Group-Sparse Matrix Factorization for Transfer Learning of Word Embeddings

Kan Xu
University of Pennsylvania, Economics, kanxu@sas.upenn.edu

Xuanyi Zhao
University of Pennsylvania, Computer and Information Science, xuanyi.zhao@hotmail.com

Hamsa Bastani
Wharton School, Operations Information and Decisions, hamsab@wharton.upenn.edu

Osbert Bastani
University of Pennsylvania, Computer and Information Science, obastani@seas.upenn.edu

Unstructured text provides decision-makers with a rich data source in many domains, ranging from product reviews in retail to nursing notes in healthcare. To leverage this information, words are typically translated into *word embeddings*—vectors that encode the semantic relationships between words—through unsupervised learning algorithms such as matrix factorization. However, learning word embeddings from *new* domains with limited training data can be challenging, because the meaning/usage may be different in the new domain, e.g., the word "positive" typically has positive sentiment, but often has negative sentiment in medical notes since it may imply that a patient tested positive for a disease. In practice, we expect that only a small number of domain-specific words may have new meanings. We propose an intuitive two-stage estimator that exploits this structure via a group-sparse penalty to efficiently *transfer learn* domain-specific word embeddings by combining large-scale text corpora (such as Wikipedia) with limited domain-specific text data. We bound the generalization error of our transfer learning estimator, proving that it can achieve high accuracy with substantially less domain-specific data when only a small number of embeddings are altered between domains. Furthermore, we prove that all local minima identified by our nonconvex objective function are statistically indistinguishable from the global minimum under standard regularization conditions, implying that our estimator can be computed efficiently. Our results provide the first bounds on group-sparse matrix factorization, which may be of independent interest. We empirically evaluate our approach compared to state-of-the-art fine-tuning heuristics from natural language processing.

Key words: word embeddings, transfer learning, group sparsity, matrix factorization, natural language processing (NLP), text analytics

1. Introduction

Natural language processing is an increasingly important part of the analytics toolkit for leveraging unstructured text data in a variety of domains. For instance, service providers mine online consumer reviews to inform operational decisions on platforms (Mankad et al. 2016) or to infer market structure and the competitive landscape for products (Netzer et al. 2012); Twitter posts are used

to forecast TV show viewership (Liu et al. 2016); analyst reports of S&P 500 firms are used to measure innovation (Bellstam et al. 2020); medical notes are used to predict operational metrics such as readmissions rates (Hsu et al. 2020); online ads or reviews are used to flag service providers that are likely engaging in illicit activities (Ramchandani et al. 2021, Li et al. 2021).

To leverage unstructured text in decision-making, we must preprocess the text to capture the semantic content of words in a way that can be passed as an input to a predictive machine learning algorithm. In the past, this involved domain experts performing costly and imperfect feature engineering. A much more powerful, data-driven approach is to use unsupervised learning algorithms to learn *word embeddings*, which represent words as vectors (Mikolov et al. 2013, Pennington et al. 2014); we focus on widely-used word embedding models that are based on low-rank matrix factorization (Pennington et al. 2014, Levy and Goldberg 2014). These word embeddings translate semantic similarities between words and the context within which they appear into statistical relationships. Typically, they are trained to encode how frequently pairs of words co-occur in text; these co-occurrence counts implicitly contain semantic properties of words since words with similar meanings tend to occur in similar contexts. Given the large number of words in the English language, to be effective in practice, embeddings must be trained on large-scale and comprehensive text data, e.g., popular embeddings such as Word2Vec (Mikolov et al. 2013) and GloVe (Pennington et al. 2014) are trained on Wikipedia articles.

However, it is well-known that pre-trained word embeddings can miss out on important domain-specific meaning/usage, hurting downstream interpretation and effectiveness. Take the healthcare domain as an example. The word "positive" is typically associated with positive sentiment on Wikipedia; yet, in the context of medical notes, it typically indicates the presence of a medical condition, corresponding to negative sentiment. Thus, using a generic word embedding for "positive" may diminish performance in medical applications. Similarly, words like "adherence" (referring to medication adherence) have a specific meaning in a healthcare context (relative to its context on general Wikipedia entries) and are strongly predictive of patient outcomes; failing to account for its healthcare-specific meaning may result in a loss in the downstream accuracy of healthcare-specific prediction tasks (Blitzer et al. 2007). Consequently, there has been a large body of work training specialized embeddings in a number of diverse contexts, ranging from radiology reports (Ong et al. 2020), stock market prediction (Li and Shah 2017), cybersecurity vulnerability reports (Roy et al. 2017), and patent classification (Risch and Krestel 2019). This approach only works when the decision-maker has access to a sufficiently large domain-specific text corpus, allowing her to train high-quality embeddings. In practice, decision-makers often have limited domain-specific text data, yielding poor results when training new word embeddings, which hurts the quality of

downstream modeling and decisions that leverage these embeddings. In other words, word embeddings trained on domain-specific data alone are unbiased but can have high variance due to limited sample size; in contrast, pre-trained word embeddings have low variance but can be significantly biased depending on the extent of domain mismatch.

Then, a natural question is whether we can combine large-scale publicly available text corpora (which we call the *proxy* data hereafter) with limited domain-specific text data (which we call the *gold* data hereafter) to train precise but domain-specific word embeddings. In particular, we aim to use transfer learning to achieve a better bias-variance tradeoff than using gold or proxy data alone. Our key insight to enable transfer learning is that the meaning/usage of most words do not change when changing domains; rather, we expect that only a small number of domain-specific words will have new meaning/usage. To illustrate, Figure 1 shows text data (paragraphs) from a variety of domain-specific Wikipedia articles, including finance, math, computing, and politics. Words that have a domain-specific meaning are enclosed in a red box,[1] while the remaining words share the same meaning/usage as in the standard English language. We observe that only a small number of unique words have domain-specific meaning/usage.

More formally, consider a corpus of d words. Let $U_p \in \mathbb{R}^{d\times r}$ denote the true (unobserved) proxy word embedding matrix, of which the i^{th} row $U_p^{(i,\cdot)}$ is the true r-dimensional word embedding of word $i \in [d] = \{1, \cdots, d\}$ based on the proxy data; analogously, let $U_g \in \mathbb{R}^{d\times r}$ denote the true (unobserved) gold word embedding matrix. We expect that the meaning/usage for most words are preserved in both domains—i.e., the word embeddings $U_g^{(i,\cdot)} \neq U_p^{(i,\cdot)}$ for only a small number $s \ll d$ values of $i \in [d]$. This induces a *group-sparse* structure for the difference matrix $U_g - U_p$, i.e., only a small number s of the rows (groups) are nonzero. Figure 2 illustrates this notion of "sparsity" on a toy example with $d = 10$ words, embeddings with dimension $r = 5$, and $s = 3$ words with shifted meaning/usage. Indeed, we find support for this group-sparse structure in our previous examples from Wikipedia—e.g., in the finance domain (Fig 1(a)), we observe only $s = 4$ unique finance domain-specific words (put, options, stock, strike) out of a total of $d = 51$ distinct words, yielding a sparsity ratio $s/d \lesssim 0.08$. Similarly for the other domains in Fig 1, the sparsity ratios s/d are approximately 0.11, 0.07 and 0.05 for the math, computing, and politics examples respectively. (Details and experiments on the Wikipedia data can be found in §5.2.)

Based on this intuition, we formulate an objective that incorporates a group-sparse penalty (Friedman et al. 2010, Simon et al. 2013) on $U_g - U_p$, where each row is treated as a group. In particular, we estimate domain-specific embeddings from gold data, incorporating $\ell_{2,1}$ regularization to impose group sparsity relative to the (estimated) word embeddings trained on the large proxy

[1] Briefly, we categorize a word as domain-specific if any of the word's definitions on Wiktionary is labeled with key words from that specific domain; see §5.2 for details.

Put options are most commonly used in the stock market to protect against a fall in the price of a stock below a specified price. If the price of the stock declines below the strike price, the holder of the put has the right, but not the obligation, to sell the asset at the strike price, while the seller of the put has the obligation to purchase the asset at the strike price if the owner uses the right to do so (the holder is said to *exercise* the option). In this way the buyer of the put will receive at least the strike price specified, even if the asset is currently worthless.

(a) Finance Domain - "Put Option"

The divisors of a natural number n are the natural numbers that divide n evenly. Every natural number has both 1 and itself as a divisor. If it has any other divisor, it cannot be prime. This leads to an equivalent definition of prime numbers: they are the numbers with exactly two positive divisors. Those two are 1 and the number itself. As 1 has only one divisor, itself, it is not prime by this definition.[6] Yet another way to express the same thing is that a number n is prime if it is greater than one and if none of the numbers $2, 3, \ldots, n-1$ divides n evenly.[7]

(b) Math Domain - "Prime Number"

Client–server systems are usually most frequently implemented by (and often identified with) the request–response model: a client sends a request to the server, which performs some action and sends a response back to the client, typically with a result or acknowledgment. Designating a computer as "server-class hardware" implies that it is specialized for running servers on it. This often implies that it is more powerful and reliable than standard personal computers, but alternatively, large computing clusters may be composed of many relatively simple, replaceable server components.

(c) Computing Domain - "Server"

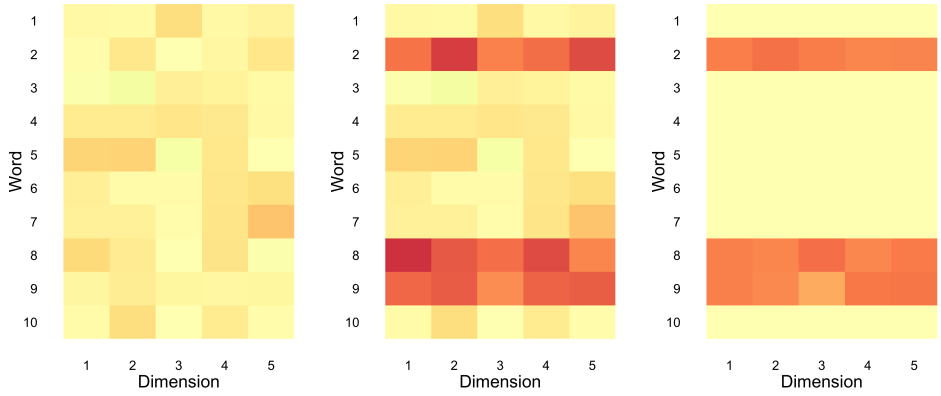
Some political scientists, such as Samuel P. Huntington, have seen conservatism as situational. Under this definition, conservatives are seen as defending the established institutions of their time.[12] According to Quintin Hogg, the chairman of the British Conservative Party in 1959: "Conservatism is not so much a philosophy as an attitude, a constant force, performing a timeless function in the development of a free society, and corresponding to a deep and permanent requirement of human nature itself."[13] Conservatism is often used as a generic term to describe a "right-wing viewpoint occupying the political spectrum between [classical] liberalism and fascism".[1]

(d) Politics Domain - "Conservatism"

Figure 1 Paragraphs extracted from four Wikipedia articles of four domains respectively. We enclose domain-specific words in red boxes, distinguishing the first occurrence (solid line) from subsequent occurrences (dashed line). See §5.2 for our definition of domain words and other experiments on Wikipedia data.

data. Our approach balances the need to update the embeddings of important domain-specific words based on the gold data (i.e., reduce bias), while matching most words to the embeddings estimated from the large proxy text corpus (i.e., reduce variance).

Our main result establishes that the word embedding estimator trained by group-sparse transfer learning achieves a sample complexity bound that, to leading order, scales quadratically in s (the number of words with altered meaning/usage), as opposed to the conventional bound that scales quadratically in d (the total number of words). In other words, transfer learning allows us to



(a) U_p (b) U_g (c) $U_g - U_p$

Figure 2 **Toy example of (a) proxy and (b) gold word embedding matrices for $d = 10$ and $r = 5$. Only $s = 3$ words change meaning/usage, inducing a group-sparse structure in the (c) difference matrix. The colors represent the magnitude of coefficients, ranging from zero (yellow) to large (red).**

accurately identify domain-specific word embeddings with substantially less domain-specific data than classical low-rank matrix factorization methods. We build on prior work establishing error bounds for the group LASSO (Lounici et al. 2011) and low-rank matrix problems (Ge et al. 2017, Negahban and Wainwright 2011). We face two additional technical challenges. First, the literature on nonconvex low-rank matrix problems typically studies the Hessian to ensure that local minima are well-behaved; however, the Hessian may not be well-defined under our nonsmooth group-sparse penalty (since the gradient is not continuous). Second, unlike the traditional high-dimensional literature, transfer learning introduces a quartic form (in terms of $U_g - U_p$) in our objective function. We address both challenges through a new analysis that relies on an assumption we term "quadratic compatibility condition." We show that quadratic compatibility is implied by a natural restricted strong convexity (RSC) assumption, which we prove holds with high probability in a general low-rank matrix factorization problem for the illustrative cases of gaussian data and word co-occurrence count data. Furthermore, under a slightly weaker condition that can characterize all local minima (Loh and Wainwright 2015), all local minima identified by our algorithm are statistically indistinguishable from the global minimum, implying that our estimator can be computed efficiently.

While our technical results hold for embeddings trained using matrix factorization, our algorithm straightforwardly applies to nonlinear objectives such as GloVe. Simulations on synthetic data and domain-specific Wikipedia articles show that our estimator significantly outperforms common heuristics given rich proxy data and limited domain-specific data. Importantly, we show that this is an *interpretable* strategy to identifying key words with distinct meanings in specific domains such as finance, math, and computing.

1.1. Related Literature

Transfer learning involves transferring knowledge from a data-rich source domain to a data-poor target domain (also called "domain adaptation"). In order for such approaches to be effective, the two domains must be related in some way. For instance, the two domains may have the same label distribution $p(y \mid x)$ but different covariate distributions $p(x)$, a setting typically termed as "covariate shift" (see, e.g., Ben-David et al. 2007, 2010, Ganin and Lempitsky 2015). Our problem falls into the more challenging category known as "label shift," where $p(y \mid x)$ itself differs across the two domains (since the underlying embeddings change for some words). A number of approaches have been proposed for addressing label shift in supervised learning problems (see, e.g., Lipton et al. 2018, Zhang et al. 2013).[2] Our approach is most closely related to recent work applying LASSO for transfer learning (Bastani 2020), where the label shift is driven by a *sparse* shift in the underlying parameter vectors. Their key theoretical result is that relative sparsity between the gold and proxy parameter vectors is sufficient to enable efficient transfer learning in high dimensions. Existing theoretical results are critically limited to supervised learning. To the best of our knowledge, we propose the first framework for theoretically understanding the value of transfer learning in natural language processing (generally considered an unsupervised learning problem), which introduces new technical challenges.

However, a number of practical heuristics have been proposed for domain adaptation for natural language processing. A surprisingly effective transfer learning strategy is to simply *fine-tune* pre-trained word embeddings on data from the target domain. Intuitively, stochastic gradient descent has regularization properties similar to ℓ_2 regularization (Ali et al. 2020), so this strategy can be interpreted as regularizing the target word embeddings towards the pre-trained word embeddings (Dingwall and Potts 2018, Yang et al. 2017). We demonstrate empirically that our approach of using ℓ_1 regularization outperforms these heuristics in the low-data regime.

We build on approaches that construct word embeddings based on low-rank matrix factorization (Pennington et al. 2014, Levy and Goldberg 2014). Levy and Goldberg (2014) show that one popular approach—skip-gram with negative sampling—implicitly factorizes a word-context matrix shifted by a global constant. Another popular approach is GloVe (Pennington et al. 2014), which uses a nonlinear version of our loss function; our estimator extends straightforwardly to this setting.

Accordingly, we build on the theoretical literature on low-rank matrix factorization—specifically the Burer-Monteiro approach (Burer and Monteiro 2003), which replaces Θ with a low-rank representation UU^T, with $U \in \mathbb{R}^{d \times r}$, and minimizes the objective in U. Ge et al. (2017) shows that the local minima of this nonconvex problem are also global minima under the restricted isometry

[2] Problems with labeled source data and unlabeled target data are sometimes referred to as "unsupervised"; we categorize them as "supervised" to distinguish from problems where both source and target data are unlabeled.

property; Li et al. (2019) extend this by considering a more general objective function that satisfies a restricted well-conditioned assumption. One alternative is nuclear-norm regularization (Recht et al. 2007, Candes and Plan 2011, Negahban and Wainwright 2011), but this algorithm lends less naturally to our transfer learning objective and is often computationally inefficient.

This paper extends our earlier short conference paper (Anonymous 2021) as follows. First, we show that the quadratic compatibility condition (a critical component of our proofs) is implied by a natural restricted strong convexity condition, which we prove holds with high probability in a general low-rank matrix factorization problem for the illustrative cases of gaussian data and word co-occurrence count data (§2.4). Second, more importantly, we prove that all local minima identified by our estimator are statistically indistinguishable from the global minimum under a slightly weaker condition proposed by Loh and Wainwright (2015) that is likely to hold for all local minima (§3.3). This result significantly strengthens our main result by showing that the optimization problem used to compute our estimator is tractable in practice. Third, we relate our error bounds back to the scaling specific to word embedding models (Corollary 1–3). Finally, we significantly expand the experimental results on both synthetic and real data to illustrate the value and robustness of our approach.

2. Problem Formulation

We first formalize the problem of learning word embeddings as a low-rank matrix sensing problem (§2.1), and describe our transfer learning approach (§2.2). We then state our assumptions (§2.3) and provide intuition for our quadratic compatibility condition (§2.4).

Notation. For any vector $v \in \mathbb{R}^d$, let $\|v\|$ denote its ℓ_2 norm. For a matrix $\Theta \in \mathbb{R}^{d_1 \times d_2}$ of rank r, we denote its singular values by $\sigma_{\max}(\Theta) = \sigma_1(\Theta) \geq \sigma_2(\Theta) \geq \cdots \geq \sigma_r(\Theta) = \sigma_{\min}(\Theta) > 0$, its Frobenius norm by $\|\Theta\|_F = \sqrt{\sum_{j=1}^r \sigma_j^2(\Theta)}$, its operator norm by $\|\Theta\| = \sigma_1(\Theta)$, its vector ℓ_∞ norm by $|\Theta|_\infty = \max_{i,j} |\Theta^{(i,j)}|$, its vector ℓ_1 norm by $|\Theta|_1 = \sum_{i,j} |\Theta^{(i,j)}|$, and its matrix $\ell_{2,1}$ norm by $\|\Theta\|_{2,1} = \sum_{j=1}^{d_1} \|\Theta^j\|$. We use superscript (i,j) to represent entry (i,j) of a matrix Θ, $(i,\cdot)$ the i^{th} row of the matrix, and $(\cdot,j)$ the j^{th} column. Given $\Theta, \Theta' \in \mathbb{R}^{d_1 \times d_2}$, we denote the matrix dot product by $\langle \Theta, \Theta' \rangle = \sum_{i=1}^{d_1} \sum_{j=1}^{d_2} \Theta^{(i,j)} \Theta'^{(i,j)}$. Finally, let $[k] = \{1, 2, \cdots, k\}$.

