Feinabstimmung

Autor und Herausgeber

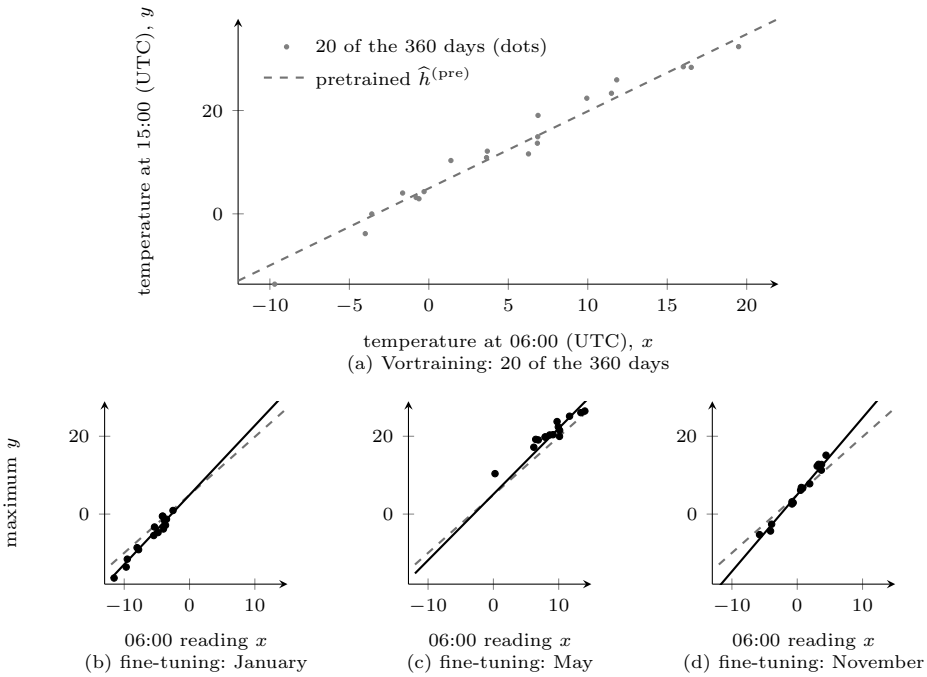
Alexander Jung*, Department of Computer Science, Aalto University, Espoo, Finland (ORCID 0000-0001-7538-0990)
Editors: Konstantina Olioumtsevits (ORCID 0000-0003-2697-8659), Ekkehard Schnoor (ORCID 0000-0001-8252-1499), Aalto University, Espoo, Finland
*Corresponding author: alex.jung@aalto.fi

Zusammenfassung

Feinabstimmung setzt das Training eines Modell auf einem aufgabenspezifischen Datensatz fort, ausgehend von Modellparameter, die durch Vortraining auf einem größeren Datensatz erhalten wurden. Sie löst ein empirische Risikominimierung (ERM)-Problem über einem kleinen Trainingsdatensatz unter der Bedingung, dass die Modellparameter nahe an den vortrainierten bleiben. Eine Form der Feinabstimmung führt einige Gradientenabstieg-Schritte auf dem kleinen Trainingsdatensatz aus, beginnend bei den vortrainierten Modellparameter. Das Erzwingen, dass die Modellparameter nahe an den vortrainierten bleiben, ist eine Form der Regularisierung, da es den effektiven Hypothesenraum begrenzt, der während der Feinabstimmung durchsucht wird. Für ein lineares Modell wächst der generalization gap proportional dazu, wie weit sich die Modellparameter von den vortrainierten entfernen dürfen, weshalb ein kleines Trainingsdatensatz ausreicht.

Definition

Man betrachte das tägliche Wetter in Krems an der Donau. Ein lineares Modell, das die Lufttemperatur um 15:00 (UTC) aus dem Messwert um 06:00 desselben Tages vorhersagt, wird auf 360 Tagen eines Jahres trainiert, was eine vortrainierte Hypothese ergibt. Die aufgabenspezifischen Datensätze sind die Temperaturmessungen eines einzelnen Monats, und eine Aufgabe besteht darin, einen auf diesen Monat zugeschnittenen Vorhersager zu lernen. Für einen einzelnen Monat passen dreißig Gradientenabstieg-Schritte auf den 30 Tagen dieses Monats die Modellparameter an, ausgehend von denen, die über das ganze Jahr gelernt wurden (Abb. 1). Feinabstimmung ist dieser zweite Schritt: Sie setzt das Training eines Modell auf einem aufgabenspezifischen Datensatz fort, ausgehend von Modellparameter, die durch Vortraining auf einem größeren Datensatz erhalten wurden (Bishop and Bishop, 2024, Abschn. 6.3.4).



