Intrinsic Dimensionality Explains the Effectiveness of Language Model Fine-Tuning

第13回 最先端NLP勉強会

2021年 9月 16日

紹介者：篠田一聡 (東大 相澤研 D2)

書誌情報

Intrinsic Dimensionality Explains the Effectiveness of Language Model Fine-Tuning

Armen Aghajanyan
Facebook AI
armenag@fb.com

Sonal Gupta
Facebook
sonalgupta@fb.com

Luke Zettlemoyer
Facebook AI
University of Washington
lsz@fb.com

ACL2021 long
Outstanding paper

導入

事前学習済み言語モデルのFine-tuningはなぜうまくいくのか？

背景

- BERT等の事前学習済み言語モデルを何らかのタスクでfine-tuningすることが、ほとんどのNLPタスクで行われている。

- しかし、なぜタスクのデータが少ないのにfine-tuningがうまく行くのかまだよく分かっていない。
  - モデルのパラメータ数は数億で、なぜか多い方が精度がいい
  - 訓練データ数は数百〜数千で、モデルのパラメータ数よりかなり少ない
  - 単純な勾配降下法で十分に学習が行えて高い精度を出せる

やったこと

- Intrinsic dimensionality (Li et al., 2018) を使って、事前学習済み言語モデルを分析

Intrinsic dimensionとは (一言で)

- Intrinsic dimension (Li et al., 2018) は、あるデータセットで精度や指標がある一定の値をとるのに必要な最小のパラメータ数

本研究の主な貢献

事前学習言語モデルについてわかったこと

- よく使われるNLPタスクのintrinsic dimensionは数百～数千 << モデルのパラメータ数
  - ＝数千のパラメータを学習するだけで高い精度を出せてしまう
- 言語モデルの事前学習をするほどintrinsic dimensionは減少する
- モデルのパラメータ数が多いほど、 intrinsic dimensionが小さい
- 実験的・理論的に、intrinsic dimensionが小さいほど汎化誤差が小さい

Intrinsic dimensionとは (詳しく)

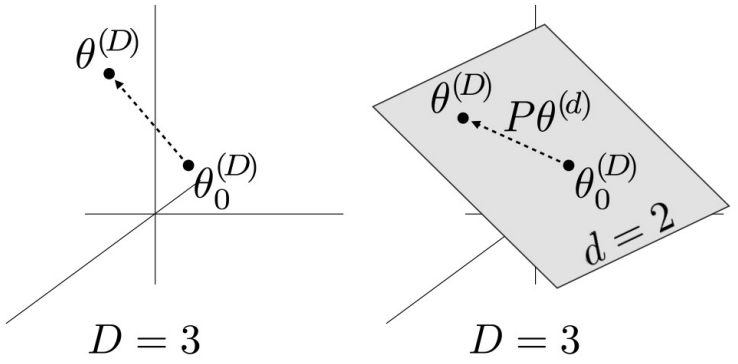
- Intrinsic dimension (Li et al., 2018) は、あるデータセットで精度や指標がある一定の値をとるのに必要な最小の学習可能なパラメータ数

$$\theta^D = \theta_0^D + P(\theta^d)$$

θ_0^D: モデルの学習前のパラメータ (D次元空間の１点)
θ^D: モデルの学習後のパラメータ
θ^d: d次元空間のベクトル (d < D)

$$P : \mathbb{R}^d \rightarrow \mathbb{R}^D$$

P: d次元空間の点(ベクトル)をD次元空間の点(ベクトル)に変換する関数



(出典: Li et al., 2018)

この例では2(=d)次元ベクトルをPで3(=D)次元に変換している。

- Pによって総パラメータ数がDのモデルを、dのパラメータだけで学習できる。
- 原理的には、モデル、データセット、関数P、達成すべき指標の値が決まると、Intrinsic dimensionが決まる

$P : \mathbb{R}^d \to \mathbb{R}^D$ をどうやって計算するか(1/2)

- Li et al. (2018)が提案したもののうち、計算コストの削減を優先して、Fastfood transform (Le et al., 2013) (以下M) を採用

$$\theta^D = \theta_0^D + \theta^d M \qquad M = HG\Pi HB$$

  - H: Hadamard matrix
  - G: random diagonal matrix with independent standard normal entries
  - B: random diagonal matrix with equal probability ± 1 entries
  - Π: random permutation matrix

