み次元 d (候補 $\{1,2,...,16\}$)をシナリオごとに検証データで選択	バス3 / 自動車10 / 歩行者6 / 静止2 / 列車1 ($d=1$ はVanilla相当)
PatchTST	パッチ長・ストライド等	シナリオごとにOptunaで30試行のハイパーパラメータ探索

シナリオ別標準化RMSE(t+30)

シナリオ	LSTM	Vanilla	TDE	PatchTST	AE	MLP
バス	0.846	0.656	0.672	0.770	0.834	0.861
自動車	0.456	0.507	0.483	0.475	0.514	0.510
歩行者	0.430	0.548	0.473	0.429	0.471	0.503
静止	0.166	0.159	0.162	0.183	0.174	0.182
列車	0.783	0.836	0.836	0.804	0.849	0.877
5シナリオ平均	0.536	0.541	0.525	0.532	0.568	0.586
5シナリオ平均標準偏差	0.047	0.035	0.039	0.011	0.022	0.026

- TDE Transformer : 5シナリオ平均RMSEが最小 LSTM比 -2.0%, Vanilla比 -3.0%
- PatchTSTはAE・MLPを一貫して上回り (-6.4%/-9.3%) , 標準偏差も全手法中最小 (0.011)で最も安定した予測を実現

予測先別RMSE(バス/自動車)



バス：Vanilla / PatchTST が全予測先でLSTM未満

自動車：LSTMが全予測先で最小，Transformer系はやや劣る

→ RMSEが最小となる手法は移動シナリオ・予測先に依存し、一貫してどれか一手法が優位というわけではない

計算コスト (Irish自動車, h=450, seed42)

手法	RMSE	時間 [s]	GPU [MiB]
LSTM	0.455	428	377
Vanilla	0.502	1994	1292
TDE (d=5)	0.501	2935	1292
PatchTST	0.501	357	100

- PatchTSTが実行時間・GPUメモリともに最小：LSTM比で時間 **-16.6%**, GPUメモリ **-73.5%**
- Vanilla比では時間 **-82.1%**, GPUメモリ **-92.3%**
(自己回帰復号がない分, 一括出力が有利)
- この条件ではLSTMが最小RMSEだが, PatchTSTはVanillaと同程度の精度をより少ない計算資源で達成

考察：シナリオ別の結果

- RMSE最小のモデルはシナリオごとに異なり，Transformer導入だけでは精度向上は保証されない
- バス：Vanilla / PatchTST が全予測先でLSTM未満
→ 自己注意機構が時点間の関係を的確に捉えたためと考えられる
- 自動車・歩行者：LSTMが最小
→ 急峻・不規則な変動への対応は今後の課題
- シード間の標準偏差（5シード平均）はPatchTSTが0.011で全手法中最小，LSTM（0.047）の約1/4：シードによらず一貫して安定した予測性能を発揮する

考察：計算コストとモデル選択

- PatchTSTの最適パッチ長は移動状態に依存（バス/自動車/歩行者/静止=64, 列車=8）→ 移動シナリオに応じたパッチ長設計が精度に影響
- TDE Transformerは精度最良だが計算コストは最長：時間遅延埋め込みが入力次元を増やし、学習・推論コストを増大させる
- PatchTSTはVanilla同等の精度を、大幅に少ない計算資源（実行時間-82.1%, GPUメモリ-92.3%）で達成、一括出力（Decoderなし）が自己回帰復号のコストを解消
- PatchTSTはシード間の標準偏差も6手法中最小（0.011, LSTM比約1/4）
- PatchTSTは精度で6手法中2位, 標準偏差は6手法中最小, 計算コストは比較した4手法中最小：単独最良ではないが、3指標すべてで上位に入る唯一の手法である

まとめ

- WLQEに向けたPatchTSTおよびTDE Transformerの適用方法を検討し，LSTM等との比較実験系を構築
 - 5シナリオ平均RMSE：
LSTM 0.536/Vanilla 0.541/TDE 0.525/PatchTST 0.532
 - TDE TransformerはLSTM比 -2.0%，PatchTSTはAE比 -6.4%・MLP比 -9.3%
 - バスシナリオではVanilla / PatchTSTが全予測先でLSTMを上回る一方，最小RMSEの手法はシナリオ依存
 - PatchTSTは自動車シナリオでVanilla同等精度を
実行時間 -82.1%，GPUメモリ -92.3%で達成
- PatchTSTは精度で6手法中2位，標準偏差は6手法中最小，計算コストは比較した4手法中最小：3指標すべてで上位に入る唯一の手法であり，複数ステップ予測における実用的な手法である