2.1. Matrix Sensing

Our word embedding model is an instance of the more general setting of matrix sensing (Recht et al. 2007), where one aims to recover an unknown symmetric matrix $\Theta^* \in \mathbb{R}^{d \times d}$ with rank $r \ll d$. In other words, we can write $\Theta^* = U^* U^{*T}$ where $U^* \in \mathbb{R}^{d \times r}$. The typical goal in matrix sensing is to estimate Θ^* given observation matrices $A_i \in \mathbb{R}^{d \times d}$ and $X_i \in \mathbb{R}$, for $i \in [n]$, where

$$X_i = \langle A_i, \Theta^* \rangle + \epsilon_i, \tag{1}$$

and $\epsilon_1, \cdots, \epsilon_n$ are independent σ-subgaussian random variables (Definition 1). This model is introduced to generalize low-rank matrix factorization (Recht et al. 2007)—the observation matrices A_i allow for general observation models of the underlying low-rank model Θ^*, intuitively playing a similar role as covariates in classical linear regression.

To simplify notation, we define the linear operator $\mathcal{A}: \mathbb{R}^{d\times d} \to \mathbb{R}^n$, where $\mathcal{A}(\Theta)_i = \langle A_i, \Theta\rangle$. Then, we can write

$$X = \mathcal{A}(\Theta^*) + \epsilon,$$

where $X = [X_1, \cdots, X_n]^T$ and $\epsilon = [\epsilon_1, \cdots, \epsilon_n]^T$.

DEFINITION 1. A random variable Z is σ-subgaussian if, for any $t \in \mathbb{R}$, $\mathbb{E}[Z] = 0$ and $\mathbb{E}[\exp(tZ)] \leq \exp(\sigma^2 t^2/2)$.

As we will discuss at the end of this subsection, in natural language processing, Θ^* corresponds to the word co-occurrence probability matrix, while U^* corresponds to the word embeddings. Thus, in contrast to the matrix sensing literature which aims to estimate Θ^*, our goal is to estimate the low-rank representation U^*. However, we can only compute U^* up to an orthogonal change-of-basis since Θ^* is preserved under such a transformation—i.e., if we let $\widetilde{U}^* = U^*R$ for an orthogonal matrix $R \in \mathbb{R}^{r\times r}$, then we still obtain $\widetilde{U}^*\widetilde{U}^{*T} = U^*RR^TU^{*T} = U^*U^{*T} = \Theta^*$. Thus, our goal is to compute $\widehat{U}$ such that $\widehat{U} \approx U^*R$ for some orthogonal matrix R.

We build on Burer and Monteiro (2003), which solves the following optimization problem:

$$\min_{U\in\mathbb{R}^{d\times r}} \frac{1}{n}\|X - \mathcal{A}(UU^T)\|^2.$$

Despite its nonconvex loss, this estimator performs well in practice, and has desirable theoretical properties (i.e., no spurious local minima) under the restricted isometry property (Ge et al. 2017).

We measure the estimation error of $\widehat{U}$ using the $\ell_{2,1}$ norm, which is more compatible with the group-sparse structure that we will impose shortly. In addition, since we can only identify U^* up to orthogonal change-of-basis, we consider the following rotation-invariant error.

DEFINITION 2. Given $\widehat{U}, U^* \in \mathbb{R}^{d\times r}$, the error of $\widehat{U}$ is

$$\ell(\widehat{U}, U^*) = \|\widehat{U} - U^*R_{(\widehat{U},U^*)}\|_{2,1},$$

where $R_{(\widehat{U},U^*)} = \arg\min_{R:R^TR=RR^T=\mathbf{I}} \|\widehat{U} - U^*R\|_F$.

REMARK 1. An alternative approach to Burer-Monteiro is to estimate Θ^* directly using nuclear norm regularization (see, e.g., Candes and Plan 2011, Negahban and Wainwright 2011). However, this approach is often too computationally costly in large-scale problems (Recht et al. 2007). Furthermore, estimating U^* is more natural in our setting since our final goal is to recover U^* (rather than Θ^*), and our transfer learning strategy penalizes deviations in U^*.

Word embeddings. Word embedding models typically consider how often pairs of words co-occur within a fixed-length window. Without loss of generality, we consider neighboring word pairs, i.e., a window with length 1. Let the length of our text corpus be $n+1$ so that the total number of neighboring word pairs is n. Recall that we have d unique words, and we define our word co-occurrence matrix to be $\Theta^* \in \mathbb{R}^{d\times d}$, where the (j,k) entry $\Theta^{*(j,k)}$ is the probability that word j and word k appear together. To estimate each of these d^2 probabilities, e.g., $\Theta^{*(j,k)}$ of word pair (j,k), we randomly draw n word pairs from the text with replacement and record the outcome as a binary indicator for whether the draw matches the pair (j,k). We draw samples independently across all d^2 possible word pairs.[3] This yields d^2n samples in total; for each $i \in [d^2n]$, the outcome is a binary variable named X_i that takes value 1 if the draw i is exactly the pair (j,k) and 0 otherwise. We encode the corresponding word pair (j,k) in a basis matrix $A_i \in \mathbb{R}^{d\times d}$, whose (j,k) entry equals 1 and 0 otherwise — i.e., $A_i = E_{jk}$ where E_{jk} is the basis matrix with entry (j,k) being 1 and 0 otherwise. Note that the i^{th} draw corresponds to the word pair (j,k) with probability $\Theta^{*(j,k)} = \langle A_i, \Theta^* \rangle$. Therefore, we can think of X_i as a Bernoulli random variable with mean $\langle A_i, \Theta^* \rangle$, i.e., $X_i \sim \texttt{Bernoulli}(\langle A_i, \Theta^* \rangle)$, and our observation model has the form

$$X_i = \langle A_i, \Theta^* \rangle + \epsilon_i, \tag{2}$$

where A_i is a basis matrix, X_i is a Bernoulli random variable, and ϵ_i is the noise. The model in (2) has been used in other applications (without transfer learning), e.g., for recommendation systems in Farias and Li (2019)—their outcome X_i is a binary indicator that equals 1 if customer j has purchased product k in the past month and 0 otherwise, and their $\langle A_i, \Theta^* \rangle$ is the probability of such transactions. We discuss how our general results scale under this model in §3.2.

2.2. Transfer Learning

We now consider transfer learning from a large text corpus to the desired target domain. Let $U_p^* \in \mathbb{R}^{d\times r}$ denote the unknown word embeddings from the proxy (source) domain, and $U_g^* \in \mathbb{R}^{d\times r}$ denote the unknown word embeddings from the gold (target) domain. Our goal is to use data from both domains to estimate U_g^* (up to rotations). In particular, we are given proxy data $\mathcal{A}_p : \mathbb{R}^{d\times d} \to \mathbb{R}^{n_p}$ and $X_p \in \mathbb{R}^{n_p}$ from the source domain, along with gold data $\mathcal{A}_g : \mathbb{R}^{d\times d} \to \mathbb{R}^{n_g}$ and $X_g \in \mathbb{R}^{n_g}$ from the target domain, such that

$$X_p = \mathcal{A}_p(\Theta_p^*) + \epsilon_p \quad \text{and} \quad X_g = \mathcal{A}_g(\Theta_g^*) + \epsilon_g,$$

[3] In practice, one could simply enumerate all n word pairs to construct each $\Theta^{*(j,k)}$ instead of using sampling, e.g., this is typically how the GloVe model (Pennington et al. 2014) is trained. However, for the purpose of establishing convergence guarantees, using the same n observed word pairs to estimate each $\Theta^{*(j,k)}$ induces non-independence in our observations. To remedy this, we randomly draw n word pairs with replacement for each pair (j,k) independently.

where $\epsilon_p \in \mathbb{R}^{n_p}$ and $\epsilon_g \in \mathbb{R}^{n_g}$ are independent σ_p- and σ_g-subgaussian random variables respectively.

We are interested in the setting where $(n_g/\sigma_g^2) \ll (n_p/\sigma_p^2)$. As we will discuss later, this regime holds when we have limited domain-specific data but a large text corpus from other domains.

Group-Sparse Structure. To enable transfer learning, we must assume some relationship between the proxy and gold domains. Motivated by our previous discussion, we assume that the bias term

$$\Delta_U^* = U_g^* - U_p^*,$$

has a row-sparse structure—i.e., most of its rows are 0. This structure arises when the embeddings of most words are preserved across domains, but a few words have a different meaning/usage in the new domains (see illustration in Figure 2c). More precisely, let the index set

$$J = \left\{ j \in [d] \;\middle|\; \|\Delta_U^{*j}\| \neq 0 \right\},$$

correspond to the set of rows with nonzero entries. The *group sparsity* of Δ_U^* is $s = |J|$. Then, a high-quality estimate of U_p^* (from the large text corpus) can help us recover U_g^* with less data, since the sample complexity of estimating Δ_U^* (due to its sparse structure) is less than that of U_g^*.

Note that the row-sparse structure of Δ_U^* is preserved under orthogonal transformations that are applied to both U_g^* and U_p^*—i.e., if $\widetilde{U}_p^* = U_p^* R$ and $\widetilde{U}_g^* = U_g^* R$ for an orthogonal matrix R, then $\widetilde{\Delta}_U^* = \widetilde{U}_g^* - \widetilde{U}_p^* = (U_g^* - U_p^*)R = \Delta_U^* R$ has the same group sparsity as Δ_U^*.

2.3. Assumptions

We make two assumptions on the proxy and gold linear operators. Our first assumption is a standard restricted well-conditionedness (RWC) property on $\mathcal{A}_p$ from the matrix factorization literature (Li et al. 2019), which allows us to recover high-quality estimates of the proxy word embeddings U_p^*.

Definition 3. A linear operator $\mathcal{A}$ satisfies the r-$\mathsf{RWC}(\alpha, \beta)$ condition if

$$\alpha \|Z\|_F^2 \leq \frac{1}{n} \|\mathcal{A}(Z)\|^2 \leq \beta \|Z\|_F^2,$$

with $3\alpha > 2\beta$ and for any $Z \in \mathbb{R}^{d \times d}$ with $\mathsf{rank}(Z) \leq r$.

Assumption 1. *The proxy linear operator $\mathcal{A}_p$ satisfies $2r$-RWC(α_p, β_p).*

The RWC condition ensures sufficient convexity of the loss function near U_p^*, and further, guarantees statistical consistency for all local minima in a nonconvex matrix factorization problem (Li et al. 2019). Specifically, $\alpha\|Z\|_F^2 \leq \frac{1}{n}\|\mathcal{A}(Z)\|^2$ ensures that the loss function has sufficient convexity to recover the low-rank matrix $\Theta_p^* = U_p^* U_p^{*T}$ consistently. This is comparable to the minimum eigenvalue condition in a linear regression problem. The rest of the definition provides sufficient

smoothness in terms of U_p^* so that the nonconvex matrix factorization problems have no spurious local minima—i.e., they are all global minima (Bhojanapalli et al. 2016, Park et al. 2017, Ge et al. 2017). The RWC condition is a generalization of the standard restricted isometry property (RIP) in the matrix factorization literature (see, e.g., Candes and Tao 2005). However, RIP is very restrictive as it requires all the eigenvalues of the Hessian matrix to be within a small range of 1.

Our first assumption is mild since we have a large proxy dataset, i.e., $n_p \gg d^2$. The degrees of freedom of a $d \times d$ matrix Z of rank r is $r(2d-r)$; thus, in general, we only require $n \geq r(2d-r)$ observations to achieve the lower bound in Definition 3. For instance, when $\mathcal{A}$ is a gaussian ensemble, RIP holds with high probability when $n \gtrsim dr$ (Candes and Plan 2011, Recht et al. 2007).

Remark 2. Note that the operator $\mathcal{A}$ in our word embedding model consists of basis matrices; as a result, our model has a relatively lower signal-to-noise ratio (e.g., compared to the case where $\mathcal{A}$ is from a gaussian ensemble), since $\frac{1}{n}\|\mathcal{A}(\Theta)\|^2 \approx \frac{1}{d^2}\|\Theta\|_F^2$. Therefore, when we later present our bounds for the word embedding model, we will scale the parameters α, β in the RWC assumption by $\frac{1}{d^2}$. Such a scaling is standard in the low-rank matrix literature when using basis matrix observations (see, e.g., Ge et al. 2016).

Our second assumption is a quadratic compatibility condition (QCC) on $\mathcal{A}_g$, which allows us to recover U_g^* despite our nonsmooth and quartic objective function. This condition is adapted from the standard compatibility condition in the high dimensional statistics literature (Van De Geer and Bühlmann 2009, Bühlmann and Van De Geer 2011, Lounici et al. 2011, Negahban et al. 2012).

Definition 4 (QCC). A linear operator $\mathcal{A}$ satisfies the quadratic compatibility condition ($\mathsf{QCC}(U^*, \kappa)$) with matrix U^* and constant κ if

$$\frac{1}{n}\|\mathcal{A}(\Delta U^{*T} + U^*\Delta^T + \Delta\Delta^T)\|^2 \geq \frac{\kappa}{s}\left(\sum_{j\in J}\|\Delta^j\|\right)^2,$$

for any $\Delta \in \mathbb{R}^{d\times r}$ that satisfies $\sum_{j\in J^c}\|\Delta^j\| \leq 7\sum_{j\in J}\|\Delta^j\|$.

Assumption 2. *The gold linear operator $\mathcal{A}_g$ satisfies $\mathsf{QCC}(U_g^*, \kappa)$.*

QCC imposes a much weaker convexity requirement than RWC, since RWC is unlikely to hold in the low-data regime ($n_g < dr$). Intuitively, we cannot guarantee a minimum eigenvalue condition holds for $\mathcal{A}_g$ with very few gold samples, precluding us from obtaining high-quality estimates of Θ_g^*. However, we can instead impose a convexity guarantee on a *restricted* subspace that contains $U_g^* - U_p^*$. The same intuition can be found in the LASSO literature (Van De Geer and Bühlmann 2009, Bühlmann and Van De Geer 2011, Negahban et al. 2012) for linear regression—in the low-data regime, we cannot impose the standard minimum eigenvalue condition on the covariance matrix, so we instead impose a compatibility condition on a restricted subspace that contains the non-sparse

elements of the true parameter. Note that our QCC assumption takes a different form than the compatibility condition in group-sparse linear regression (Lounici et al. 2011)—specifically, QCC includes an additional quadratic term $\Delta\Delta^T$ on the left hand side due to the fact that we are studying a nonconvex matrix factorization problem. We give a detailed discussion of this condition in the next subsection.

REMARK 3. Analogous to Remark 2, when we later present our bounds for the word embedding model, we will scale the parameter κ in the QCC assumption by $\frac{1}{d^2}$. Proposition 3 in the next section provides support for this argument.

2.4. Quadratic Compatibility Condition

We now bridge our QCC assumption (Definition 4) with the more standard restricted strong convexity (RSC) condition adapted to our setting; the RSC condition is common in the high-dimensional statistics and low-rank matrix factorization literature (Negahban and Wainwright 2011, 2012, Negahban et al. 2012, Klopp 2014). We prove that the RSC condition holds with high probability in the low-rank matrix factorization problem for the commonly-studied case of gaussian data as well as our word co-occurrence count data.

DEFINITION 5 (RSC). The operator $\mathcal{A}$ satisfies restricted strong convexity ($\mathsf{RSC}(U^*, \eta, \tau)$) with matrix U^*, constant η and function τ if

$$\frac{1}{n}\|\mathcal{A}(\Delta U^{*T} + U^*\Delta^T + \Delta\Delta^T)\|^2 \geq \eta\|\Delta U^{*T} + U^*\Delta^T + \Delta\Delta^T\|_F^2 - \tau(n,d,r)\|\Delta\|_{2,1}^2$$

for any $\Delta \in \mathcal{D} \subset \mathbb{R}^{d\times r}$.

Our condition closely resembles Definition 2 of Negahban et al. (2012). RSC conditions are an alternative to compatibility conditions that provide a weak convexity guarantee for the problem. Indeed, without the last term on the right hand side, the RSC condition is reduced to a minimum eigenvalue condition on a specific low-rank subspace. Typically, the function τ is a small term that is model-dependent and relies on parameters such as n, d, r (see, e.g., Section 4 in Negahban et al. 2012). The following proposition shows that QCC holds given the above RSC condition when considering a bounded set of feasible Δ, i.e., $\|\Delta\|_{2,1} \leq \bar{L}$ for some positive constant $\bar{L}$. Focusing on bounded Δ is not restrictive since we will formulate our transfer learning optimization problem over a compact set in the following section.

PROPOSITION 1. *Assume $\mathcal{A}$ satisfies $\mathsf{RSC}(U^*, \eta, \tau)$ on $\mathbb{R}^{d\times r}$ and $\|U^*\|_{2,\infty} \leq \frac{D}{\sqrt{d}}$ for some constant $D > 0$. If n and d are such that $\frac{\eta\sigma_r^2(U^*)}{32s} \geq 4\frac{\eta D\bar{L}}{\sqrt{d}} + \tau(n,d,r)$, then $\mathcal{A}$ satisfies $\mathsf{QCC}(U^*, \kappa)$ with $\kappa = 2\eta\sigma_r^2(U^*)$.*

The proof is provided in Appendix A. Note that we've imposed that the "row-spikiness" of the matrix U^* is bounded, i.e., $\|U^*\|_{2,\infty} \leq \frac{D}{\sqrt{d}}$, to ensure identifiability (see, e.g., similar assumptions in Agarwal et al. 2012, Negahban and Wainwright 2012). In other words, U^* itself is unlikely to be row-sparse. This matches practice since individual word embeddings (rows) are never zero. Furthermore, one need not employ our transfer learning approach when U^* is row-sparse, since the sample complexity of directly estimating U^* is already low.