- 要は掛け算の計算コストが低い5つの行列をつなげただけ。
- 訓練中、動かせるパラメータはθ^dのみで、 他 (θ_0^D やM) はランダムに初期化して固定
- これを Direct Intrinsic Dimension (**DID**) と呼ぶことにする。

$P : \mathbb{R}^d \to \mathbb{R}^D$ をどうやって計算するか(2/2)

- Layerを考慮してIntrinsic dimensionを計算した方がモデルの構造を考慮できて良い

$$\theta_i^D = \theta_{0,i}^D + \lambda_i P(\theta^{d-m})_i$$

  - Layerごとに係数λ_iをかける。
  - 動かせるパラメータは、θ^{d-m} と λ_1~λ_mのd個
  - Structure-Aware Intrinsic Dimension (**SAID**) と呼ぶことにする。

この研究での Intrinsic dimensionの求め方

何を求めるか？

- パラメータ数Dのモデルを使って85%の精度を達成できるなら、その9割の精度 (85%*0.9 = 76.5%) を達成するのに必要な最小のパラメータ数=d_{90}を求める

どうやって求めるか？

1. 動かせるパラメータ数がDの時の精度αを求める。
2. dを適当に決める→SGDで最適なθ^dを探す→精度β_dを求める
3. もし余裕があれば、とにかくいろいろなdで 2. を試す
   余裕がなければ、二分探索など
4. 試したdの中で、精度β_dがαの9割を超えているdのうち最小のdがd_{90}

実験

実験1: NLPタスクでのIntrinsic Dimensionality

- 手始めにパラフレーズか否かを予測する以下の 2 つのタスクで d_{90} を求める。
  - MRPC（訓練データ：3700）
  - QQP（訓練データ：363k）

各データセット・各モデルについて、 learning rate を4種類、d は10から10000の間の100種類を試して、9割の精度に達した最小のdがintrinsic dimension。

実験1: NLPタスクでのIntrinsic Dimensionality

わかったこと

- **全体的にモデルのパラメータ数D（数億）に比べてかなり少ないパラメータ（数百～数千）を学習するだけで元の9割の精度を達成できる。**
- d_{90} はRoBERTa-Large < RoBERTa-Base < BERT-Large < BERT-Base
- 一貫して、SAID < DID … モデルの構造を考慮することでd_{90}を減らせる。

考察

- P: d→Dの決め方がランダムでかなり雑であったことを考えると、真のd_{90}はもっと小さいことが予想される。

Model	SAID		DID	
	MRPC	QQP	MRPC	QQP
BERT-Base	1608	8030	1861	9295
BERT-Large	1037	1200	2493	1389
RoBERTa-Base	896	896	1000	1389
RoBERTa-Large	**207**	**774**	322	**774**

Table 1: Estimated d_{90} intrinsic dimension computed with SAID and DID for a set of sentence prediction tasks and common pre-trained models.

Figure 1: Evaluation accuracy on two datasets and four models across a range of dimensions d for the DID method. The horizontal lines in each figure represent the 90% solution of the respective full model.

実験2: Pre-Training and Intrinsic Dimensionality

- 言語モデルの事前学習によってIntrinsic dimensionが小さくなっているのではないか？
  - → RoBERTa-Baseをスクラッチから訓練して、訓練中のチェックポイントのintrinsic dimensionalityを計算