(a) Vortraining: 20 of the 360 days

(b) fine-tuning: January

(c) fine-tuning: May

(d) fine-tuning: November

Abbildung 1: Vortraining und Feinabstimmung eines lineares Modells, das die Temperatur um 15:00 (UTC) aus dem 06:00-Messwert in Krems an der Donau vorhersagt. (a) Zwanzig der 360 Tage eines Jahres, jeder 18. Tag (graue Punkte), und die Hypothese $\widehat{h}^{(\text{pre})}$, die aus allen 360 gelernt wurde (gestrichelte Linie). (b) bis (d) Jeder zweite Tag im Januar, Mai und November, 15 von 30 Tagen jedes Monats (schwarze Punkte; alle drei Felder im gleichen Temperaturbereich), die vortrainierte Hypothese (gestrichelte Linie) und die Hypothese, die durch dreißig Gradientenabstieg-Schritte auf allen 30 Tagen dieses Monats, ausgehend von den vortrainierten Modellparameter, erhalten wurde (durchgezogene Linie). Jede feingestimmte Hypothese senkt den durchschnittlichen quadratischen Fehler des Monats unter den der vortrainierten Hypothese, während die Modellparameter um weniger als 1.5 verschoben werden. Daten generiert von `pythondemos/finetuning.py`

Formal sei $f(\mathbf{w})$ der durchschnittliche Verlust auf dem aufgabenspezifischen Trainingsdatensatz $\mathcal{D}^{(\text{train})} = \{\mathbf{z}^{(r)}\}_{r=1}^{m}$,

$$f(\mathbf{w}) := \frac{1}{m} \sum_{r=1}^{m} L\left(\mathbf{z}^{(r)}, \mathbf{w}\right).$$

Feinabstimmung löst das empirische Risikominimierung (ERM)-Problem für f nicht über dem gesamten Hypothesenraum, sondern unter der Bedingung, dass

die Modellparameter innerhalb eines Radius B der vortrainierten $\widehat{\mathbf{w}}^{(\mathrm{pre})}$ bleiben,

$$\widehat{\mathbf{w}} \in \operatorname*{arg\,min}_{\mathbf{w} \in \overline{\mathcal{B}}_B(\widehat{\mathbf{w}}^{(\mathrm{pre})})} f(\mathbf{w}).$$

Hier ist $\overline{\mathcal{B}}_B(\widehat{\mathbf{w}}^{(\mathrm{pre})}) := \{\mathbf{w} : \|\mathbf{w} - \widehat{\mathbf{w}}^{(\mathrm{pre})}\| \leq B\}$ die abgeschlossene Kugel mit Radius B um $\widehat{\mathbf{w}}^{(\mathrm{pre})}$ (siehe Abb. 2).

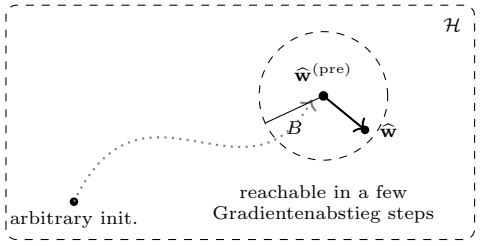


Abbildung 2: Feinabstimmung im Raum der Modellparameter. Die Modellparameter, die die Methode zurückgeben kann, liegen in der Kugel $\overline{\mathcal{B}}_B(\widehat{\mathbf{w}}^{(\mathrm{pre})})$ (gestrichelter Kreis) und nicht irgendwo im Hypothesenraum; einige Gradientenabstieg-Schritte auf einem kleinen Trainingsdatensatz können sie nicht weiter tragen. Das Erreichen desselben Bereichs von einer beliebigen Initialisierung (gepunkteter Pfad) würde weitaus mehr gelabelte Datenpunkte erfordern.