Proposition 2 (proof in Appendix A) below shows that an RSC condition holds with high probability when the linear operator $\mathcal{A}$ is sampled from a gaussian ensemble (the most commonly considered setting in the literature). To simplify notation, we define the matrix vectorization operator $\mathsf{vec} : \mathbb{R}^{d_1 \times d_2} \to \mathbb{R}^{d_1 d_2}$ with $\mathsf{vec}(\Theta) = [\Theta^{(\cdot,1)T}, \Theta^{(\cdot,2)T}, \cdots, \Theta^{(\cdot,d_1)T}]^T$. Define an operator $T_\Sigma : \mathbb{R}^{d\times d} \to \mathbb{R}^{d\times d}$ such that $\mathsf{vec}(T_\Sigma(\Theta)) = \sqrt{\Sigma}\,\mathsf{vec}(\Theta)$. We still consider $\|\Delta\|_{2,1} \leq \bar{L}$.

Proposition 2. *Consider a random operator $\mathcal{A}$ sampled from a Σ-gaussian ensemble, i.e., $\mathsf{vec}(A_i) \sim N(0, \Sigma)$. Let $\Sigma' = K^{(d,d)}\Sigma K^{(d,d)}$ with $K^{(d,d)}$ being the commutation matrix, and let*

$$\Sigma = \begin{bmatrix} \bar{\Sigma}_{11} & \bar{\Sigma}_{12} & \cdots & \bar{\Sigma}_{1d} \\ \vdots & \vdots & \ddots & \vdots \\ \bar{\Sigma}_{d1} & \bar{\Sigma}_{d2} & \cdots & \bar{\Sigma}_{dd} \end{bmatrix}, \quad \text{and} \quad \Sigma' = \begin{bmatrix} \bar{\Sigma}'_{11} & \bar{\Sigma}'_{12} & \cdots & \bar{\Sigma}'_{1d} \\ \vdots & \vdots & \ddots & \vdots \\ \bar{\Sigma}'_{d1} & \bar{\Sigma}'_{d2} & \cdots & \bar{\Sigma}'_{dd} \end{bmatrix},$$

with $\bar{\Sigma}_{ij} \in \mathbb{R}^{d\times d}$ the covariance matrix of the i^{th} and j^{th} columns of A_i. Then, with probability greater than $1 - c\exp(-c'n)$ for some constants $c, c' > 0$, we have for any Δ,

$$\frac{\|\mathcal{A}(\Delta U^{*T} + U^*\Delta^T + \Delta\Delta^T)\|}{\sqrt{n}} \geq \frac{1}{4}\|T_\Sigma(\Delta U^{*T} + U^*\Delta^T + \Delta\Delta^T)\|_F - 3C_6\left(\sqrt{\frac{r}{n}} + \frac{3}{2}\sqrt{\frac{\log d}{n}}\right)\|\Delta\|_{2,1},$$

where $C_6 = 2\bar{L}\max_{i\in[d^2]}\sqrt{\Sigma^{(i,i)}} + \sigma_1(U^*)\left(\max_{i\in[d]}\sqrt{\sigma_1(\bar{\Sigma}_{ii})} + \max_{i\in[d]}\sqrt{\sigma_1(\bar{\Sigma}'_{ii})}\right)$.

We now move to our word embedding model, where our observation matrices A_i are basis matrices. The next result reinforces our claim that a similar RSC condition holds with high probability in this setting. Recall that we encode a randomly sampled word pair (j,k) in a basis matrix $A_i \in \mathbb{R}^{d\times d}$, whose (j,k) entry equals 1 and 0 otherwise — i.e., $A_i = E_{jk}$ where E_{jk} is the basis matrix with entry (j,k) being 1 and 0 otherwise. Thus, we consider the linear operator $\mathcal{A}$ being sampled from a standard weighted sampling distribution $\Pi = \{\pi_{jk}\}_{j,k\in[d]}$ with bounded π_{jk}, where $\pi_{jk} = \mathbb{P}(A_i = E_{jk})$; a similar sampling distribution is considered in prior work, e.g., Klopp (2014) and Negahban and Wainwright (2012). Define the $L_2(\Pi)$ norm of a matrix Θ as $\|\Theta\|^2_{L_2(\Pi)} = \mathbb{E}[\langle A_i, \Theta\rangle^2]$.

Proposition 3. *Consider a random operator $\mathcal{A}$ sampled from a weighted sampling ensemble $\Pi = \{\pi_{jk}\}_{j,k\in[d]}$ with $\frac{\mu_1}{d^2} \leq \pi_{jk} \leq \frac{\mu_2}{d^2}$ for some constant μ_1, μ_2. Then, with probability greater than $1 - c\exp(\frac{c'}{B^4}n)$ for some constants $c, c' > 0$, we have for any $\Delta \in \{\Delta \mid \frac{\|\Delta U^{*T} + U^*\Delta^T + \Delta\Delta^T\|_\infty}{\|\Delta U^{*T} + U^*\Delta^T + \Delta\Delta^T\|_{L_2(\Pi)}} \leq B\}$,*

$$\frac{1}{n}\|\mathcal{A}(\Delta U^{*T} + U^*\Delta^T + \Delta\Delta^T)\|^2 \geq \frac{\mu_1}{4d^2}\|\Delta U^{*T} + U^*\Delta^T + \Delta\Delta^T\|^2_F - 36C_7^2\left(\sqrt{\frac{\log d}{nd^2}} + \frac{\log d}{n}\right)^2\|\Delta\|^2_{2,1},$$

where $C_7 = 88B(\bar{L}(\sqrt{2\mu_2}+4/3)+2\sigma_1(U^*)(\sqrt{4r\mu_2}+8/3))$*, and* B *is a positive constant.*

We give a proof in Appendix A. Note that, by construction of the observation matrices in our word embedding model (described in §2.1), each $\pi_{jk} = 1/d^2$ so $\mu_1, \mu_2 = 1$ in Proposition 3 above. This is because, for a given pair (j,k), we draw exactly n samples, among which each sample i has the observation matrix $A_i = E_{jk}$. Therefore, out of the total d^2n samples, we can find A_i takes value E_{jk} with equal probability for all d^2 pairs of (j,k); that is, $\pi_{jk} = \mathbb{P}(A_i = E_{jk}) = 1/d^2$.

As discussed earlier, due to the low signal-to-noise ratio of the operator $\mathcal{A}$ in the word embedding setting, the RSC condition (as well as the QCC condition, by Proposition 1) hold with parameters scaling as $\mathcal{O}(\frac{1}{d^2})$. Therefore, when we discuss our bounds for the word embedding problem (Corollary 1–3), we will scale the parameters by $\frac{1}{d^2}$.

3. Group-Sparse Transfer Learning

In this section, we describe our proposed transfer learning estimator that combines gold and proxy data to learn domain-specific word embeddings. We prove sample complexity bounds, discuss local minima, and illustrate how our estimator can also be leveraged with nonlinear word embedding algorithms such as GloVe.

3.1. Estimation Procedure

Our proposed two-step transfer learning estimator is as follows:

$$\begin{aligned}\widehat{U}_p &= \arg\min_{U_p} \frac{1}{n_p}\|X_p - \mathcal{A}_p(U_pU_p^T)\|^2, \\ \widehat{U}_g^{TL} &= \arg\min_{U_g:\|U_g-\widehat{U}_p\|_{2,1}\le 2L} \frac{1}{n_g}\|X_g - \mathcal{A}_g(U_gU_g^T)\|^2 + \lambda\|U_g - \widehat{U}_p\|_{2,1}. \end{aligned} \tag{3}$$

The first step estimates the proxy word embeddings from a large text corpus; the second step estimates gold word embeddings from limited domain-specific data, regularizing our estimates towards the estimated proxy embeddings via a group-sparse penalty term.

As discussed earlier, our estimator aims to exploit the fact that the bias term $\Delta_U^* = U_g^* - U_p^*$ is group-sparse, and can therefore be estimated much more efficiently than U_g^* itself. In particular, a simple variable transformation on (3) in terms of Δ_U yields:

$$\widehat{\Delta}_U = \arg\min_{\Delta_U:\|\Delta_U\|_{2,1}\le 2L} \frac{1}{n_g}\|X_g - \mathcal{A}_g((\widehat{U}_p+\Delta_U)(\widehat{U}_p+\Delta_U)^T)\|^2 + \lambda\|\Delta_U\|_{2,1}, \tag{4}$$

where our final estimator for the gold data is $\widehat{U}_g^{TL} = \widehat{\Delta}_U + \widehat{U}_p$. Since we have a large proxy dataset, we expect $\widehat{U}_p \approx U_p^*$; when this is the case, we will show that the second stage can efficiently debias the proxy estimator using limited gold domain-specific data. Since our problem is nonconvex and nonsmooth, we follow Loh and Wainwright (2015) and define a compact search region for Δ_U —

i.e., $\|\Delta_U\|_{2,1} \leq 2L$. Here, L is a tuning parameter that should be chosen large enough to ensure feasibility, i.e., we will assume that $\|\Delta_U^*\|_{2,1} = \|U_g^* - U_p^*\|_{2,1} \leq L$.

In (3), the regularization parameter λ trades off bias and variance. When $\lambda \to 0$, we recover the usual low-rank estimator on gold data, which is unbiased but has high variance due to the scarcity of domain-specific data; when $\lambda \to \infty$, we simply obtain the proxy word embeddings, which have low variance but are biased due to domain mismatch. Our main result will provide a suitable value of λ to appropriately balance the bias-variance tradeoff in this setting.

One technical challenge is that, while the group-sparse penalty in (4) would normally be operationalized to recover a group-sparse "true" parameter, this is not the case here due to estimation noise from our first stage. Specifically, the true minimizer of the (expected) low-rank objective on gold data is U_g^*; then, under our variable transformation $\Delta = U_g - \widehat{U}_p$, the corresponding parameter we wish to recover in (4) is not Δ_U^* but rather

$$\widetilde{\Delta}_U = \Delta_U^* - \nu,$$

where $\nu = \widehat{U}_p - U_p^*$ is the residual noise from estimating the proxy word embeddings in the first step. But $\widetilde{\Delta}_U$ is *not* row-sparse unlike Δ_U^*, since ν is not sparse. Thus, we may be concerned that the faster convergence rates promised for the group LASSO estimator may not apply here. On the other hand, we expect our estimation error $\|\nu\|$ to be small since we are in the regime where our proxy dataset is large. Thus, we expect $\widetilde{\Delta}_U$ to be *approximately* row-sparse. We will prove that this is sufficient to recover $\widetilde{\Delta}_U$ (and therefore U_g^*) at faster rates.

REMARK 4. The two-step design of our estimator provides significant practical benefits. In practice, training on a large text corpus can be computationally intensive, so analysts often prefer to download pre-trained word embeddings $\widehat{U}_p$; these can directly be used in the second step of our estimator, which is then trained on the much smaller domain-specific dataset. Furthermore, our approach does not require the proxy and gold datasets to be simultaneously available at training time, which is desirable in the presence of regulatory or privacy constraints.

3.2. Main Result

Our main result characterizes the estimation error of our transfer learning estimator $\widehat{U}_g^{TL}$. We first introduce the following concept of smoothness of our operator $\mathcal{A}$ from Chi et al. (2019); we obtain tighter bounds with higher smoothness, but we show that our problem always satisfies some level of smoothness (as will be made precise in Remark 5).[4]

[4] Note that smoothness here refers to the operator $\mathcal{A}$; our objective function is not smooth due to the group-sparse penalty.

DEFINITION 6. A linear operator $\mathcal{A}:\mathbb{R}^{d\times d}\to\mathbb{R}^n$ satisfies the r-smoothness(β) condition if for any $Z\in\mathbb{R}^{d\times d}$ with $\mathsf{rank}(Z)\le r$, we have that

$$\frac{1}{n}\|\mathcal{A}(Z)\|^2\le\beta\|Z\|_F^2.$$

Intuitively, smoothness alleviates the nonconvexity of the problem, making it easier to identify U_g^* in spite of the nonconvex loss function (Chi et al. 2019). Note that the weakest form of the assumption is when $r=1$, i.e., the upper bound is only imposed for matrices Z with $\mathsf{rank}(Z)\le 1$. Thus, we state the following result with a 1-smoothness assumption on the gold operator:

THEOREM 1. *Assume $\mathcal{A}_g$ satisfies 1-smoothness(β_g). Let*

$$\lambda=\max\left\{\sqrt{\frac{2048L^2\beta_g\sigma_g^2}{n_g}\log(\frac{10d^2}{\delta})},\sqrt{\frac{256\beta_g\sigma_g^2\sigma_1^2(U_g^*)}{n_g}\left(r+2\sqrt{r\log(\frac{5d}{\delta})}+2\log(\frac{5d}{\delta})\right)}\right\}.$$

Suppose n_p and d are such that $\frac{L\sigma_r(U_p^*)(3\alpha_p-2\beta_p)}{8\sqrt{d}}\ge\sqrt{\frac{8\beta_p\sigma_p^2}{n_p}\left(2r(2d+1)\log(36\sqrt{2})+\log(\frac{10}{\delta})\right)}$. Then, with probability at least $1-\delta$, we have

$$\begin{aligned}\ell(\widehat{U}_g^{TL},U_g^*)\le&C_1s\sqrt{\frac{\sigma_g^2}{n_g}\log(\frac{10d^2}{\delta})}+C_2s\sqrt{\frac{\sigma_g^2}{n_g}\left(2r+3\log(\frac{5d}{\delta})\right)}\\&+C_3\sqrt{\frac{\sigma_p^2}{n_p}d\left(2r(2d+1)\log(36\sqrt{2})+\log(\frac{10}{\delta})\right)}\\=&\mathcal{O}\left(\sqrt{\frac{\sigma_g^2s^2(r+\log(\frac{d^2}{\delta}))}{n_g}}+\sqrt{\frac{\sigma_p^2(rd^2+d\log(\frac{1}{\delta}))}{n_p}}\right)\end{aligned}$$

where $C_1=\frac{16\sqrt{2048L^2\beta_g}}{\kappa}$, $C_2=\frac{16\sqrt{256\beta_g\sigma_1^2(U_g^*)}}{\kappa}$, and $C_3=\frac{128\sqrt{2\beta_p}}{(3\alpha_p-2\beta_p)\sigma_r(U_p^*)}$.

We provide a proof in Appendix B. The estimation error bound of our transfer learning estimator consists of two parts and depends on the gold and proxy data respectively. The second term only depends on the proxy data, and captures the variance of estimating the proxy embeddings U_p^* in the first step of (3). The first term characterizes the estimation accuracy of identifying the bias term Δ_U^* via group-sparse penalty in the second stage of (3). Note that the required condition on n_p and d in Theorem 1 is easily satisfied in our "proxy-rich and gold-scarce" setting—i.e., as long as $n_p\gg d^2$, we only require $n_g\gg\log(d)$.

REMARK 5. The operator $\mathcal{A}$ naturally satisfies smoothness (Definition 6) as long as $\mathcal{A}$ has bounded eigenvalues. Specifically, let $\sigma_{\max}(\mathcal{A}^*\mathcal{A})$ be the maximum eigenvalue of $\mathcal{A}^*\mathcal{A}$, defined as

$$\sigma_{\max}(\mathcal{A}^*\mathcal{A})=\sup_{\|R\|_F=1}\langle R,\mathcal{A}^*(\mathcal{A}(R))\rangle.$$

Then $\mathcal{A}$ satisfies r-smoothness(β) for any $\beta\le\sigma_{\max}(\frac{\mathcal{A}^*\mathcal{A}}{n})$ and $r\le d$. Thus, we can simply take $\beta_g=\sigma_{\max}(\frac{\mathcal{A}_g^*\mathcal{A}_g}{n})$ and $r=1$ to satisfy the smoothness assumption in Theorem 1.

Our proof strategy differs from the standard analysis of the Burer-Monteiro method for low-rank problems (Ge et al. 2017) because our focus is on identifying group-sparse structure within a low-rank problem instead of identifying the low-rank structure itself. Furthermore, Ge et al. (2017) mainly base their analysis on the Hessian of the objective function, while the Hessian of our nonsmooth objective function (4) is not well-defined. Our proof adapts high-dimensional techniques for the group LASSO estimator (Lounici et al. 2011) to the nonconvex low-rank matrix factorization problem. Our analysis accounts for quartic (rather than the typical quadratic) dependence on the target parameter, for which we leverage QCC rather than the standard compatibility condition.

In §4, we contrast the error bounds for our transfer learning estimator with those we obtain on classical low-rank estimators (on just proxy or gold data), illustrating significant gains via transfer learning.

Word Embeddings. Next, we examine the scaling of this bound specifically for word embedding model given in (2) (described in §2.1). Recall that, for word co-occurrence count data, the observation matrices A_i are basis matrices, resulting in a lower signal-to-noise ratio than the more typical gaussian ensemble observation matrices studied in the general low-rank matrix factorization literature. Thus, as discussed in Remarks 2–3, we scale the parameters in the QCC and RWC assumptions by $\frac{1}{d^2}$. However, this is counter-balanced by the fact that we have d^2 more samples in the word embedding setting. In particular, for a corpus of $n+1$ words (n consecutive word pairs), we obtain one observation for each observed word pair and each of d^2 possible basis matrices $A_i = E_{jk}$ with $j,k \in [d]$ (see details in §2.1). This results in $d^2 n$ samples. Put together, we obtain the following result on the error of our transfer learning estimator for our word embedding model:

Corollary 1. *Assume $\mathcal{A}_g$ satisfies $\mathsf{QCC}(U_g^*, \frac{\kappa}{d^2})$ and 1-smoothness$(\frac{\beta_g}{d^2})$, and $\mathcal{A}_p$ satisfies r-$\mathsf{RWC}(\frac{\alpha_p}{d^2}, \frac{\beta_p}{d^2})$. Let*

$$\lambda = \max\left\{ \sqrt{\frac{2048L^2\beta_g\sigma_g^2}{d^4 n_g}\log(\frac{10d^2}{\delta})}, \sqrt{\frac{256\beta_g\sigma_g^2\sigma_1^2(U_g^*)}{d^4 n_g}\left(r + 2\sqrt{r\log(\frac{5d}{\delta})} + 2\log(\frac{5d}{\delta})\right)} \right\}.$$

Suppose n_p and d are such that $\frac{L\sigma_r(U_p^*)(3\alpha_p - 2\beta_p)}{8\sqrt{d}} \geq \sqrt{\frac{8\beta_p\sigma_p^2}{n_p}(2r(2d+1)\log(36\sqrt{2}) + \log(\frac{10}{\delta}))}$. Then, with probability at least $1-\delta$, the estimate $\widehat{U}_g^{TL}$ of problem (2) satisfies

$$\ell(\widehat{U}_g^{TL}, U_g^*) = \mathcal{O}\left(\sqrt{\frac{\sigma_g^2 s^2 (r + \log(\frac{d^2}{\delta}))}{n_g}} + \sqrt{\frac{\sigma_p^2(rd^2 + d\log(\frac{1}{\delta}))}{n_p}} \right).$$

The result follows Theorem 1 directly by appropriately scaling the parameters and noting that we have $d^2 n_g$ gold and $d^2 n_p$ proxy observations. Note that the signal-to-noise ratio and sample sizes counter-balance each other, so the error bound in Corollary 1 is of the same scale as the

general bound we obtained in Theorem 1, despite the different setting/assumptions. Corollary 1 shows that our transfer learning estimator only requires a small amount of domain-specific textual data (i.e., $n_g \gg \log d$) to obtain sufficient accuracy, when our analysis is supported by substantial domain-agnostic data such as Wikipedia text (i.e., $n_p \gg d^2$).