実験2: Pre-Training and Intrinsic Dimensionality

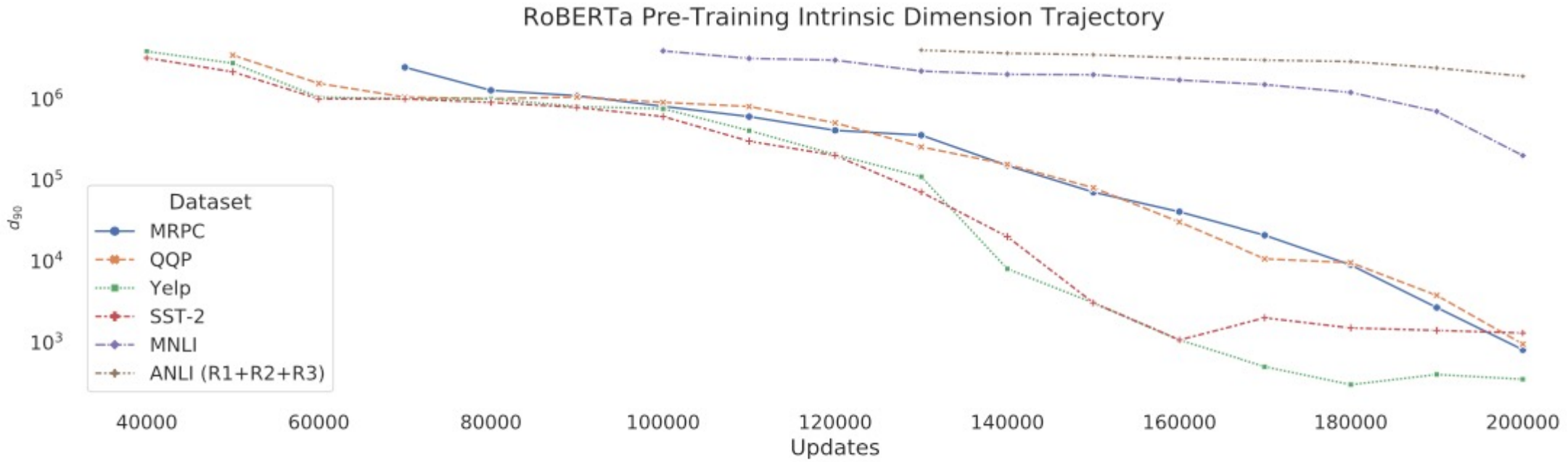


Figure 2: Every 10k updates of RoBERTa-Base that we trained from scratch, we compute d_{90} for six datasets; MRPC, QQP, Yelp Polarity, SST-2, MNLI, and ANLI. If we were unable to compute a d_{90} for a specific checkpoint, we do not plot the point, hence some datasets start at later points. Unable to compute means either we could not fine-tune the full checkpoint to accuracy above majority class or stabilize SAID training.

実験2: Pre-Training and Intrinsic Dimensionality

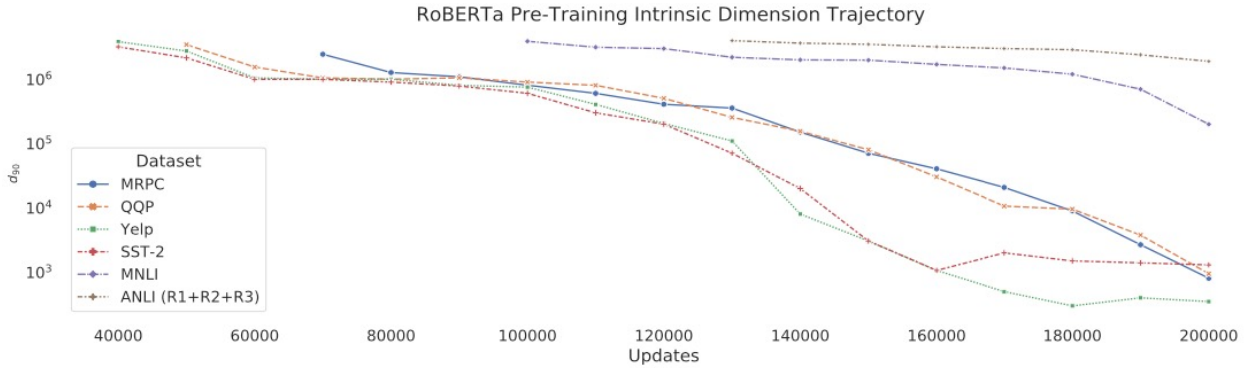


Figure 2: Every 10k updates of RoBERTa-Base that we trained from scratch, we compute d_{90} for six datasets; MRPC, QQP, Yelp Polarity, SST-2, MNLI, and ANLI. If we were unable to compute a d_{90} for a specific checkpoint, we do not plot the point, hence some datasets start at later points. Unable to compute means either we could not fine-tune the full checkpoint to accuracy above majority class or stabilize SAID training.

わかったこと

- **ダウンストリームのデータセットにアクセスできないのにも関わらず、事前学習が進むほど、どのタスクでもIntrinsic dimension が減少していく**
- 難しいダウンストリームタスク(ANLIなど)ほど、 Intrinsic dimension が大きい⇒汎化と関係がある

実験3: Parameter Count and Intrinsic Dimension

- モデルのパラメータ数とIntrinsic dimensionには関係があるのではないか？
  - →様々な事前学習済み言語モデルでMRPCデータセットでのIntrinsic dimensionを計算した