参考文献

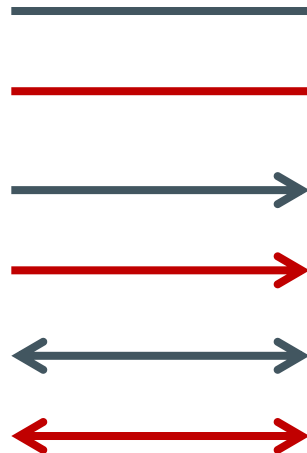
- [1] G. Cerar, H. Yetgin, M. Mohorčič, and C. Fortuna, "Machine Learning for Wireless Link Quality Estimation: A Survey," IEEE Communications Surveys and Tutorials, vol.23, no.2, pp.696–728, 2021.
- [2] Y. Kanto and K. Watabe, "Wireless Link Quality Estimation Using LSTM Model," Proc. IEEE/IFIP NOMS 2024, Seoul, Korea, 2024.
- [3] Y. Nie, N. Nguyen, P. Sinthong, and J. Kalagnanam, "A Time Series is Worth 64 Words: Long-term Forecasting with Transformers," Proc. ICLR 2023, Kigali, Rwanda, 2023.
- [4] N. Wu, B. Green, X. Ben, and S. O'Banion, "Deep Transformer Models for Time Series Forecasting: The Influenza Prevalence Case," arXiv, 2020.
- [5] C. Boucetta, B. Nour, A. Cusin, and H. Moun gla, "QoS in IoT networks based on link quality prediction," Proc. IEEE ICC 2021, Montreal, QC, Canada, 2021.
- [6] P. Millan, C. Molina, E. Medina, D. Vega, R. Meseguer, B. Braem, and C. Blondia, "Time Series Analysis to Predict Link Quality of Wireless Community Networks," Computer Networks, vol.93, no.2, pp.342–358, 2015.
- [7] S. Hochreiter and J. Schmidhuber, "Long Short-Term Memory," Neural Computation, vol.9, no.8, pp.1735–1780, 1997.
- [8] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A.N. Gomez, L. Kaiser, and I. Polosukhin, "Attention Is All You Need," NIPS 2017.
- [9] F. Takens, "Detecting Strange Attractors in Turbulence," Dynamical Systems and Turbulence, Warwick 1980, Lecture Notes in Mathematics vol.898, pp.366–381, Springer, 1981.
- [10] D. Raca, J.J. Quinlan, A.H. Zahran, and C.J. Sreenan, "Beyond Throughput: A 4G LTE Dataset with Channel and Context Metrics," Proc. ACM MMSys 2018, Amsterdam, Netherlands, 2018.

テンプレート集

黒文字



赤文字

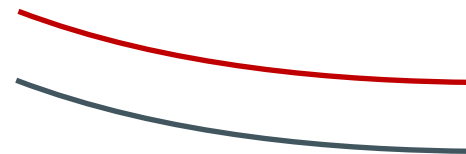


×



●

●



[1] K.Watabe, xxx

作成時の注意点

文字サイズは28ptを基本とし、24pt以下の文字は原則禁止。

図中の文字についてもこれの基準に準じる。

ただし英文はやや小さい文字を許容可能とする。

フォントはヒラギノ角ゴ6WまたはHelvetica boldに統一する。

基本カラーは #46555f (R70:G85:B95, R27%:G33%:B37%)。

強調は**アクセント1**のカラーで、無意味にそれ以外を使うことは不可。

聞き手は異なる表現は異なる意味があると捉えるので、同じ意味なら統一する。

体言止め(語尾の省略)は意味をわかりにくくするので極力避ける。

箇条書きでも必ず句読点を付ける。

付けないと改行により文が区切れるか否かがわかりにくい。

参考文献は引用しているページの下にテンプレート集にある文字サイズで表示する。