3.3. Local Minima

An important practical consideration is that the nonconvexity of the optimization problem in (3) may result in our algorithm converging to a local rather than global minimum. Characterizing these local minima is important to ensure that our estimator is computationally tractable in practice. The RSC condition in Definition 5 (or equivalently QCC in Definition 4) holds for the global minimum but may not apply to local minima; to that end, Loh and Wainwright (2015) propose an alternative restricted strong convexity condition for nonconvex problems, enabling them to show that the resulting local minima are within statistical precision of the global minimum. We build on this last approach, adapting to our loss function:

$$f(\Delta_U) = \frac{1}{n_g}\|X_g - \mathcal{A}_g((\widehat{U}_p + \Delta_U)(\widehat{U}_p + \Delta_U)^T)\|^2. \tag{5}$$

In particular, we introduce the following weaker restricted strong convexity condition (that we term LRSC) directly to the loss function (5) that is more likely to hold for the optimization landscape of all local minima, yielding non-asymptotic bounds for local minima.

Assumption 3 **(LRSC)**. *The loss function in* (5) *satisfies the following restricted strong convexity with constant* η_1, η_2 *and functions* $\tau_1(n,d,r), \tau_2(n,d,r)$*:*

$$\mathbb{E}_{X_g|\mathcal{A}_g}[\langle \nabla f(\widetilde{\Delta}_U + \Delta) - \nabla f(\widetilde{\Delta}_U), \Delta\rangle] \geq \begin{cases} \eta_1\|\Delta\|_F^2 - \tau_1(n_g,d,r)\|\Delta\|_{2,1}^2, & \forall \|\Delta\|_F \leq \rho, \quad (6a) \\ \eta_2\|\Delta\|_F - \tau_2(n_g,d,r)\|\Delta\|_{2,1}, & \forall \|\Delta\|_F \geq \rho. \quad (6b) \end{cases}$$

for any $\Delta \in \mathbb{R}^{d\times r}$ *and some constant* $\rho > 0$*.*

Our LRSC condition provides a lower bound on the *expected* Hessian of the loss function in (5), conditioned on a fixed design, where the expectation is taken over the randomness of the noise terms. Note that the LRSC condition we propose is weaker than the original RSC condition proposed in Loh and Wainwright (2015), which lower bounds the *realized* Hessian directly. This is because the usual problem formulation is quadratic in the target parameter, and thus the Hessian is a deterministic quantity given a fixed design. In contrast, our transfer learning objective induces a quartic dependence on the target parameter Δ_U, and thus our Hessian is a random variable that depends on the realized noise terms, introducing additional complexity.

Intuitively, the LRSC condition serves a similar function as our earlier RSC condition (Definition 5), imposing restricted weak convexity on our loss so that we can recover high-quality estimates

of the gold embeddings U_g^*. We now show that the LRSC (for local minima) is weaker than the RSC (for the global minimum) we used in Theorem 1. Note that LRSC is composed of two separate statements; condition (6a) restricts the geometry locally around the global minimum, and condition (6b) provides a lower bound for parameters that are well-separated from the global minimum. First, the following Proposition 4 shows that condition (6a) is equivalent to the more traditional RSC condition for convex problems (Definition 5) in a neighborhood of the global minimum. Next, Lemmas 8–9 in Loh and Wainwright (2015) show that (6a) usually implies (6b), given that the function $\tau_2(n,d,r)$ in (6b) typically scales as $\mathcal{O}(\sqrt{\tau_1(n,d,r)})$.

PROPOSITION 4. *When $\|\Delta\|_F \le \rho$, (i) for any $\mathcal{A}_g$ that satisfies $\mathsf{RSC}(\sqrt{\frac{2}{3}}U_g^*, \eta, \tau)$ and r-smoothness(β_g) with $9\eta \ge \beta_g$, condition (6a) holds with $\rho \le \sigma_r(U_g^*)/3$, $\eta_1 = 4\eta\sigma_r(U_g^*)^2$ and $\tau_1 = 3\tau/2$; (ii) for any loss function that satisfies condition (6a), $\mathcal{A}_g$ satisfies $\mathsf{RSC}(\sqrt{\frac{2}{3}}U_g^*, \eta, \tau)$ with $\eta = \frac{\eta_1}{2(2\sigma_1(U_g^*)+3\rho/2)^2}$ and $\tau = \tau_1/3$.*

The proof is provided in Appendix C. The following theorem shows that LRSC ensures all local minima are within statistical precision of the true parameter.

THEOREM 2. *Assume LRSC holds for loss function f in (5) and $\mathcal{A}_g$ satisfies 1-smoothness(β_g). Let*

$$\lambda = \max\left\{\sqrt{\frac{32768L^2\beta_g\sigma_g^2}{n_g}\log(\frac{10d^2}{\delta})}, \sqrt{\frac{512\beta_g\sigma_g^2\sigma_1^2(U_g^*)}{n_g}\left(r + 2\sqrt{r\log(\frac{5d}{\delta})} + 2\log(\frac{5d}{\delta})\right)}, \right.$$
$$\left. \frac{4}{3}\tau_2(n_g,d,r), 16L\tau_1(n_g,d,r)\right\}.$$

Suppose n_p and d are such that $\frac{L\sigma_r(U_p^*)(3\alpha_p - 2\beta_p)}{8\sqrt{d}} \ge \sqrt{\frac{8\beta_p\sigma_p^2}{n_p}(2r(2d+1)\log(36\sqrt{2}) + \log(\frac{10}{\delta}))}$, and n_g and d are such that $\lambda \le \rho\eta_2/(8L)$. Then, any local minimum $\widehat{U}_g^{TL}$ satisfies

$$\ell(\widehat{U}_g^{TL}, U_g^*) = \mathcal{O}\left(\sqrt{\frac{\sigma_g^2 s^2(r + \log(\frac{d^2}{\delta}))}{n_g}} + \sqrt{\frac{\sigma_p^2(rd^2 + d\log(\frac{1}{\delta}))}{n_p}}\right)$$

with probability at least $1-\delta$.

We provide a proof in Appendix C. In particular, the above estimation error bound for all local minima has the same scale as Theorem 1 for the global minimum, ensuring that our estimator is computationally tractable in practice.

3.4. Transfer Learning with GloVe

Our transfer learning approach extends straightforwardly to nonlinear loss functions such as GloVe (Pennington et al. 2014), a state-of-the-art technique often used to construct word embeddings in practice. The original GloVe method solves the following optimization problem:

$$\min_{U_i,V_j,b_i,c_j} \sum_{i,j\in[d]} f(Y_{ij})(\log(Y_{ij}) - (U_i V_j^T + b_i + c_j))^2, \tag{7}$$

where d is the number of unique words, Y_{ij} is the total number of co-occurrences of word pair (i,j), and $\{U_i\}_{i\in[d]}$ and $\{V_j\}_{j\in[d]}$ are two sets of word embeddings (one typically takes the sum of the two $U_i + V_i$ as the final word embedding for word i in a post-processing step). $\{b_i\}$ and $\{c_j\} \in \mathbb{R}$ are bias terms (tuning parameters) designed to improve fit. Finally, $f(x)$ is a non-decreasing weighting function defined as

$$f(x) = \begin{cases} (x/x_{\max})^\alpha, & \text{if } \ x < x_{\max}, \\ 1, & \text{otherwise.} \end{cases}$$

Pennington et al. (2014) set the tuning parameters above to be $x_{\max} = 100$ and $\alpha = 3/4$.

We first show that our model (2) includes a linear version of GloVe as a special case. Define the index set $I_{jk} = \{i \in [d^2 n] \mid A_i = E_{jk}\}$, where remember E_{jk} is a basis matrix with entry (j,k) being 1 and 0 otherwise. Taking the average of (2) over the set I_{jk}, we have

$$\frac{1}{|I_{jk}|}\sum_{i\in I_{jk}} X_i = \left\langle \frac{1}{|I_{jk}|}\sum_{i\in I_{jk}} A_i, \Theta^* \right\rangle + \frac{1}{|I_{jk}|}\sum_{i\in I_{jk}} \epsilon_i = \Theta^{*(j,k)} + \frac{1}{|I_{jk}|}\sum_{i\in I_{jk}} \epsilon_i.$$

In other words, we can create a sample word co-occurrence matrix as an empirical estimate of Θ^*; factorizing this provides an estimate of U^*. GloVe then deviates from our linear model by taking the logarithm of $Y_{jk} = \sum_{i\in I_{jk}} X_i$, adding bias terms for extra model complexity, and weighting up frequent word pairs through f. Moreover, it implements alternating-minimization with asymmetric factorization to speed up optimization; recall that GloVe takes the sum $U_i + V_i$ to obtain the word embedding for word i. To leverage our transfer learning approach, we can simply add an analogous group LASSO penalty to this objective:

$$\min_{U_i,V_j,b_i,c_j} \sum_{i,j\in[d]} f(Y_{ij})(\log(Y_{ij}) - (U_i V_j^T + b_i + c_j))^2 + \lambda \sum_{i\in[d]} \|(U^i + V^i) - \widehat{U}_p^i\|, \tag{8}$$

where $\widehat{U}_p$ is a matrix of pre-trained (proxy) word embeddings. We also evaluate this approach empirically in §5.2 on real datasets.

4. Comparing Error Bounds

In this section, we assess the value of transfer learning by comparing to the bounds we obtain if we trained our embeddings on only gold or proxy data.

4.1. Gold Estimator

A natural unbiased approach to learning domain-specific embeddings U_g^* is to apply the Burer-Monteiro approach to only gold data:

$$\widehat{U}_g = \arg\min_{U_g} \frac{1}{n_g} \|X_g - \mathcal{A}_g(U_g U_g^T)\|^2. \tag{9}$$

We follow the approach of Ge et al. (2017) to obtain error bounds on $\widehat{U}_g$ under the following standard regularity assumption:

ASSUMPTION 4. *The gold linear operator $\mathcal{A}_g$ satisfies $2r$-RWC(α_g, β_g).*

Note that Assumption 4 may not hold in our regime of interest where $n_g \ll d$. As discussed in §2.3, in general, we need $n \gtrsim dr$ observations to satisfy RWC, and so the gold estimator may not satisfy any nontrivial guarantees under our data-scarce setting. In contrast, our QCC (Assumption 2) is mild and holds in the high-dimensional setting when $n_g \gg \log(d)$. For the purposes of comparison, we examine the conventional error bounds for problem (9) under RWC.

THEOREM 3. *The estimation error of the gold estimator has*

$$\begin{aligned}\ell(\widehat{U}_g, U_g^*) &\le C_4 \sqrt{\frac{\sigma_g^2 d(2r(2d+1)\log(36\sqrt{2}) + \log(\frac{2}{\delta}))}{n_g}} \\ &= \mathcal{O}\left(\sqrt{\frac{\sigma_g^2(rd^2 + d\log(\frac{1}{\delta}))}{n_g}}\right)\end{aligned}$$

with probability at least $1-\delta$, where $C_4 = \frac{16\sqrt{2\beta_g}}{(3\alpha_g - 2\beta_g)\sigma_r(U_g^*)}$.

We give a proof in Appendix D. Theorem 3 shows that when we have sufficient gold samples (i.e., $n_g \gg d^2$), the gold estimator achieves estimation error scaling as $\mathcal{O}(\sqrt{d^2/n_g})$. However, when $n_g \lesssim d^2$, the gold estimator has very high variance, resulting in substantial estimation error.

Next we apply this result to our word embedding model as described in §3.2.

COROLLARY 2. *Assume $\mathcal{A}_g$ satisfies r-RWC$(\frac{\alpha_g}{d^2}, \frac{\beta_g}{d^2})$. Then, with probability at least $1-\delta$, the estimation error of the gold estimator of problem (2) satisfies*

$$\ell(\widehat{U}_g, U_g^*) = \mathcal{O}\left(\sqrt{\frac{\sigma_g^2(rd^2 + d\log(\frac{1}{\delta}))}{n_g}}\right).$$

This result directly follows Theorem 3 by scaling the parameters in our assumptions by $1/d^2$ and expanding the number of samples to $d^2 n_g$ (see analogous discussion under Corollary 1). Indeed, Corollary 2 shares a similar scaling and insight as Theorem 3.

4.2. Proxy Estimator

An alternative approach is to estimate domain-agnostic word embeddings U_p^* from the proxy data, and ignore the domain-specific bias Δ_U^*:

$$\widehat{U}_p = \arg\min_{U_p} \frac{1}{n_p} \|X_p - \mathcal{A}_p(U_p U_p^T)\|^2. \tag{10}$$

This corresponds to the common practice of using pre-trained word embeddings. Recall that we have already made the RWC assumption for $\mathcal{A}_p$ in Assumption 1.

THEOREM 4. *The estimation error of the proxy estimator has*

$$\begin{aligned}\ell(\widehat{U}_p, U_g^*) &\le \|\Delta_U^*\|_{2,1} + \omega + C_5 \sqrt{\frac{\sigma_p^2 d(2r(2d+1)\log(36\sqrt{2}) + \log(\frac{2}{\delta}))}{n_p}} \\ &= \mathcal{O}\left(\|\Delta_U^*\|_{2,1} + \omega + \sqrt{\frac{\sigma_p^2(rd^2 + d\log(\frac{1}{\delta}))}{n_p}} \right)\end{aligned}$$

with probability at least $1-\delta$, *where* $\omega = \|U_p^*(R_{(\widehat{U}_p, U_p^*)} - R_{(\widehat{U}_p, U_g^*)})\|_{2,1}$ *and* $C_5 = \frac{16\sqrt{2\beta_p}}{(3\alpha_p - 2\beta_p)\sigma_r(U_p^*)}$.

We give a proof in Appendix E, following the approach of Ge et al. (2017). However, as discussed in §2.1, recall that U^* is only identifiable only up to an orthogonal change-of-basis, so we consider the rotation $R_{(\widehat{U}, U^*)}$ that best aligns $\widehat{U}$ with the true parameter U^*. Therefore, to compare $\widehat{U}_p$ with the true gold word embeddings U_g^*, we use the rotation $R_{(\widehat{U}_p, U_g^*)}$. Yet, $\widehat{U}_p$ is best aligned with U_p^* under a different rotation $R_{(\widehat{U}_p, U_p^*)}$. The choice of rotation affects the error from the group-sparse bias term $\Delta_U^* = U_g^* - U_p^*$, resulting in a term ω accounting for the misalignment between the two rotations $R_{(\widehat{U}_p, U_g^*)}$ and $R_{(\widehat{U}_p, U_p^*)}$ in Theorem 4.

Since we n_p is large in our regime of interest, the third term in the estimation error bound (capturing the error of $\widehat{U}_p - U_p^*$) is small, scaling as $\mathcal{O}(d/\sqrt{n_p})$. Instead, the first two terms capturing the bias between U_p^* and U_g^* dominate the estimation error. Note that when $\Delta_U^* \to 0$, we have $R_{(\widehat{U}_p, U_g^*)} \to R_{(\widehat{U}_p, U_p^*)}$. Thus, when there are few domain-specific differences between the gold and proxy data, the proxy estimator can be more accurate than the gold estimator.

Next we apply this result to our word embedding model as described in §3.2.

COROLLARY 3. *Assume* $\mathcal{A}_p$ *satisfies* r-RWC$(\frac{\alpha_p}{d^2}, \frac{\beta_p}{d^2})$. *Then, with probability at least* $1-\delta$, *using* ω *specified in Theorem 4, the estimation error of the proxy estimator of problem (2) satisfies*

$$\ell(\widehat{U}_p, U_g^*) = \mathcal{O}\left(\|\Delta_U^*\|_{2,1} + \omega + \sqrt{\frac{\sigma_p^2(rd^2 + d\log(\frac{1}{\delta}))}{n_p}} \right).$$

This result directly follows Theorem 4 by scaling the parameters in our assumptions by $1/d^2$ and expanding the number of samples to $d^2 n_p$ (see analogous discussion under Corollary 1). Indeed, Corollary 3 shares a similar scaling and insight as Theorem 4.

Estimator	TL	Gold	Proxy
Error Bound	$\mathcal{O}\left(\sqrt{\frac{s^2 r}{n_g}} + \sqrt{\frac{rd^2}{n_p}}\right)$	$\mathcal{O}\left(\sqrt{\frac{rd^2}{n_g}}\right)$	$\mathcal{O}\left(\|\Delta_U^*\|_{2,1} + \omega + \sqrt{\frac{rd^2}{n_p}}\right)$

Table 4.1 **Error bound for the transfer learning, gold and proxy estimators. ω is defined in Theorem 4. The error bounds for word embeddings are the same.**

4.3. Comparison of Error Bounds

We now summarize and compare the estimation error bounds we have derived so far in Table 4.1. We first consider the general low-rank matrix factorization environment. In the regime of interest—i.e., lots of proxy data ($n_p \gg d^2$) but limited gold data ($n_g \ll d^2$)—the upper bound of our transfer learning estimator is much smaller than the conventional scaling of error bounds applied to the gold or proxy data alone. In particular, when our n_p is sufficiently large, our error bound scales as $\sqrt{\log d / n_g}$ whereas the gold error bound scales as $\sqrt{d^2/n_g}$, i.e., transfer learning yields a significant improvement in the vocabulary size d (recall that $s, r \ll d$). On the other hand, the proxy error bound is dominated by the size of the domain bias term $\|\Delta_U^*\|_{2,1}$, implying that it never recovers the true gold word embeddings U_g^*. In contrast, transfer learning can leverage limited gold data to efficiently estimate U_g^* by recovering the bias between U_g^* and U_p^* based on a sufficiently good estimate of U_p^*. Note that Corollary 1-3 for the word embedding model yield the same error bounds, and thus Table 4.1 applies to the word embedding setting as well.