実験3: Parameter Count and Intrinsic Dimension

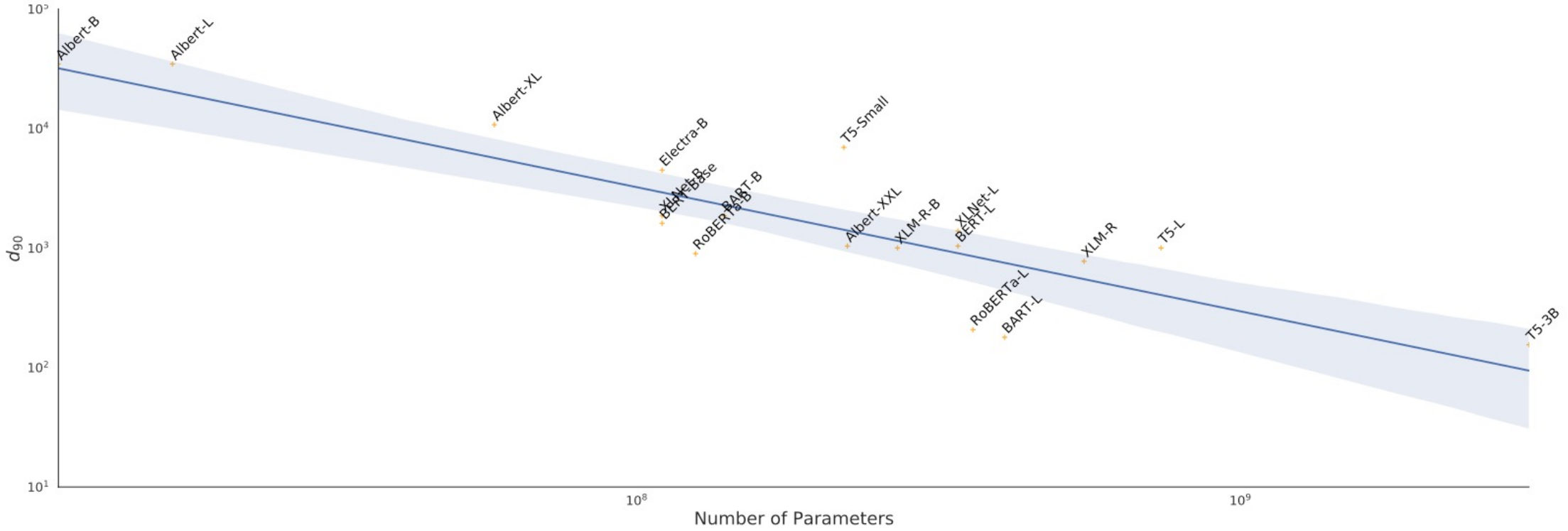


Figure 3: We calculate the intrinsic dimension for a large set of pre-trained models using the SAID method on the MRPC dataset.

実験3: Parameter Count and Intrinsic Dimension

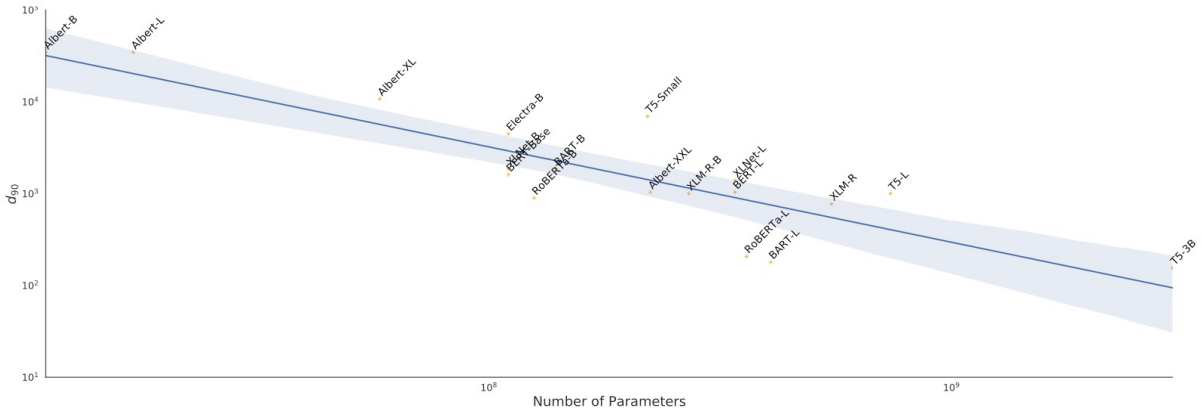


Figure 3: We calculate the intrinsic dimension for a large set of pre-trained models using the SAID method on the MRPC dataset.