Der Radius B ist es, der ein kleines Trainingsdatensatz ausreichen lässt. Man betrachte ein lineares Modell, dessen Hypothesen $h^{(\mathbf{w})}(\mathbf{x}) = \mathbf{w}^\top \mathbf{x}$ sind. Seien die Merkmalsvektoren von Norm höchstens R, und sei der Verlust ρ-Lipschitz in der Vorhersage und auf der Kugel durch c beschränkt, wie es der hinge loss ist. Seien die Datenpunkte Realisierungen von unabhängig und identisch verteilt (u.i.v.) Zufallsvariablen (ZV). Dann erfüllt mit Wahrscheinlichkeit von mindestens $1-\delta$ jedes $\mathbf{w} \in \overline{\mathcal{B}}_B(\widehat{\mathbf{w}}^{(\mathrm{pre})})$

$$\bar{L}\big(h^{(\mathbf{w})}\big) \leq f(\mathbf{w}) + \frac{2\rho B R}{\sqrt{m}} + c\sqrt{\frac{2\ln(2/\delta)}{m}}.$$

Dies ist die Schranke von Shalev-Shwartz and Ben-David (2014), Satz 26.12 für Modellparameter von Norm höchstens B, angewendet auf die verschobenen Modellparameter $\mathbf{w} - \widehat{\mathbf{w}}^{(\mathrm{pre})}$. Die Verschiebung addiert die feste Zahl $(\widehat{\mathbf{w}}^{(\mathrm{pre})})^\top \mathbf{x}$ zur Vorhersage für jeden Datenpunkt, was der Kontraktionsschritt des Beweises erlaubt, da er eine separate ρ-Lipschitz-Funktion pro Datenpunkt zulässt (Shalev-Shwartz and Ben-David, 2014, Lemma 26.9). Der zweite Term folgt aus der Rademacher complexity der Kugel, die höchstens $BR/\sqrt{m}$ beträgt (Shalev-Shwartz and Ben-David, 2014, Lemma 26.10).

Der generalization gap zwischen Risiko und Trainingsfehler $f(\mathbf{w})$ wächst daher proportional zum Radius B und nicht mit der Anzahl d der Modellparameter. Eine Lücke von höchstens ε wird durch $m = \left\lceil \left(2\rho B R + c\sqrt{2\ln(2/\delta)}\right)^2 / \varepsilon^2 \right\rceil$

Datenpunkte garantiert, eine Zahl, die mit dem Radius B wächst: Das Nahebleiben an den vortrainierten Modellparameter ist es, was die Feinabstimmung mit wenigen Datenpunkte erfolgreich macht. Die Bedingung schränkt das Modell ein, das die Methode durchsucht, was Regularisierung durch das Beschneiden eines Hypothesenraum erreicht. Der Unterschied ist, dass die beschnittene Menge um die vortrainierten Modellparameter zentriert ist, anstatt um eine beliebige Initialisierung.

Gradientenabstieg mit wenigen Schritten und einer kleinen Schrittweite, beginnend bei den vortrainierten Modellparameter, implementiert diese Bedingung, ohne sie explizit zu erzwingen. Jeder Schritt verschiebt die Modellparameter um das η-fache der Norm des Gradient. Nach T Schritten mit Schrittweite η, beginnend bei $\mathbf{w}^{(0)} = \widehat{\mathbf{w}}^{(\text{pre})}$,

$$\|\mathbf{w}^{(T)} - \widehat{\mathbf{w}}^{(\text{pre})}\| \leq \eta \sum_{t=0}^{T-1} \|\nabla f(\mathbf{w}^{(t)})\| \leq \eta T G,$$

wobei G die Norm des Gradient entlang der Iterationen begrenzt. Die Iteration liegt daher in der Kugel mit Radius $B = \eta T G$ um $\widehat{\mathbf{w}}^{(\text{pre})}$. Die Begrenzung der Anzahl der Iterationen ist somit eine Form der Regularisierung: Sie begrenzt den effektiven Hypothesenraum, der während der Feinabstimmung durchsucht wird, wobei die Schrittweite und die Iterationsanzahl die Rolle des Radius spielen.