5. Experiments

We evaluate our approach on synthetic data and real Wikipedia data. On real data, we also compare our transfer learning estimator with a state-of-the-art fine-tuning heuristic Mittens (Dingwall and Potts 2018) to identify domain-specific words. We find that a significant drawback of fine-tuning heuristics is that they are relatively uninterpretable, in addition to providing no theoretical guarantees. We primarily present our main results here and relegate experimental details to Appendix G; we also provide additional robustness checks and experimental support in Appendix H.

5.1. Synthetic Data

We first generate synthetic data satisfying our problem formulation—in particular, we have abundant proxy data ($n_p = 5000$), limited gold data ($n_g = 50$), and a sparse number $s \ll d$ of words with altered embeddings. We consider various values of the vocabulary size d, sparsity s, and rank r. Recall that our estimates of the embeddings U_g^* are invariant to rotations (see discussion in §2.1), so we evaluate the estimation error of Θ_g^* using the rotation-invariant Frobenius norm. Figure 3 shows the Frobenius error of our transfer learning estimator as well as the classical low-rank estimators using gold (§4.1) or proxy (§4.2) data alone. Details on data generation and hyperparameter selection (through cross-validation) are provided in Appendix G.1.

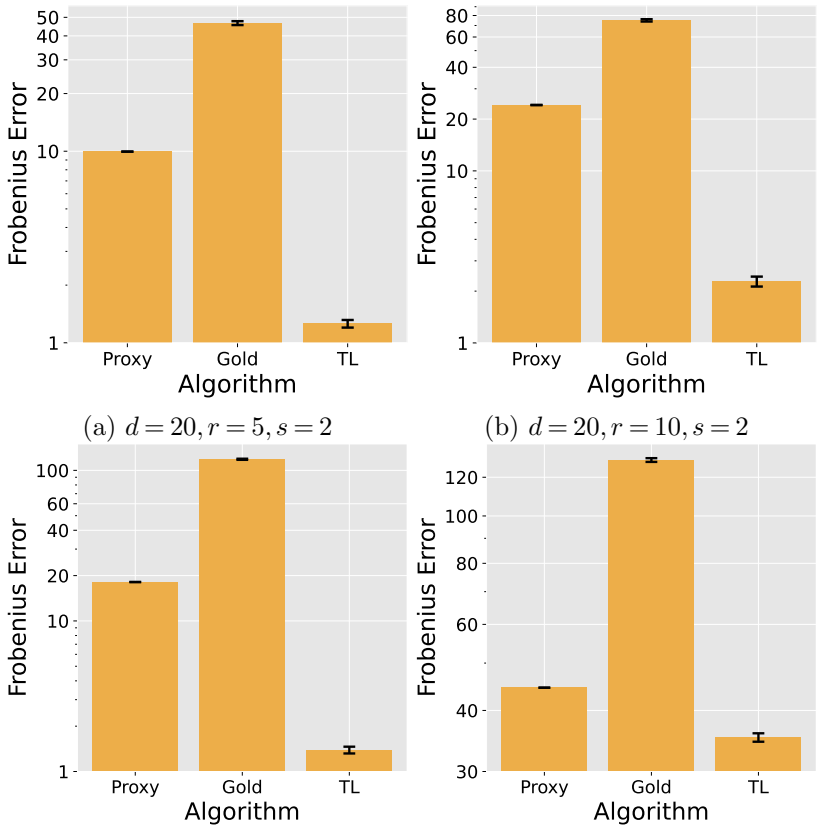


Figure 3 **Bars depict Frobenius norm estimation errors of Θ_g averaged over 100 trials, with error bars the corresponding 95% confidence intervals. 'TL' represents our transfer learning estimator.**

Matching our theoretical results in Table 4.1, we find that our transfer learning estimator substantially outperforms the gold and proxy estimators by exploiting group-sparse structure, efficiently debiasing the proxy data with very limited gold data. The gold estimator generally performs poorly in the low-data regime, and its accuracy deteriorates with increasing model complexity (i.e., larger d and r) as suggested by Theorem 3. The proxy estimator generally performs better than the gold estimator due to its large sample size (reflecting its popularity in practice) but performs worse than our transfer learning estimator, particularly when the domain-specific bias is large (i.e., larger s) as suggested by Theorem 4.

We conduct a series of additional experiments evaluating our approach in Appendix H.1. At a high level, we find the following. Our transfer learning estimator performs substantially better than the gold estimator for small to moderate gold sample sizes—we improve performance even for moderate $n_g \geq d^2$, and perform comparably for large $n_g \gg d^2$. Next, we find that transfer learning becomes more challenging as the gold and proxy tasks become more heterogeneous, i.e., larger magnitude of s, Δ_U^*. Finally, we test the robustness of our estimator's error to the specification of the hyperparameter λ, which is often unknown and must be estimated via cross-validation on

noisy data. We find that our performance is remarkably stable with changes of even an order of magnitude in λ. Details and results are provided in Appendix H.1.

5.2. Wikipedia

While synthetic data is generated to meet our assumptions, this may not be the case on real data. To this end, we now evaluate performance on real Wikipedia articles (a commonly used source for text data). We cannot directly evaluate the estimation error of our embeddings since we don't have access to ground truth; instead, we aim to identify domain-specific words (i.e., words that have special meaning/usage in the target domain). A significant advantage of our method is that it is more *interpretable*, accurately identifying domain-specific words.

In this experiment, we evaluate our approach on 37 individual domain-specific Wikipedia articles from the following four domains: finance, math, computer science, and politics. The articles selected all have a domain-specific word in their title—e.g., "put" in the article "put option" (in finance), "closed" in "closed set" (in math), "object" in computing, and "left" in "left wing politics" (in politics). We define a word to be a domain-specific word if any of its definitions on Wiktionary is labeled with key words from that domain—i.e., "finance" or "business" for finance, "math", "geometry", "algebra", or "group theory" for math, "computing", "computer" or "programming" for computer science, and "politics" for politics.

We leverage our transfer learning approach using the popular GloVe pre-trained word embeddings as the proxy estimator,[5] and evaluate its performance based on the identification accuracy of domain-specific words for each individual Wikipedia article. We compare our approach with a state-of-the-art fine-tuning heuristic Mittens (Dingwall and Potts 2018), as well as random word selection. Table 5.1 shows the average F_1 score of identifying domain-specific words (normalized by article length) across articles in each domain. While we observe that other approaches also identify domain-specific words, our approach does so more effectively, most likely since our group-sparsity assumption is at least partly supported by these datasets (recall Fig. 1 from the introduction). Table 5.2 shows the top 10 words ranked by our approach and by Mittens for one article in each domain—indeed, we observe that our approach is much more effective at identifying domain-specific words (shown in bold). Details provided in Appendix G.2.

Additionally, we consider a version of our transfer learning estimator adapted to the popular GloVe objective (see Eq. (8) and accompanying discussion in §3.4). As shown in Table 5.3, our transfer learning estimator and its GloVe analog perform comparably, demonstrating that our technical insights carry over naturally to off-the-shelf word embedding approaches.

[5] Note that the goal of our transfer learning approach is to efficiently use publicly available pre-trained word embeddings together with domain textual data. It is especially computationally costly to train word embeddings from the whole Wikipedia dump, so we use the GloVe pre-trained word embedding as our proxy estimator.

Domain	TL	Mittens	Random
Finance	**0.2320**	0.1829	0.1376
Math	**0.2660**	0.2175	0.1543
Computing	**0.2527**	0.1963	0.1430
Politics	**0.1873**	0.1571	0.0640

Table 5.1 Average $F1$ score of domain-specific word identification (normalized by article length) for four domains respectively. "TL" represents our transfer learning approach.

Short		Prime Number		Cloud Computing		Conservatism	
TL	Mittens	TL	Mittens	TL	Mittens	TL	Mittens
short	**short**	**prime**	**prime**	**cloud**	**cloud**	**party**	**party**
shares	percent	**formula**	still	**data**	**private**	**conservative**	**conservative**
price	due	**numbers**	**formula**	**computing**	large	social	second
stock	public	**number**	de	**service**	information	conservatism	social
security	customers	**primes**	**numbers**	**services**	devices	government	research
selling	prices	**theorem**	**number**	**applications**	**applications**	**liberal**	svp
securities	high	**natural**	great	**private**	security	**conservatives**	government
position	hard	integers	side	users	work	political	de
may	**shares**	**theory**	way	use	**engine**	**right**	also
margin	price	**product**	algorithm	**software**	allows	economic	church

Table 5.2 Top 10 words, sorted by absolute change of word embedding from source to target domain. Domain-specific words (threshold set to top 10% of the rank) are labeled in bold.

Domain	TL	GloVeTL
Finance	0.2320	0.2336
Math	0.2660	0.2499
Computing	0.2527	0.2437
Politics	0.1873	0.1817

Table 5.3 Average $F1$ score of domain word identification (normalized by article length) for four domains respectively. "TL" represents our transfer learning approach, and "GloVeTL" represents our method adapted to the GloVe objective.

We conduct a series of additional experiments evaluating our approach in Appendix H.2. In particular, we compare performance with two more recent algorithms from the natural language processing literature that combine domain-specific word embeddings with pre-trained word embeddings: Canonical Correlation Analysis (CCA), and its closely related kernelized variant KCCA (Sarma et al. 2018). Like Mittens, these are also heuristics that do not provide any theoretical guarantees on their performance. We find our estimator outperforms these approaches in the same domain-specific word identification task. Furthermore, we show that this improvement is consistent across different thresholds that determine the criteria for a domain-specific word, illustrating that our results are robust to parameter selection.

6. Conclusions

We propose a novel estimator for transferring knowledge from large text corpora to learn word embeddings in a data-scarce domain of interest. We cast this as a low-rank matrix factorization

problem with a group-sparse penalty, regularizing the domain embeddings towards existing pre-trained embeddings. Under a group-sparsity assumption and standard regularity conditions, we prove that our estimator requires substantially less data to achieve the same error compared to conventional estimators that do not leverage transfer learning. Our experiments demonstrate the effectiveness of our approach in the low-data regime, both on synthetic data and a domain word identification task on Wikipedia articles.

While our focus has been on learning word embeddings, unsupervised matrix factorization models have also been widely applied for recommender systems and causal inference, which may open up new lines of inquiry. For instance, in recommender systems, one could consider a bandit approach that further collects domain-specific data in an online fashion (Kallus and Udell 2020); in causal inference, one could treat counterfactuals as missing data and leverage a factor model (Xiong and Pelger 2019). There has also been significant recent interest in low-rank tensor recovery problems (Goldfarb and Qin 2014, Zhang et al. 2019, Shah and Yu 2019), where one aims to learn to make recommendations across multiple types of outcomes (Farias and Li 2019) or to learn treatment effects across multiple experiments (Agarwal et al. 2020). Our transfer learning approach can be used in conjunction with these methods in order to leverage data from other domains.

Acknowledgments

The authors gratefully acknowledge indispensable financial support from the Wharton Analytics Initiative and the Wharton Dean's Fund.

References

Agarwal, Alekh, Sahand Negahban, Martin J Wainwright. 2012. Noisy matrix decomposition via convex relaxation: Optimal rates in high dimensions. *The Annals of Statistics* 1171–1197.

Agarwal, Anish, Devavrat Shah, Dennis Shen. 2020. Synthetic interventions. *arXiv preprint arXiv:2006.07691* .

Ali, Alnur, Edgar Dobriban, Ryan Tibshirani. 2020. The implicit regularization of stochastic gradient flow for least squares. *International Conference on Machine Learning*. PMLR, 233–244.

Athey, Susan, Mohsen Bayati, Nikolay Doudchenko, Guido Imbens, Khashayar Khosravi. 2021. Matrix completion methods for causal panel data models. *Journal of the American Statistical Association* **116**(536) 1716–1730.

Bastani, Hamsa. 2020. Predicting with proxies: Transfer learning in high dimension. *Management Science* .

Bellstam, Gustaf, Sanjai Bhagat, J Anthony Cookson. 2020. A text-based analysis of corporate innovation. *Management Science* .

Ben-David, Shai, John Blitzer, Koby Crammer, Alex Kulesza, Fernando Pereira, Jennifer Wortman Vaughan. 2010. A theory of learning from different domains. *Machine learning* **79**(1) 151–175.

Ben-David, Shai, John Blitzer, Koby Crammer, Fernando Pereira, et al. 2007. Analysis of representations for domain adaptation. *Advances in neural information processing systems* **19** 137.

Bhojanapalli, Srinadh, Behnam Neyshabur, Nathan Srebro. 2016. Global optimality of local search for low rank matrix recovery. *arXiv preprint arXiv:1605.07221* .

Blitzer, John, Mark Dredze, Fernando Pereira. 2007. Biographies, bollywood, boom-boxes and blenders: Domain adaptation for sentiment classification. *Proceedings of the 45th annual meeting of the association of computational linguistics*. 440–447.

Boucheron, Stéphane, Gábor Lugosi, Pascal Massart. 2013. *Concentration inequalities: A nonasymptotic theory of independence*. Oxford university press.

Bühlmann, Peter, Sara Van De Geer. 2011. *Statistics for high-dimensional data: methods, theory and applications*. Springer Science & Business Media.

Burer, Samuel, Renato DC Monteiro. 2003. A nonlinear programming algorithm for solving semidefinite programs via low-rank factorization. *Mathematical Programming* **95**(2) 329–357.

Cai, Tianxi, T Tony Cai, Anru Zhang. 2016. Structured matrix completion with applications to genomic data integration. *Journal of the American Statistical Association* **111**(514) 621–633.

Candes, Emmanuel J, Yaniv Plan. 2011. Tight oracle inequalities for low-rank matrix recovery from a minimal number of noisy random measurements. *IEEE Transactions on Information Theory* **57**(4) 2342–2359.

Candes, Emmanuel J, Terence Tao. 2005. Decoding by linear programming. *IEEE transactions on information theory* **51**(12) 4203–4215.

Chen, Kun, Hongbo Dong, Kung-Sik Chan. 2013. Reduced rank regression via adaptive nuclear norm penalization. *Biometrika* **100**(4) 901–920.

Chi, Yuejie, Yue M Lu, Yuxin Chen. 2019. Nonconvex optimization meets low-rank matrix factorization: An overview. *IEEE Transactions on Signal Processing* **67**(20) 5239–5269.

Dingwall, Nicholas, Christopher Potts. 2018. Mittens: an extension of glove for learning domain-specialized representations. *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*. 212–217.

Farias, Vivek F, Andrew A Li. 2019. Learning preferences with side information. *Management Science* **65**(7) 3131–3149.

Friedman, Jerome, Trevor Hastie, Robert Tibshirani. 2010. A note on the group lasso and a sparse group lasso. *arXiv preprint arXiv:1001.0736* .

Ganin, Yaroslav, Victor Lempitsky. 2015. Unsupervised domain adaptation by backpropagation. *International conference on machine learning*. PMLR, 1180–1189.

Ge, Rong, Chi Jin, Yi Zheng. 2017. No spurious local minima in nonconvex low rank problems: A unified geometric analysis. *Proceedings of the 34th International Conference on Machine Learning-Volume 70*. JMLR. org, 1233–1242.

Ge, Rong, Jason D Lee, Tengyu Ma. 2016. Matrix completion has no spurious local minimum. *Advances in neural information processing systems* **29**.

Goldfarb, Donald, Zhiwei Qin. 2014. Robust low-rank tensor recovery: Models and algorithms. *SIAM Journal on Matrix Analysis and Applications* **35**(1) 225–253.

Hastie, Trevor, Robert Tibshirani, Jerome H Friedman, Jerome H Friedman. 2009. *The elements of statistical learning: data mining, inference, and prediction*, vol. 2. Springer.

Hsu, Chao-Chun, Shantanu Karnwal, Sendhil Mullainathan, Ziad Obermeyer, Chenhao Tan. 2020. Characterizing the value of information in medical notes. *Findings of the Association for Computational Linguistics: EMNLP 2020*. Association for Computational Linguistics, Online, 2062–2072. doi:10.18653/v1/2020.findings-emnlp.187. URL `https://www.aclweb.org/anthology/2020.findings-emnlp.187`.

Hsu, Daniel, Sham Kakade, Tong Zhang, et al. 2012. A tail inequality for quadratic forms of subgaussian random vectors. *Electronic Communications in Probability* **17**.

Huang, Junzhou, Tong Zhang. 2010. The benefit of group sparsity .

Kallus, Nathan, Madeleine Udell. 2020. Dynamic assortment personalization in high dimensions. *Operations Research* **68**(4) 1020–1037.

Klopp, Olga. 2014. Noisy low-rank matrix completion with general sampling distribution .

Kohavi, Ron, et al. 1995. A study of cross-validation and bootstrap for accuracy estimation and model selection. *Ijcai*, vol. 14. Montreal, Canada, 1137–1145.

Koltchinskii, Vladimir. 2011. *Oracle inequalities in empirical risk minimization and sparse recovery problems: École D'Été de Probabilités de Saint-Flour XXXVIII-2008*, vol. 2033. Springer Science & Business Media.

Levy, Omer, Yoav Goldberg. 2014. Neural word embedding as implicit matrix factorization. *Advances in neural information processing systems* **27** 2177–2185.

Li, Qiuwei, Zhihui Zhu, Gongguo Tang. 2019. The non-convex geometry of low-rank matrix optimization. *Information and Inference: A Journal of the IMA* **8**(1) 51–96.

Li, Quanzhi, Sameena Shah. 2017. Learning stock market sentiment lexicon and sentiment-oriented word vector from stocktwits. *Proceedings of the 21st conference on computational natural language learning (CoNLL 2017)*. 301–310.

Li, Ruoting, Margaret Tobey, Maria Mayorga, Sherrie Caltagirone, Osman Özaltın. 2021. Detecting human trafficking: Automated classification of online customer reviews of massage businesses. *Available at SSRN 3982796* .

Lipton, Zachary, Yu-Xiang Wang, Alexander Smola. 2018. Detecting and correcting for label shift with black box predictors. *International conference on machine learning*. PMLR, 3122–3130.

Liu, Xiao, Param Vir Singh, Kannan Srinivasan. 2016. A structured analysis of unstructured big data by leveraging cloud computing. *Marketing Science* **35**(3) 363–388.