わかったこと

- **モデルのパラメータ数が多いほど、Intrinsic dimensionが減少する傾向**
- 違うデータセットでも試したが同様の傾向が見られた
- モデルのパラメータ数が同程度の場合は、事前学習方法が重要になる。

実験4: Generalization Bounds through Intrinsic Dimension

- Intrinsic dimensionが減少すれば汎化性能も向上するのではないか？
  - → 実験3で作ったRoBERTaの事前学習中のチェックポイントを使って実験

実験4: Generalization Bounds through Intrinsic Dimension

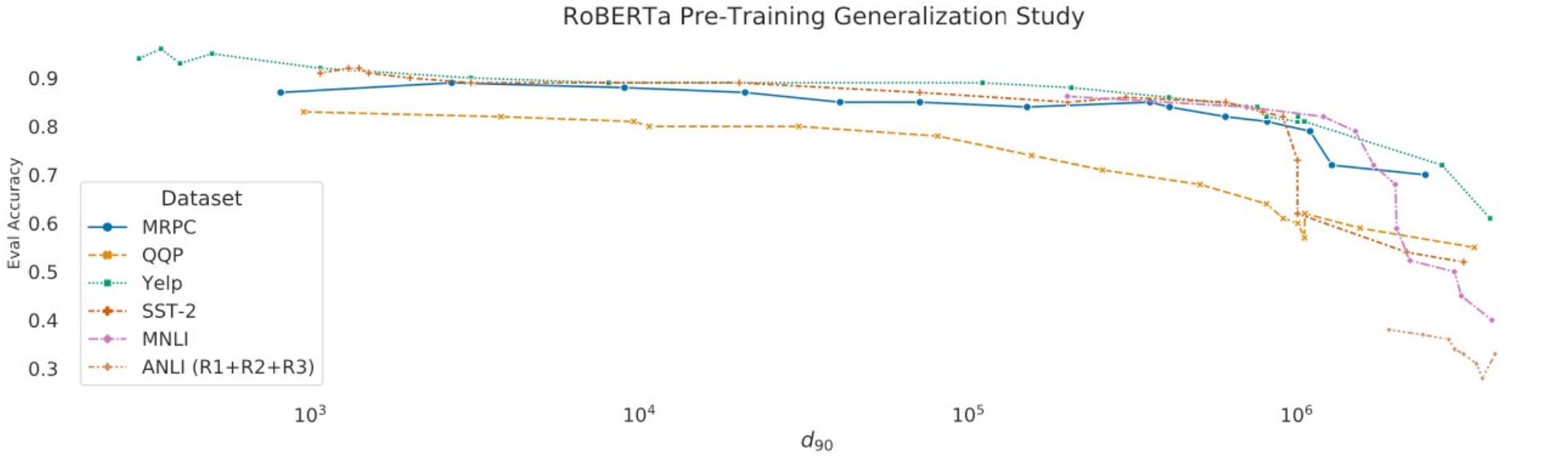


Figure 4: The evaluation accuracy of six datasets across various intrinsic dimensionalities. There is a strong general trend that pre-trained models that are able to attain lower intrinsic dimensions generalize better.