Feinabstimmung ist nützlich, wenn das aufgabenspezifische Datensatz klein ist. Ein Krankenhaus verfügt über einige hundert annotierte Scans und benötigt einen Tumordetektor. Das Training eines tiefes Netz aus einer beliebigen Initialisierung auf so vielen gelabelte Datenpunkte führt normalerweise zu Überanpassung; das Starten von Modellparameter, die bereits aus Millionen von nicht verwandten Fotografien gelernt wurden, erreicht einen niedrigeren Validierungsfehler (Goodfellow et al., 2016, Abschn. 15.2; Yosinski et al., 2014). Große Sprachmodelle (LLM) werden auf die gleiche Weise angepasst: Ein auf unbeschriftetem Text vortrainiertes Modell wird für jede gewünschte Aufgabe feingestimmt (Devlin et al., 2019). Eine gängige Variante friert einige Schichten ein und passt nur den Rest an (Yosinski et al., 2014); parameter-effiziente Feinabstimmung beschränkt das Update auf eine niedrigdimensionale Menge neuer Modellparameter, während $\widehat{\mathbf{w}}^{(\text{pre})}$ unberührt bleibt (Hu et al., 2022).

Das Experiment `pythondemos/finetuning.py` misst den Effekt auf synthetischen Daten für lineare Regression mit $m = 15$ Datenpunkte und $d = 30$ Modellparameter. Gradientenabstieg, gestartet bei den vortrainierten Modellparameter mit Schrittweite $\eta = 0.02$, fällt nach $T = 16$ Schritten auf einen Validierungsfehler unter das Doppelte des Rauschpegels. Es hat dann die Modellparameter um eine Abstand von 0.32 verschoben, innerhalb der Kugel mit Radius $\eta T G = 0.72$. Die gleiche Iteration von einer beliebigen Initialisierung treibt den Trainingsfehler auf Null, endet aber mit einem Validierungsfehler, der mehr als viermal größer ist, nachdem sie eine Abstand von 2.6 zurückgelegt hat. Abb. 3 skizziert den gleichen Effekt auf einem schematischen aufgabenspezifischen Datensatz von vier Datenpunkte.

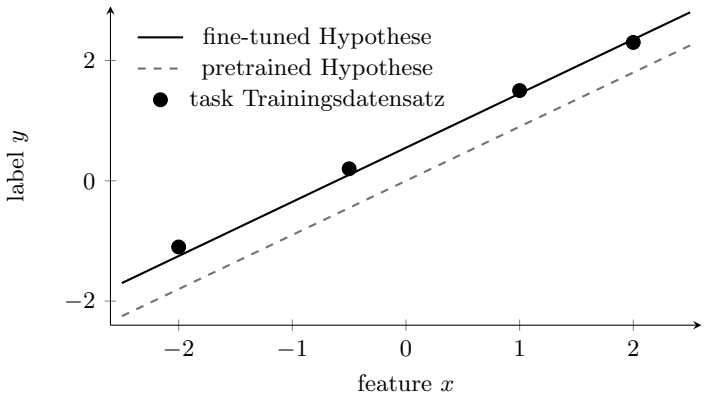


Abbildung 3: Feinabstimmung auf einem aufgabenspezifischen Datensatz von vier Datenpunkte. Die vortrainierte Hypothese (gestrichelte Linie) ist bereits nah an der Aufgabe; einige Gradientenabstieg-Schritte auf den vier Datenpunkte des Aufgaben-Trainingsdatensatzs (Punkte) verschieben sie auf die durchgezogene Kurve. Von einer beliebigen Initialisierung aus müssten die vier Datenpunkte allein die gesamte Kurve bestimmen.

Synonyme: fine tuning.

Siehe auch: Vortraining, Transfer-Learning, foundation model, selbstüberwachtes Lernen, ERM, Gradientenabstieg, Regularisierung, Rademacher complexity, Überanpassung, generalization gap, Validierungsfehler.

Querverweise

Literatur

1. Bishop and Bishop (2024). Deep Learning: Foundations and Concepts. Springer Nature, Cham, Switzerland.

2. Devlin et al. (2019). BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In: Proc. 2019 Conf. North American Chapter Assoc. Computational Linguistics: Human Lang. Technologies, Volume 1 (Long and Short Papers), pp. 4171–4186.

3. Goodfellow et al. (2016). Deep Learning. MIT Press, Cambridge, MA, USA.

4. Hu et al. (2022). LoRA: Low-Rank Adaptation of Large Language Models. In: Int. Conf. Learn. Represent. (ICLR).

5. Shalev-Shwartz and Ben-David (2014). Understanding Machine Learning: From Theory to Algorithms. Cambridge Univ. Press, New York, NY, USA.

6. Yosinski et al. (2014). How Transferable Are Features in Deep Neural Networks?. In: Adv. Neural Inf. Process. Syst. (NIPS), pp. 3320–3328.