Loh, Po-Ling, Martin J Wainwright. 2015. Regularized m-estimators with nonconvexity: Statistical and algorithmic theory for local optima. *The Journal of Machine Learning Research* **16**(1) 559–616.

Lounici, Karim, Massimiliano Pontil, Sara Van De Geer, Alexandre B Tsybakov, et al. 2011. Oracle inequalities and optimal inference under group sparsity. *The Annals of Statistics* **39**(4) 2164–2204.

Mankad, Shawn, Hyunjeong "Spring" Han, Joel Goh, Srinagesh Gavirneni. 2016. Understanding online hotel reviews through automated text analysis. *Service Science* **8**(2) 124–138.

Mikolov, Tomas, Ilya Sutskever, Kai Chen, Greg S Corrado, Jeff Dean. 2013. Distributed representations of words and phrases and their compositionality. *Advances in neural information processing systems*. 3111–3119.

Negahban, Sahand, Martin J Wainwright. 2011. Estimation of (near) low-rank matrices with noise and high-dimensional scaling. *The Annals of Statistics* 1069–1097.

Negahban, Sahand, Martin J Wainwright. 2012. Restricted strong convexity and weighted matrix completion: Optimal bounds with noise. *The Journal of Machine Learning Research* **13** 1665–1697.

Negahban, Sahand N, Pradeep Ravikumar, Martin J Wainwright, Bin Yu, et al. 2012. A unified framework for high-dimensional analysis of m-estimators with decomposable regularizers. *Statistical science* **27**(4) 538–557.

Netzer, Oded, Ronen Feldman, Jacob Goldenberg, Moshe Fresko. 2012. Mine your own business: Market-structure surveillance through text mining. *Marketing Science* **31**(3) 521–543.

Ong, Charlene Jennifer, Agni Orfanoudaki, Rebecca Zhang, Francois Pierre M Caprasse, Meghan Hutch, Liang Ma, Darian Fard, Oluwafemi Balogun, Matthew I Miller, Margaret Minnig, et al. 2020. Machine learning and natural language processing methods to identify ischemic stroke, acuity and location from radiology reports. *PloS one* **15**(6) e0234908.

Park, Dohyung, Anastasios Kyrillidis, Constantine Carmanis, Sujay Sanghavi. 2017. Non-square matrix sensing without spurious local minima via the burer-monteiro approach. *Artificial Intelligence and Statistics*. PMLR, 65–74.

Pennington, Jeffrey, Richard Socher, Christopher D Manning. 2014. Glove: Global vectors for word representation. *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*. 1532–1543.

Ramchandani, Pia, Hamsa Bastani, Emily Wyatt. 2021. Unmasking human trafficking risk in commercial sex supply chains with machine learning. *Available at SSRN 3866259* .

Raskutti, Garvesh, Martin J Wainwright, Bin Yu. 2010. Restricted eigenvalue properties for correlated gaussian designs. *The Journal of Machine Learning Research* **11** 2241–2259.

Recht, Benjamin, Maryam Fazel, Pablo A Parrilo. 2007. Guaranteed minimum-rank solutions of linear matrix equations via nuclear norm minimization. *arXiv preprint arXiv:0706.4138* .

Rigollet, Philippe. 2015. 18. s997: High dimensional statistics. *Lecture Notes, Cambridge, MA, USA: MIT Open-CourseWare* .

Risch, Julian, Ralf Krestel. 2019. Domain-specific word embeddings for patent classification. *Data Technologies and Applications* .

Roy, Arpita, Youngja Park, SHimei Pan. 2017. Learning domain-specific word embeddings from sparse cybersecurity texts. *arXiv preprint arXiv:1709.07470* .

Sarma, Prathusha K, Yingyu Liang, William A Sethares. 2018. Domain adapted word embeddings for improved sentiment classification. *arXiv preprint arXiv:1805.04576* .

Shah, Devavrat, Christina Lee Yu. 2019. Iterative collaborative filtering for sparse noisy tensor estimation. *2019 IEEE International Symposium on Information Theory (ISIT)*. IEEE, 41–45.

Simon, Noah, Jerome Friedman, Trevor Hastie, Robert Tibshirani. 2013. A sparse-group lasso. *Journal of computational and graphical statistics* **22**(2) 231–245.

Van De Geer, Sara A, Peter Bühlmann. 2009. On the conditions used to prove oracle results for the lasso .

Xiong, Ruoxuan, Markus Pelger. 2019. Large dimensional latent factor modeling with missing observations and applications to causal inference. *arXiv preprint arXiv:1910.08273* .

Yang, Wei, Wei Lu, Vincent Zheng. 2017. A simple regularization-based algorithm for learning cross-domain word embeddings. *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*. 2898–2904.

Zhang, Kun, Bernhard Schölkopf, Krikamol Muandet, Zhikun Wang. 2013. Domain adaptation under target and conditional shift. *International Conference on Machine Learning*. PMLR, 819–827.

Zhang, Xiaoqin, Di Wang, Zhengyuan Zhou, Yi Ma. 2019. Robust low-rank tensor recovery with rectification and alignment. *IEEE Transactions on Pattern Analysis and Machine Intelligence* **43**(1) 238–255.

Appendix A: Quadratic Compatibility Condition

Proof of Proposition 1 The RSC condition gives

$$\frac{1}{n}\|\mathcal{A}(\Delta U^{*T}+U^*\Delta^T+\Delta\Delta^T)\|^2 \geq \eta\|\Delta U^{*T}+U^*\Delta^T+\Delta\Delta^T\|_F^2-\tau(n,d,r)\|\Delta\|_{2,1}^2. \tag{11}$$

We lower bound the first term in inequality (11):

$$\begin{aligned}
\|\Delta U^{*T}+U^*\Delta^T+\Delta\Delta^T\|_F^2 =& \|\Delta U^{*T}+U^*\Delta^T\|_F^2+\|\Delta\Delta^T\|_F^2+4\langle U^*\Delta^T,\Delta\Delta^T\rangle\\
=&4\|\Delta U^{*T}\|_F^2+\|\Delta\Delta^T\|_F^2+4\langle U^*\Delta^T,\Delta\Delta^T\rangle\\
\geq&4\|\Delta U^{*T}\|_F^2+\|\Delta\Delta^T\|_F^2-4|U^*\Delta^T|_\infty|\Delta\Delta^T|_1\\
\geq&4\|\Delta U^{*T}\|_F^2+\|\Delta\Delta^T\|_F^2-4\|U^*\|_{2,\infty}\|\Delta\|_{2,\infty}\|\Delta\|_{2,1}^2\\
\geq&4\|\Delta U^{*T}\|_F^2+\|\Delta\Delta^T\|_F^2-4\frac{D\bar{L}}{\sqrt{d}}\|\Delta\|_{2,1}^2.
\end{aligned}$$

where the second equality uses $\mathsf{tr}(X^2)=\|X\|_F^2$. This gives us

$$\frac{1}{n}\|\mathcal{A}(\Delta U^{*T}+U^*\Delta^T+\Delta\Delta^T)\|^2 \geq 4\eta\|\Delta U^{*T}\|_F^2+\eta\|\Delta\Delta^T\|_F^2-4\frac{\eta D\bar{L}}{\sqrt{d}}\|\Delta\|_{2,1}^2-\tau(n,d,r)\|\Delta\|_{2,1}^2.$$

Under the condition that

$$\sum_{j\in J^c}\|\Delta^j\| \leq 7\sum_{j\in J}\|\Delta^j\|,$$

we can upper bound $\|\Delta\|_{2,1}^2$ with a constant scale of $\|\Delta\|_F^2$:

$$\|\Delta\|_{2,1}^2=(\sum_{j\in J^c}\|\Delta^j\|+\sum_{j\in J}\|\Delta^j\|)^2 \leq (8\sum_{j\in J}\|\Delta^j\|)^2 \leq 64s\|\Delta\|_F^2.$$

Therefore, we have

$$\frac{1}{n}\|\mathcal{A}(\Delta U^{*T}+U^*\Delta^T+\Delta\Delta^T)\|^2 \geq \frac{4\eta\sigma_r(U^*)^2}{s}(\sum_{j\in J}\|\Delta^j\|)^2+\eta\|\Delta\Delta^T\|_F^2-64\left(4\frac{\eta D\bar{L}}{\sqrt{d}}+\tau(n,d,r)\right)(\sum_{j\in J}\|\Delta^j\|)^2.$$

As long as n and d are such that

$$\frac{\eta\sigma_r(U^*)^2}{32s} \geq 4\frac{\eta D\bar{L}}{\sqrt{d}}+\tau(n,d,r),$$

we can derive the quadratic compatibility condition

$$\frac{1}{n}\|\mathcal{A}(\Delta U^{*T}+U^*\Delta^T+\Delta\Delta^T)\|^2 \geq \frac{2\eta\sigma_r(U^*)^2}{s}(\sum_{j\in J}\|\Delta^j\|)^2$$

with $\kappa=2\eta\sigma_r(U^*)^2$. □

Proof of Proposition 2 Our proof is adapted from the proof in Raskutti et al. (2010) and that of Proposition 1 in Negahban and Wainwright (2011). Let $\bar{\mathcal{A}}:\mathbb{R}^{d\times d}\to\mathbb{R}^n$ with $\mathsf{vec}(\bar{A}_i)\sim N(0,I)$. Then, we have by construction $\mathcal{A}(\Theta)=\bar{\mathcal{A}}(T_\Sigma(\Theta))$.

Consider the set $\mathcal{R}(t)=\{\Delta \mid \|T_\Sigma(\Delta U^{*T}+U^*\Delta^T+\Delta\Delta^T)\|_F=b, \|\Delta\|_{2,1}\leq t\}$ for any given $b>0$. We aim to lower bound

$$\inf_{\Delta\in\mathcal{R}(t)}\|\mathcal{A}(\Delta U^{*T}+U^*\Delta^T+\Delta\Delta^T)\| = \inf_{\Delta\in\mathcal{R}(t)}\sup_{u\in S^{n-1}}\langle u,\mathcal{A}(\Delta U^{*T}+U^*\Delta^T+\Delta\Delta^T)\rangle,$$

where S^{n-1} is the $(n-1)$-dimension unit sphere. We define an associated zero-mean gaussian random variable $Z_{u,\Delta} = \langle u, \bar{\mathcal{A}}(T_\Sigma(\Delta U^{*T} + U^*\Delta^T + \Delta\Delta^T))\rangle$. For any pairs (u,Δ) and (u',Δ'), we have

$$\mathbb{E}[(Z_{u,\Delta} - Z_{u',\Delta'})^2] = \|u \otimes T_\Sigma(\Delta U^{*T} + U^*\Delta^T + \Delta\Delta^T) - u' \otimes T_\Sigma(\Delta' U^{*T} + U^*\Delta'^T + \Delta'\Delta'^T)\|_F^2,$$

where $\otimes$ is the Kronecker product. Now consider a second zero-mean gaussian process $Y_{u,\Delta} = b\langle g, u\rangle + \langle G, T_\Sigma(\Delta U^{*T} + U^*\Delta^T + \Delta\Delta^T)\rangle$, where $g \in \mathbb{R}^n$ and $G \in \mathbb{R}^{d\times d}$ have i.i.d. $N(0,1)$ entries. For any pairs (u,Δ) and (u',Δ'), we have

$$\mathbb{E}[(Y_{u,\Delta} - Y_{u',\Delta'})^2] = b^2\|u-u'\|^2 + \|T_\Sigma(\Delta U^{*T} + U^*\Delta^T + \Delta\Delta^T) - T_\Sigma(\Delta' U^{*T} + U^*\Delta'^T + \Delta'\Delta'^T)\|_F^2.$$

As $\|u\| = 1$ and $\|T_\Sigma(\Delta U^{*T} + U^*\Delta^T + \Delta\Delta^T)\|_F = b$, it holds that

$$\begin{aligned}
&\|u \otimes T_\Sigma(\Delta U^{*T} + U^*\Delta^T + \Delta\Delta^T) - u' \otimes T_\Sigma(\Delta' U^{*T} + U^*\Delta'^T + \Delta'\Delta'^T)\|_F^2 \\
&\qquad \le b^2\|u-u'\|^2 + \|T_\Sigma(\Delta U^{*T} + U^*\Delta^T + \Delta\Delta^T) - T_\Sigma(\Delta' U^{*T} + U^*\Delta'^T + \Delta'\Delta'^T)\|_F^2,
\end{aligned}$$

where we use the fact that $\langle X, X - X'\rangle \ge 0$ for any matrix X, X' with $\|X\|_F = \|X'\|_F$. Note that when $\Delta = \Delta'$, the equality holds. Consequently, gaussian comparison inequalities, specifically Gordon's inequality (see, e.g., Raskutti et al. 2010), gives rise to

$$\mathbb{E}\left[\inf_{\Delta\in\mathcal{R}(t)} \sup_{u\in S^{n-1}} Z_{u,\Delta}\right] \ge \mathbb{E}\left[\inf_{\Delta\in\mathcal{R}(t)} \sup_{u\in S^{n-1}} Y_{u,\Delta}\right].$$

The gaussian process $Y_{u,\Delta}$ has

$$\begin{aligned}
\mathbb{E}\left[\inf_{\Delta\in\mathcal{R}(t)} \sup_{u\in S^{n-1}} Y_{u,\Delta}\right] =& \mathbb{E}\left[b \sup_{u\in S^{n-1}} \langle g, u\rangle\right] + \mathbb{E}\left[\inf_{\Delta\in\mathcal{R}(t)} \langle G, T_\Sigma(\Delta U^{*T} + U^*\Delta^T + \Delta\Delta^T)\rangle\right] \\
=& b\mathbb{E}\left[\|g\|\right] - \mathbb{E}\left[\sup_{\Delta\in\mathcal{R}(t)} \langle G, T_\Sigma(\Delta U^{*T} + U^*\Delta^T + \Delta\Delta^T)\rangle\right].
\end{aligned}$$

Using Lemma 10, the first term has $\mathbb{E}\left[\|g\|\right] \ge \sqrt{n}/2$ by calculating an integral of a chi-squared distribution. For the second term,

$$\begin{aligned}
\langle G, T_\Sigma(\Delta U^{*T} + U^*\Delta^T + \Delta\Delta^T)\rangle &= \langle T_\Sigma(G), \Delta U^{*T} + U^*\Delta^T + \Delta\Delta^T\rangle \\
&= \langle T_\Sigma(G)U^*, \Delta\rangle + \langle T_\Sigma(G)^T U^*, \Delta\rangle + \langle T_\Sigma(G), \Delta\Delta^T\rangle \\
&\le (\|T_\Sigma(G)U^*\|_{2,\infty} + \|T_\Sigma(G)^T U^*\|_{2,\infty} + \bar{L}|T_\Sigma(G)|_\infty)\|\Delta\|_{2,1},
\end{aligned}$$

where we use $\|\Delta\|_{2,1} \le \bar{L}$. Note that $\mathsf{vec}(T_\Sigma(G)) \sim N(0,\Sigma)$. Lemma 9 gives that

$$\mathbb{E}[|T_\Sigma(G)|_\infty] \le 2\sqrt{\max_{i\in[d^2]} \Sigma^{(i,i)} \log(\sqrt{2}d)}.$$

Finally, using Lemma 8 - 10 and Jensen's inequality, we have

$$\begin{aligned}
\mathbb{E}[\|T_\Sigma(G)U^*\|_{2,\infty}] &\le \max_{i\in[d]} \sqrt{\mathsf{tr}(U^{*T}\bar{\Sigma}_{ii}U^*)} + \sqrt{2\max_{i\in[d]} \|U^{*T}\bar{\Sigma}_{ii}U^*\| \log d} \\
&\le \sigma_1(U^*) \max_{i\in[d]} \sqrt{\sigma_1(\bar{\Sigma}_{ii})}(\sqrt{r} + \sqrt{2\log d}).
\end{aligned}$$

Similar results hold for $\mathbb{E}[\|T_\Sigma(G)^T U^*\|_{2,\infty}]$. Define Σ' to be such that $\mathsf{vec}(T_\Sigma(G)^T) \sim N(0,\Sigma')$ and hence $\Sigma' = K^{(d,d)}\Sigma K^{(d,d)}$. $K^{(d,d)} \in \mathbb{R}^{d^2\times d^2}$ is a commutation matrix that transform $\mathsf{vec}(X)$ to $\mathsf{vec}(X^T)$ for $X \in \mathbb{R}^{d\times d}$:

$$K^{(d,d)}\,\mathsf{vec}(X) = \mathsf{vec}(X^T).$$

Note that Σ' shares a similar property as Σ as $K^{(d,d)}$ is nonsingular and has only eigenvalues 1 or -1. Therefore, we can obtain

$$\mathbb{E}[\|T_\Sigma(G)U^*\|_{2,\infty} + \|T_\Sigma(G)^T U^*\|_{2,\infty} + \bar{L}|T_\Sigma(G)|_\infty] \le C_6(\sqrt{\log(\sqrt{2}d)} + (\sqrt{r} + \sqrt{2\log d})) \le C_6(2\sqrt{r} + 3\sqrt{\log d}),$$

where $C_6 = 2\bar{L}\sqrt{\max_{i\in[d^2]}\Sigma^{(i,i)}} + \sigma_1(U^*)(\max_{i\in[d]}\sqrt{\sigma_1(\bar{\Sigma}_{ii})} + \max_{i\in[d]}\sqrt{\sigma_1(\bar{\Sigma}'_{ii})})$. Combining all the above gives

$$\mathbb{E}\left[\inf_{\Delta\in\mathcal{R}(t)} \|\bar{\mathcal{A}}(T_\Sigma(\Delta U^{*T} + U^*\Delta^T + \Delta\Delta^T))\|\right] = \mathbb{E}\left[\inf_{\Delta\in\mathcal{R}(t)} \sup_{u\in S^{n-1}} Z_{u,\Delta}\right] \ge \frac{b\sqrt{n}}{2} - C_6(2\sqrt{r} + 3\sqrt{\log d})t.$$

It's easy to show that the function $f(\bar{\mathcal{A}}) := \inf_{\Delta\in\mathcal{R}(t)} \|\bar{\mathcal{A}}(T_\Sigma(\Delta U^{*T} + U^*\Delta^T + \Delta\Delta^T))\|$ is b-Lipschitz. Then, applying Lemma 7 shows that

$$\mathbb{P}\left(\sup_{\Delta\in\mathcal{R}(t)} (\frac{5b}{8} - \frac{\|\mathcal{A}(\Delta U^{*T} + U^*\Delta^T + \Delta\Delta^T)\|}{\sqrt{n}}) \ge \frac{3b}{2}g(t)\right) \le \exp\left(-\frac{ng(t)^2}{8}\right),$$

where $g(t) = \frac{1}{8} + \frac{C_6(2\sqrt{r}+3\sqrt{\log d})t}{b\sqrt{n}}$. By a peeling argument (see, e.g., Lemma 3 in Raskutti et al. 2010), we can derive that

$$\frac{\|\mathcal{A}(\Delta U^{*T} + U^*\Delta^T + \Delta\Delta^T)\|}{\sqrt{n}} \ge \frac{b}{4} - \frac{3C_6(2\sqrt{r} + 3\sqrt{\log d})}{2\sqrt{n}}\|\Delta\|_{2,1}$$

for any $\Delta \in \{\Delta \mid \|T_\Sigma(\Delta U^{*T} + U^*\Delta^T + \Delta\Delta^T)\|_F = b\}$ with probability greater than $1 - c\exp(-c'n)$ for some positive constants c, c'. $\square$

Proof of Proposition 3 Our proof strategy is adapted from the proof of Lemma 14 in Klopp (2014) and Theorem 1 in Negahban and Wainwright (2012).