わかったこと

- **Intrinsic dimensionが小さいほど、ダウンストリームタスクでの精度が高い**

実験4: Generalization Bounds through Intrinsic Dimension

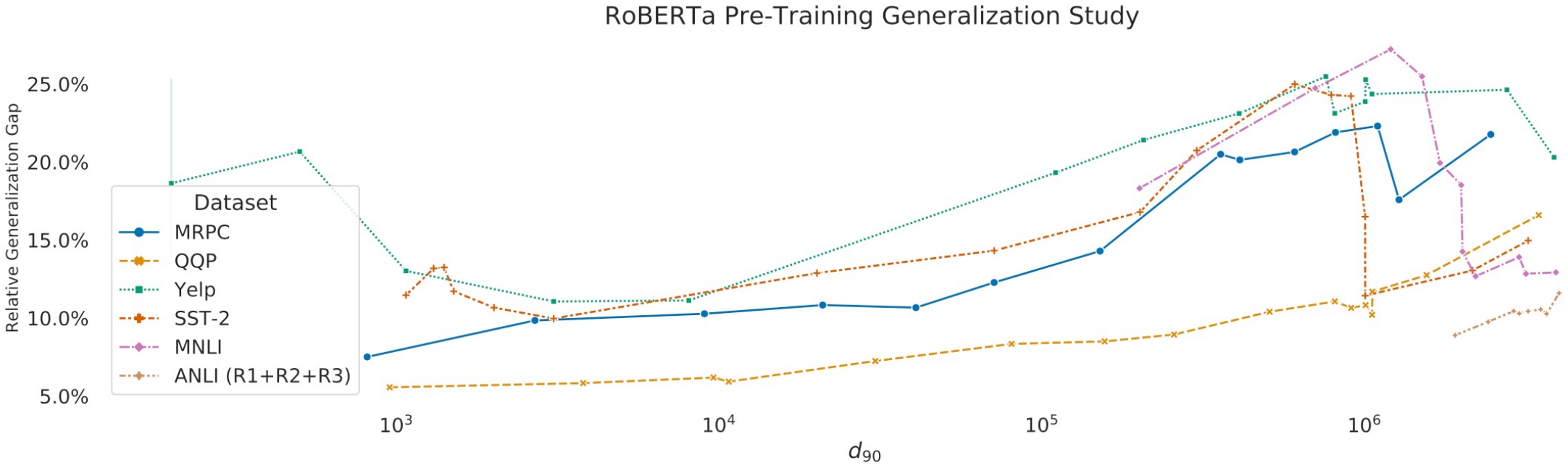


Figure 5: The intrinsic dimension and the respective relative generalization gap across a set of varied tasks.

わかったこと

- **Intrinsic dimensionが小さいほど、relative generalization gapが小さい（よく汎化できる）**

※Relative generalization gap $= \frac{acc_{train} - acc_{eval}}{1 - acc_{eval}}$

理論的な裏付け

Generalization Bounds

- Intrinsic dimension と 汎化の関係について理論的な裏付け
  - Arora et al. (2018) の Theorem 2.1 に基づく
- f: classifier, d: intrinsic dimension, m: データの数

$$\mathcal{L}_\gamma(f) = \mathbb{P}_{(x,y)\sim D}\left[f(x)[y] \leq \gamma + \max_{j\neq y} f(x)[j]\right]$$

Marginありの
多クラス分類の誤差

$$\mathcal{L}_0(f) \leq \hat{\mathcal{L}}_0(f) + \mathcal{O}\left(\sqrt{\frac{d}{m}}\right)$$

圧縮に基づく
汎化誤差のバウンド

期待誤差
（汎化誤差）

経験誤差
（訓練誤差）

- **Intrinsic dimensionが小さいほど、汎化誤差は小さいことが理論的に裏付けられる**

参考: https://www.slideshare.net/trinmu/ss-161240890

Conclusion

わかったこと

- 実験1→NLPタスクのintrinsic dimensionは数百〜数千 << モデルのパラメータ数
- 実験2→言語モデルの事前学習をするほどintrinsic dimensionは減少する
- 実験3→モデルのパラメータ数が多いほど、 intrinsic dimensionが小さい
- 実験4, 理論的な裏付け→intrinsic dimensionが小さいほど汎化誤差が小さい

筆者の主張

- Intrinsic dimensionはダウンストリームタスクの最小記述長として解釈できる
- 事前学習言語モデルのサイズが大きいほど汎化性能が高いのは、パラメータ数の多さが直接的な要因ではなく、パラメータ数の多いほどIntrinsic dimensionが小さい→ Intrinsic dimensionが小さいほど汎化性能が高いから

Future work

- SGDとintrinsic dimensionの減少の関係を明らかにする
- 事前学習時にfine-tuning時のデータを見ていないのにダウンストリームタスクのintrinsic dimensionの減少が可能なのはなぜかを明らかにする

感想

- この研究で得られた知見をどう応用していくのかは自明ではない？

References

- Chunyuan Li, Heerad Farkhoor, Rosanne Liu, and Jason Yosinski. 2018. Measuring the intrinsic dimension of objective landscapes. *arXiv preprint arXiv:1804.08838*.
- Quoc Le, Tama ´s Sarlo ´s, and Alex Smola. 2013. Fastfood-approximating kernel expansions in loglinear time. In *Proceedings of the international conference on machine learning*, volume 85.
- Sanjeev Arora, Rong Ge, Behnam Neyshabur, and Yi Zhang. 2018. Stronger generalization bounds for deep nets via a compression approach. arXiv preprint arXiv:1802.05296.
- 深層学習の数理 (https://www.slideshare.net/trinmu/ss-161240890)