Consider the following set

$$\mathcal{R}(t) = \{\Delta \mid \|\Delta U^{*T} + U^*\Delta^T + \Delta\Delta^T\|_\infty = b, \frac{\|\Delta U^{*T} + U^*\Delta^T + \Delta\Delta^T\|_\infty}{\|\Delta U^{*T} + U^*\Delta^T + \Delta\Delta^T\|_{L_2(\Pi)}} \le B, \|\Delta\|_{2,1} \le t\}$$

for any given $b > 0$. Let

$$Z_t = \sup_{\Delta\in\mathcal{R}(t)} |\frac{1}{n}\|\mathcal{A}(\Delta U^{*T} + U^*\Delta^T + \Delta\Delta^T)\|^2 - \|\Delta U^{*T} + U^*\Delta^T + \Delta\Delta^T\|^2_{L_2(\Pi)}|.$$

Note that A_i's are basis matrices with only one entry being 1 and others 0; thus,

$$|\langle A_i, \Delta U^{*T} + U^*\Delta^T + \Delta\Delta^T\rangle^2 - \|\Delta U^{*T} + U^*\Delta^T + \Delta\Delta^T\|^2_{L_2(\Pi)}| \le 2b^2.$$

for all $i \in [N]$ and any $\Delta \in \mathcal{R}(t)$. Then, we can use Massart's concentration inequality (see, e.g., Theorem 14.2 in Bühlmann and Van De Geer 2011) to obtain

$$\mathbb{P}(Z_t \ge \mathbb{E}[Z_t] + 2b^2\chi) \le \exp\left(-\frac{n\chi^2}{8}\right) \tag{12}$$

for any $\chi > 0$. Next, we bound the expectation $\mathbb{E}[Z_t]$. Using a standard symmetrization argument (see, e.g., Theorem 2.1 in Koltchinskii 2011), we have

$$\mathbb{E}[Z_t] \leq 2\mathbb{E}\left[\sup_{\Delta \in \mathcal{R}(t)} |\frac{1}{n}\sum_{i\in[n]} \zeta_i \langle A_i, \Delta U^{*T} + U^*\Delta^T + \Delta\Delta^T\rangle^2|\right],$$

where ζ_i's are i.i.d. Rademacher random variables. Since $|\langle A_i, \Delta U^{*T} + U^*\Delta^T + \Delta\Delta^T\rangle| \leq b$ for any $\Delta \in \mathcal{R}(t)$, using the contraction inequality (see, e.g., Koltchinskii 2011) gives

$$\mathbb{E}[Z_t] \leq 8b\mathbb{E}[\sup_{\Delta \in \mathcal{R}(t)} |\langle G, \Delta U^{*T} + U^*\Delta^T + \Delta\Delta^T\rangle|]$$

where $G = \frac{1}{n}\sum_{i\in[n]} \zeta_i A_i$. Now we decompose the term on the right as follows,

$$\begin{aligned}|\langle G, \Delta U^{*T} + U^*\Delta^T + \Delta\Delta^T\rangle| &= |\langle GU^*, \Delta\rangle + \langle G^TU^*, \Delta\rangle + \langle G, \Delta\Delta^T\rangle| \\ &\leq (\|GU^*\|_{2,\infty} + \|G^TU^*\|_{2,\infty} + \bar{L}|G|_\infty)\|\Delta\|_{2,1},\end{aligned}$$

where we use $\|\Delta\|_{2,1} \leq \bar{L}$. Note that each entry (j,k) of the matrix G is $G^{(j,k)} = \frac{1}{n}\sum_{i\in[n]} \zeta_i A_i^{(j,k)}$, where $\zeta_i A_i^{(j,k)}$ has mean 0, variance μ_2/d^2, and upper bound 1. Therefore, using the Bernstein inequality yields

$$\mathbb{P}(|G^{(j,k)}| \geq \chi) \leq 2\exp\left(-\frac{n\chi^2}{\frac{2\mu_2}{d^2} + \frac{2\chi}{3}}\right).$$

With a union bound, we further have

$$\mathbb{P}(\|G\|_\infty \geq \chi) \leq 2d^2\exp\left(-\frac{n\chi^2}{\frac{2\mu_2}{d^2} + \frac{2\chi}{3}}\right).$$

Then, using the proof strategy of Lemma 6 in Klopp (2014), it follows that

$$\mathbb{E}[\|G\|_\infty] \leq (\mathbb{E}[\|G\|_\infty^{2\log d}])^{1/(2\log d)} \leq 11\left(\sqrt{\frac{2\mu_2\log d}{nd^2}} + \frac{4\log d}{3n}\right).$$

Moreover, each row j of GU^* is $G^{(j,\cdot)}U^* = \frac{1}{n}\sum_{i\in[n]} \zeta_i A_i^{(j,\cdot)}U^*$, where $\zeta_i A_i^{(j,\cdot)}U^*$ has mean 0, ℓ_2-norm upper bound $\sigma_1(U^*)$, and

$$\max\left\{\|\frac{1}{n}\sum_{i\in[n]}\mathbb{E}[A_i^{(j,\cdot)}U^*U^{*T}A_i^{(j,\cdot)T}]\|, \|\frac{1}{n}\sum_{i\in[n]}\mathbb{E}[U^{*T}A_i^{(j,\cdot)T}A_i^{(j,\cdot)}U^*]\|\right\} \leq \frac{r\mu_2\sigma_1^2(U^*)}{d^2}.$$

Therefore, using the matrix Bernstein inequality (see Lemma 11) and a union bound, we have

$$\mathbb{P}(\|GU^*\|_{2,\infty} \geq x) \leq 2d^2\exp\left(\frac{-nx^2}{\frac{2r\mu_2\sigma_1^2(U^*)}{d^2} + \frac{2\sigma_1(U^*)x}{3}}\right).$$

Again using the proof strategy of Lemma 6 in Klopp (2014), we have

$$\mathbb{E}[\|GU^*\|_{2,\infty}] \leq 11\sigma_1(U^*)\left(\sqrt{\frac{4r\mu_2\log d}{nd^2}} + \frac{8\log d}{3n}\right).$$

Similarly, we can get a same upper bound of $\mathbb{E}[\|G^TU^*\|_{2,\infty}]$. Combining all the above, we have

$$\mathbb{E}[Z_t] \leq C_7 b\left(\sqrt{\frac{\log d}{nd^2}} + \frac{\log d}{n}\right)t,$$

where $\tilde{C}_7 = 88(\bar{L}(\sqrt{2\mu_2}+4/3)+2\sigma_1(U^*)(\sqrt{4r\mu_2}+8/3))$. Plugging it into (12) and setting $\chi = g(t) = \frac{1}{10B^2} + \frac{\tilde{C}_7(\sqrt{\frac{\log d}{nd^2}}+\frac{\log d}{n})t}{b}$, we derive that

$$\mathbb{P}(Z_t \geq -\frac{b^2}{10B^2} + 3b^2 g(t)) \leq \exp\left(-\frac{ng(t)^2}{8}\right).$$

Again, using the peeling argument (see, e.g., Lemma 3 in Raskutti et al. 2010), it yields that

$$\frac{1}{n}\|\mathcal{A}(\Delta U^{*T} + U^*\Delta^T + \Delta\Delta^T)\|^2 \geq \|\Delta U^{*T} + U^*\Delta^T + \Delta\Delta^T\|^2_{L_2(\Pi)} - \frac{1}{2}b^2 - 6b\tilde{C}_7\left(\sqrt{\frac{\log d}{nd^2}} + \frac{\log d}{n}\right)\|\Delta\|_{2,1},$$

for any $\Delta \in \{\Delta \mid \|\Delta U^{*T} + U^*\Delta^T + \Delta\Delta^T\|_\infty = b, \frac{\|\Delta U^{*T}+U^*\Delta^T+\Delta\Delta^T\|_\infty}{\|\Delta U^{*T}+U^*\Delta^T+\Delta\Delta^T\|_{L_2(\Pi)}} \leq B\}$ with probability greater than $1 - c\exp(-\frac{c'}{B^4}n)$ for some positive constants c, c'. Thus, it implies that

$$\begin{aligned}\frac{1}{n}\|\mathcal{A}(\Delta U^{*T} + U^*\Delta^T + \Delta\Delta^T)\|^2 \geq \frac{1}{2}&\|\Delta U^{*T} + U^*\Delta^T + \Delta\Delta^T\|^2_{L_2(\Pi)} \\ &- 6C_7\left(\sqrt{\frac{\log d}{nd^2}} + \frac{\log d}{n}\right)\|\Delta U^{*T} + U^*\Delta^T + \Delta\Delta^T\|_{L_2(\Pi)}\|\Delta\|_{2,1},\end{aligned}$$

for any $\Delta \in \{\Delta \mid \frac{\|\Delta U^{*T}+U^*\Delta^T+\Delta\Delta^T\|_\infty}{\|\Delta U^{*T}+U^*\Delta^T+\Delta\Delta^T\|_{L_2(\Pi)}} \leq B\}$, where $C_7 = B\tilde{C}_7$.

Note that when $\pi_{j,k} \geq \mu_1/d^2$ for any $j,k \in [d]$, it holds that $\frac{\mu_1}{d^2}\|\Delta U^{*T} + U^*\Delta^T + \Delta\Delta^T\|_F^2 \leq \|\Delta U^{*T} + U^*\Delta^T + \Delta\Delta^T\|^2_{L_2(\Pi)}$. Our result then follows by using the inequality $a^2 + b^2 \geq 2ab$. □

Appendix B: Error Bound of Transfer Learning Estimator

LEMMA 1. *Assume $\mathcal{A}_p$ satisfies $2r$-RWC(α_p, β_p), and $\mathcal{A}_g$ satisfies the quadratic compatibility condition. Let $A_g^{lk} = \begin{bmatrix} A_{g,1}^{(l,k)} & \cdots & A_{g,n_g}^{(l,k)} \end{bmatrix}^T$. Define $\Psi_j, \Phi_j \in \mathbb{R}^{r\times r}$ to be*

$$\Psi_j = U_g^{*T}\frac{A_g^{jT}A_g^j}{n_g}U_g^*, \quad \Phi_j = U_g^{*T}\frac{(A_g^{Tj})^T A_g^{Tj}}{n_g}U_g^*,$$

where $A_g^j, A_g^{Tj} \in \mathbb{R}^{n_g\times d}$ are matrices that stacks up the j^{th} rows of $A_{g,i}$ and $A_{g,i}^T$, $i \in [n_g]$ respectively, i.e.,

$$A_g^j = \begin{bmatrix} A_{g,1}^{(j,\cdot)} \\ A_{g,2}^{(j,\cdot)} \\ \vdots \\ A_{g,n_g}^{(j,\cdot)} \end{bmatrix}, \quad A_g^{Tj} = \begin{bmatrix} A_{g,1}^{T(j,\cdot)} \\ A_{g,2}^{T(j,\cdot)} \\ \vdots \\ A_{g,n_g}^{T(j,\cdot)} \end{bmatrix}.$$

Then, our two-stage transfer learning estimator satisfies with any chosen values of $\lambda > 0$ and $c > 0$

$$\|\widehat{U}_g^{TL} - U_g^*\|_{2,1} \geq 16\left(\frac{\lambda s}{\kappa} + \frac{4\sqrt{d}c}{\sigma_r(U_p^*)(3\alpha_p - 2\beta_p)}\right)$$

with probability at most

$$\begin{aligned}&2(36\sqrt{2})^{2r(2d+1)}\exp\left(-\frac{L^2\sigma_r^2(U_p^*)(3\alpha_p-2\beta_p)^2 n_p}{512\beta_p\sigma_p^2 d}\right) \\ &+ 2d^2\exp\left(-\frac{\lambda^2 n_g}{2048L^2\sigma_g^2(\max_{l,k}\|A_g^{lk}\|^2/n_g)}\right) \\ &+ d\max_{j\in[d]}\exp\left(-\left(\sqrt{\frac{\frac{\lambda^2 n_g}{256\sigma_g^2} - (tr(\Psi_j) - \frac{\|\Psi_j\|_F^2}{2\|\Psi_j\|})}{2\|\Psi_j\|}} - \frac{\|\Psi_j\|_F}{2\|\Psi_j\|}\right)^2\right) \\ &+ d\max_{j\in[d]}\exp\left(-\left(\sqrt{\frac{\frac{\lambda^2 n_g}{256\sigma_g^2} - (tr(\Phi_j) - \frac{\|\Phi_j\|_F^2}{2\|\Phi_j\|})}{2\|\Phi_j\|}} - \frac{\|\Phi_j\|_F}{2\|\Phi_j\|}\right)^2\right) \\ &+ 2(36\sqrt{2})^{2r(2d+1)}\exp\left(-\frac{c^2 n_p}{8\beta_p\sigma_p^2}\right).\end{aligned}$$

Proof of Lemma 1 As problem (4) is equivalent to problem (3), we analyze problem (4) for simplicity.

Note that the row sparsity is immune to rotations, that is, for any orthogonal matrix R, $\Delta_U^* R$ is still row sparse. After our first step of finding the proxy estimator, we align $\widehat{U}_p$ with U_p^* in the direction of $R_{(\widehat{U}_p, U_p^*)}$. By our definition,

$$U_g^* R_{(\widehat{U}_p, U_p^*)} = U_p^* R_{(\widehat{U}_p, U_p^*)} + \Delta_U^* R_{(\widehat{U}_p, U_p^*)}.$$

Through our previous analyses, $\widehat{U}_p$ is close to $U_p^* R_{(\widehat{U}_p, U_p^*)}$ with a high probability. Therefore, in our second step, we aim to find an estimator $\widehat{\Delta}_U$ for $\Delta_U^* R_{(\widehat{U}_p, U_p^*)}$ through $\ell_{2,1}$ penalty. For simplicity, we use U_g^*, U_p^* and Δ_U^* to represent $U_g^* R_{(\widehat{U}_p, U_p^*)}$, $U_p^* R_{(\widehat{U}_p, U_p^*)}$ and $\Delta_U^* R_{(\widehat{U}_p, U_p^*)}$ respectively in the following analyses, which are aligned in the direction of $R_{(\widehat{U}_p, U_p^*)}$. Define the first-stage estimation error $\nu = \widehat{U}_p - U_p^*$ and $\widetilde{\Delta}_U = \Delta_U^* - \nu$. Thus, $U_g^* = U_p^* + \Delta_U^* = \widehat{U}_p + \widetilde{\Delta}_U$. Since $\widehat{U}_p$ carries the estimation error from the first step, the parameter we actually want to recover is $\widetilde{\Delta}_U$, which is approximately row sparse when the proxy data is huge. We define the adjoint of an operator $\mathcal{A} : \mathbb{R}^{d\times d} \to \mathbb{R}^n$ to be $\mathcal{A}^* : \mathbb{R}^n \to \mathbb{R}^{d\times d}$, with $\mathcal{A}^*(\epsilon) = \sum_{i=1}^n \epsilon_i A_i$.

As we search within $\|\Delta_U\|_{2,1} \le 2L$ and $\|\Delta_U^*\|_{2,1} \le L$, we require the following event to hold

$$\mathcal{I} = \{\|\nu\|_{2,1} \le L\} \tag{13}$$

for $\widetilde{\Delta}_U$ to be feasible. Using a similar analysis to Theorem 3, we can show the event $\mathcal{I}$ takes place with a high probability

$$\mathbb{P}(\mathcal{I}) \ge 1 - 2(36\sqrt{2})^{2r(2d+1)} \exp\left(-\frac{L^2 \sigma_r^2(U_p^*)(3\alpha_p - 2\beta_p)^2 n_p}{512\beta\sigma_p^2 d}\right)$$

on the event $\mathcal{I}$, the global optimality of $\widehat{\Delta}_U$ implies

$$\frac{1}{n_g}\|X_g - \mathcal{A}_g((\widehat{U}_p + \widehat{\Delta}_U)(\widehat{U}_p + \widehat{\Delta}_U)^T)\|^2 + \lambda\|\widehat{\Delta}_U\|_{2,1} \le \frac{1}{n_g}\|X_g - \mathcal{A}_g((\widehat{U}_p + \widetilde{\Delta}_U)(\widehat{U}_p + \widetilde{\Delta}_U)^T)\|^2 + \lambda\|\widetilde{\Delta}_U\|_{2,1}.$$

Plugging in $X_g = \mathcal{A}_g((\widehat{U}_p + \widetilde{\Delta}_U)(\widehat{U}_p + \widetilde{\Delta}_U)^T) + \epsilon_g$ yields

$$\begin{aligned}\frac{1}{n_g}\|\mathcal{A}_g((\widehat{U}_p + \widehat{\Delta}_U)(\widehat{U}_p + \widehat{\Delta}_U)^T - (\widehat{U}_p + \widetilde{\Delta}_U)(\widehat{U}_p + \widetilde{\Delta}_U)^T)\|^2 + \lambda\|\widehat{\Delta}_U\|_{2,1} \\ \le \frac{2}{n_g}\langle \epsilon_g, \mathcal{A}_g((\widehat{U}_p + \widehat{\Delta}_U)(\widehat{U}_p + \widehat{\Delta}_U)^T - (\widehat{U}_p + \widetilde{\Delta}_U)(\widehat{U}_p + \widetilde{\Delta}_U)^T)\rangle + \lambda\|\widetilde{\Delta}_U\|_{2,1}.\end{aligned}$$

Rearranging the RHS with $U_g^* = \widehat{U}_p + \widetilde{\Delta}_U$, we get

$$\begin{aligned}\frac{1}{n_g}\|\mathcal{A}_g((\widehat{U}_p + \widehat{\Delta}_U)(\widehat{U}_p + \widehat{\Delta}_U)^T - (\widehat{U}_p + \widetilde{\Delta}_U)(\widehat{U}_p + \widetilde{\Delta}_U)^T)\|^2 + \lambda\|\widehat{\Delta}_U\|_{2,1} \\ \le \frac{2}{n_g}\langle \epsilon_g, \mathcal{A}_g((\widehat{\Delta}_U - \widetilde{\Delta}_U)U_g^{*T} + U_g^*(\widehat{\Delta}_U - \widetilde{\Delta}_U)^T + (\widehat{\Delta}_U - \widetilde{\Delta}_U)(\widehat{\Delta}_U - \widetilde{\Delta}_U)^T)\rangle + \lambda\|\widetilde{\Delta}_U\|_{2,1} \quad (14)\end{aligned}$$

The first part of the first term on the RHS of inequality (14) has

$$\begin{aligned}\langle \epsilon_g, \mathcal{A}_g((\widehat{\Delta}_U - \widetilde{\Delta}_U)U_g^{*T} + U_g^*(\widehat{\Delta}_U - \widetilde{\Delta}_U)^T)\rangle =& \langle \mathcal{A}_g^*(\epsilon_g), (\widehat{\Delta}_U - \widetilde{\Delta}_U)U_g^{*T} + U_g^*(\widehat{\Delta}_U - \widetilde{\Delta}_U)^T\rangle \\ =& \langle \mathcal{A}_g^*(\epsilon_g)U_g^*, \widehat{\Delta}_U - \widetilde{\Delta}_U\rangle + \langle \mathcal{A}_g^*(\epsilon_g)^T U_g^*, \widehat{\Delta}_U - \widetilde{\Delta}_U\rangle \\ \le& (\max_{j\in[d]}\|(\mathcal{A}_g^*(\epsilon_g)^j U_g^*\| + \max_{j\in[d]}\|\mathcal{A}_g^*(\epsilon_g)^{Tj} U_g^*\|)\|\widehat{\Delta}_U - \widetilde{\Delta}_U\|_{2,1}.\end{aligned}$$

Correspondingly, the second part of the first term on the RHS of inequality (14) has

$$\begin{aligned}\langle \epsilon_g, \mathcal{A}_g((\widehat{\Delta}_U - \widetilde{\Delta}_U)(\widehat{\Delta}_U - \widetilde{\Delta}_U)^T)\rangle =& \langle \mathcal{A}_g^*(\epsilon_g), (\widehat{\Delta}_U - \widetilde{\Delta}_U)(\widehat{\Delta}_U - \widetilde{\Delta}_U)^T\rangle \\ \leq& |\mathcal{A}_g^*(\epsilon_g)|_\infty |(\widehat{\Delta}_U - \widetilde{\Delta}_U)(\widehat{\Delta}_U - \widetilde{\Delta}_U)^T|_1 \\ \leq& |\mathcal{A}_g^*(\epsilon_g)|_\infty \|\widehat{\Delta}_U - \widetilde{\Delta}_U\|_{2,1}^2. \end{aligned} \tag{15}$$

Next, consider the following events

$$\mathcal{G}_1 = \left\{ \frac{2}{n_g} \max_{j\in[d]} \|\mathcal{A}_g^*(\epsilon_g)^j U_g^*\| \leq \frac{\lambda}{8} \right\}, \quad \mathcal{G}_2 = \left\{ \frac{2}{n_g} \max_{j\in[d]} \|\mathcal{A}_g^*(\epsilon_g)^{Tj} U_g^*\| \leq \frac{\lambda}{8} \right\},$$

and

$$\mathcal{F} = \left\{ \frac{2}{n_g} |\mathcal{A}_g^*(\epsilon_g)|_\infty \leq \frac{\lambda}{16L} \right\},$$

which we prove holds with high probability in Lemma 2 after this lemma. On the events $\mathcal{G}_1$, $\mathcal{G}_2$ and $\mathcal{F}$, we derive from inequality (14) that

$$\begin{aligned}&\frac{1}{n_g}\|\mathcal{A}_g((\widehat{U}_p + \widehat{\Delta}_U)(\widehat{U}_p + \widehat{\Delta}_U)^T - (\widehat{U}_p + \widetilde{\Delta}_U)(\widehat{U}_p + \widetilde{\Delta}_U)^T)\|^2 + \lambda\|\widehat{\Delta}_U\|_{2,1} \\ &\qquad \leq \frac{\lambda}{4}\|\widehat{\Delta}_U - \widetilde{\Delta}_U\|_{2,1} + \frac{\lambda}{16L}\|\widehat{\Delta}_U - \widetilde{\Delta}_U\|_{2,1}^2 + \lambda\|\widetilde{\Delta}_U\|_{2,1} \\ &\qquad \leq \frac{\lambda}{2}\|\widehat{\Delta}_U - \widetilde{\Delta}_U\|_{2,1} + \lambda\|\widetilde{\Delta}_U\|_{2,1}.\end{aligned}$$

The second inequality uses

$$\|\widehat{\Delta}_U - \widetilde{\Delta}_U\|_{2,1} \leq 4L,$$

which is derived from the definition of the search region $\|\Delta_U\|_{2,1} \leq 2L$, the definition of event $\mathcal{I}$, and the feasibility of Δ_U^* that $\|\Delta_U^*\|_{2,1} \leq L$. We can further arrange the inequality to get

$$\begin{aligned}&\frac{1}{n_g}\|\mathcal{A}_g((\widehat{U}_p + \widehat{\Delta}_U)(\widehat{U}_p + \widehat{\Delta}_U)^T - (\widehat{U}_p + \widetilde{\Delta}_U)(\widehat{U}_p + \widetilde{\Delta}_U)^T)\|^2 + \frac{\lambda}{2}\sum_{j\in J^c}\|(\widehat{\Delta}_U - \widetilde{\Delta}_U)^j\| \\ &\qquad \leq \frac{3\lambda}{2}\sum_{j\in J}\|(\widehat{\Delta}_U - \widetilde{\Delta}_U)^j\| + 2\lambda\sum_{j\in J^c}\|\nu^j\|. \end{aligned} \tag{16}$$

Now consider the following two cases respectively:

(i). $\sum_{j\in J^c}\|\nu^j\| \leq \sum_{j\in J}\|(\widehat{\Delta}_U - \widetilde{\Delta}_U)^j\|$,

(ii). $\sum_{j\in J^c}\|\nu^j\| > \sum_{j\in J}\|(\widehat{\Delta}_U - \widetilde{\Delta}_U)^j\|$.

Under Case (i), we derive from the inequality (16) that

$$\frac{1}{n_g}\|\mathcal{A}_g((\widehat{U}_p + \widehat{\Delta}_U)(\widehat{U}_p + \widehat{\Delta}_U)^T - (\widehat{U}_p + \widetilde{\Delta}_U)(\widehat{U}_p + \widetilde{\Delta}_U)^T)\|^2 + \frac{\lambda}{2}\sum_{j\in J^c}\|(\widehat{\Delta}_U - \widetilde{\Delta}_U)^j\| \leq \frac{7\lambda}{2}\sum_{j\in J}\|(\widehat{\Delta}_U - \widetilde{\Delta}_U)^j\|.$$

Thus, we have $\sum_{j\in J^c}\|(\widehat{\Delta}_U - \widetilde{\Delta}_U)^j\| \leq 7\sum_{j\in J}\|(\widehat{\Delta}_U - \widetilde{\Delta}_U)^j\|$ and $\mathcal{A}_g$ satisfies QCC. Further write the above as

$$\frac{1}{n_g}\|\mathcal{A}_g((\widehat{U}_p + \widehat{\Delta}_U)(\widehat{U}_p + \widehat{\Delta}_U)^T - (\widehat{U}_p + \widetilde{\Delta}_U)(\widehat{U}_p + \widetilde{\Delta}_U)^T)\|^2 + \frac{\lambda}{2}\|\widehat{\Delta}_U - \widetilde{\Delta}_U\|_{2,1} \leq \frac{8\lambda^2 s}{\kappa} + \frac{\kappa}{2s}(\sum_{j\in J}\|(\widehat{\Delta}_U - \widetilde{\Delta}_U)^j\|)^2,$$

where we use the inequality $2ab \le a^2 + b^2$. Apply QCC to the RHS, and

$$\frac{1}{2n_g}\|\mathcal{A}_g((\widehat{U}_p+\widehat{\Delta}_U)(\widehat{U}_p+\widehat{\Delta}_U)^T-(\widehat{U}_p+\widetilde{\Delta}_U)(\widehat{U}_p+\widetilde{\Delta}_U)^T)\|^2+\frac{\lambda}{2}\|\widehat{\Delta}_U-\widetilde{\Delta}_U\|_{2,1}\le\frac{8\lambda^2 s}{\kappa}$$

Under Case (ii), the inequality (16) gives

$$\frac{1}{n_g}\|\mathcal{A}((\widehat{U}_p+\widehat{\Delta}_U)(\widehat{U}_p+\widehat{\Delta}_U)^T-(\widehat{U}_p+\widetilde{\Delta}_U)(\widehat{U}_p+\widetilde{\Delta}_U)^T)\|^2+\frac{\lambda}{2}\|\widehat{\Delta}_U-\widetilde{\Delta}_U\|_{2,1}\le 4\lambda\sum_{j\in J^c}\|\nu^j\|.$$

Therefore, under any circumstances, we have

$$\|\widehat{\Delta}_U-\widetilde{\Delta}_U\|_{2,1}\le 8(\frac{2\lambda s}{\kappa}+\sum_{j\in J^c}\|\nu^j\|)\le 8(\frac{2\lambda s}{\kappa}+\|\nu\|_{2,1}).$$

Consider the event

$$\mathcal{J}=\left\{\|\nu\|_{2,1}\le\frac{8\sqrt{d}c}{(3\alpha_p-2\beta_p)\sigma_r(U_p^*)}\right\}. \tag{17}$$

Using a similar analysis to Theorem 3 as our analysis on event $\mathcal{I}$, we have

$$\mathbb{P}(\mathcal{J})\ge 1-2(36\sqrt{2})^{2r(2d+1)}\exp(-\frac{c^2 n_p}{8\beta_p\sigma_p^2}).$$

Therefore, on the event $\mathcal{J}$, the estimation error is bounded by

$$\|\widehat{\Delta}_U-\widetilde{\Delta}_U\|_{2,1}\le 16(\frac{\lambda s}{\kappa}+\frac{4\sqrt{d}c}{(3\alpha_p-2\beta_p)\sigma_r(U_p^*)}).$$

Combining all the above and using Lemma 2, we have the following concentration inequality

$$\begin{aligned}
&\mathbb{P}\left(\|\widehat{\Delta}_U-\widetilde{\Delta}_U\|_{2,1}\ge 16(\frac{\lambda s}{\kappa}+\frac{4\sqrt{d}c}{(3\alpha_p-2\beta_p)\sigma_r(U_p^*)})\right)\\
&\qquad\le\mathbb{P}(\mathcal{I}^c)+\mathbb{P}(\mathcal{F}^c)+\mathbb{P}(\mathcal{G}_1^c)+\mathbb{P}(\mathcal{G}_2^c)+\mathbb{P}(\mathcal{J}^c)\\
&\qquad\le 2(36\sqrt{2})^{2r(2d+1)}\exp(-\frac{L^2\sigma_r^2(U_p^*)(3\alpha_p-2\beta_p)^2 n_p}{512\beta_p\sigma_p^2 d})\\
&\qquad\quad+2d^2\exp\left(-\frac{\lambda^2 n_g}{2048L^2\sigma_g^2(\max_{l,k}\|A_g^{lk}\|^2/n_g)}\right)\\
&\quad+d\max_{j\in[d]}\exp\left(-(\sqrt{\frac{\frac{\lambda^2 n_g}{256\sigma_g^2}-(\mathsf{tr}(\Psi_j)-\frac{\|\Psi_j\|_F^2}{2\|\Psi_j\|})}{2\|\Psi_j\|}}-\frac{\|\Psi_j\|_F}{2\|\Psi_j\|})^2\right)\\
&\quad+d\max_{j\in[d]}\exp\left(-(\sqrt{\frac{\frac{\lambda^2 n_g}{256\sigma_g^2}-(\mathsf{tr}(\Phi_j)-\frac{\|\Phi_j\|_F^2}{2\|\Phi_j\|})}{2\|\Phi_j\|}}-\frac{\|\Phi_j\|_F}{2\|\Phi_j\|})^2\right)\\
&\qquad\qquad+2(36\sqrt{2})^{2r(2d+1)}\exp(-\frac{c^2 n_p}{8\beta_p\sigma_p^2}).\quad\square
\end{aligned}\tag{18}$$

LEMMA 2. *The events $\mathcal{G}_1$, $\mathcal{G}_2$ and $\mathcal{F}$ satisfy the following concentration inequalities*

$$\mathbb{P}(\mathcal{G}_1^c)\le d\max_{j\in[d]}\exp\left(-(\sqrt{\frac{\frac{\lambda^2 n_g}{256\sigma_g^2}-(\mathit{tr}(\Psi_j)-\frac{\|\Psi_j\|_F^2}{2\|\Psi_j\|})}{2\|\Psi_j\|}}-\frac{\|\Psi_j\|_F}{2\|\Psi_j\|})^2\right),$$

$$\mathbb{P}(\mathcal{G}_2^c) \leq d \max_{j\in[d]} \exp\left(-(\sqrt{\frac{\frac{\lambda^2 n_g^2}{256\sigma_g^2} - (\mathsf{tr}(\Phi_j) - \frac{\|\Phi_j\|_F^2}{2\|\Phi_j\|})}{2\|\Phi_j\|}} - \frac{\|\Phi_j\|_F}{2\|\Phi_j\|})^2\right),$$

and

$$\mathbb{P}(\mathcal{F}^c) \leq 2d^2 \exp\left(-\frac{\lambda^2 n_g}{2048L^2\sigma_g^2(\max_{l,k}\|A_g^{lk}\|^2/n_g)}\right).$$

Proof of Lemma 2 Consider the event $\mathcal{F}$ first. With ϵ_g being σ_g-subgaussian,

$$\begin{aligned}\mathbb{P}(\mathcal{F}^c) &= \mathbb{P}(\frac{2}{n_g}|\mathcal{A}_g^*(\epsilon_g)|_\infty \geq \frac{\lambda}{16L}) \\ &\leq d^2 \max_{l,k\in[d]} \mathbb{P}(\frac{2}{n_g}|\sum_{i=1}^{n_g} A_{g,i}^{(l,k)}\epsilon_{g,i}| \geq \frac{\lambda}{16L}) \\ &\leq 2d^2 \exp\left(-\frac{\lambda^2 n_g}{2048L^2\sigma_g^2(\max_{l,k}\|A_g^{lk}\|^2/n_g)}\right),\end{aligned}$$

In the last inequality, we use the fact that ϵ_g is σ_g-subgaussian in the final inequality.

Next, we look at the event $\mathcal{G}_1$.

$$\mathbb{P}(\mathcal{G}_1^c) = \mathbb{P}(\frac{2}{n_g}\max_{j\in[d]}\|\mathcal{A}_g^*(\epsilon_g)^j U_g^*\| \geq \frac{\lambda}{8}) \leq d\max_{j\in[d]}\mathbb{P}(\frac{2}{n_g}\|\mathcal{A}_g^*(\epsilon_g)^j U_g^*\| \geq \frac{\lambda}{8}).$$

For a given j, observe that

$$\frac{4}{n_g^2}\|\mathcal{A}_g^*(\epsilon_g)^j U_g^*\|^2 = \frac{4}{n_g^2}\|\sum_{i=1}^{n_g} A_{g,i}^j U_g^*\epsilon_{g,i}\|^2 = \frac{4}{n_g}\epsilon_g^T\frac{A_g^j U_g^* U_g^{*T} A_g^{jT}}{n_g}\epsilon_g,$$

Note that Ψ_j has the same positive eigenvalues as $\frac{A_g^j U_g^* U_g^{*T} A_g^{jT}}{n_g}$. Different from Lounici et al. (2011), we assume subgaussian random noises instead of gaussian noises. Therefore, instead, we have from Lemma 6

$$\mathbb{P}(\frac{4}{n_g}\epsilon_g^T\frac{A_g^j U_g^* U_g^{*T} A_g^{jT}}{n_g}\epsilon_g \geq \frac{\lambda^2}{64}) \leq \exp\left(-(\sqrt{\frac{\frac{\lambda^2 n_g}{256\sigma_g^2} - (\mathsf{tr}(\Psi_j) - \frac{\|\Psi_j\|_F^2}{2\|\Psi_j\|})}{2\|\Psi_j\|}} - \frac{\|\Psi_j\|_F}{2\|\Psi_j\|})^2\right).$$

Combining the results above, we can derive that

$$\mathbb{P}(\mathcal{G}_1^c) \leq d \max_{j\in[d]} \exp\left(-(\sqrt{\frac{\frac{\lambda^2 n_g}{256\sigma_g^2} - (\mathsf{tr}(\Psi_j) - \frac{\|\Psi_j\|_F^2}{2\|\Psi_j\|})}{2\|\Psi_j\|}} - \frac{\|\Psi_j\|_F}{2\|\Psi_j\|})^2\right).$$

Similarly for event $\mathcal{G}_2$, we have

$$\mathbb{P}(\mathcal{G}_2^c) \leq d \max_{j\in[d]} \exp\left(-(\sqrt{\frac{\frac{\lambda^2 n_g^2}{256\sigma_g^2} - (\mathsf{tr}(\Phi_j) - \frac{\|\Phi_j\|_F^2}{2\|\Phi_j\|})}{2\|\Phi_j\|}} - \frac{\|\Phi_j\|_F}{2\|\Phi_j\|})^2\right). \quad \square$$

Proof of Theorem 1 Theorem 1 follows Lemma 1. Suppose $\frac{L\sigma_r(U_p^*)(3\alpha_p - 2\beta_p)}{8\sqrt{d}} \geq c$. On this event, the first term on the RHS of inequality (18) is smaller than the last term on the RHS. In order to make each term on the RHS to be smaller than $\frac{\delta}{5}$, we require

$$\begin{aligned}\lambda \geq \max\Bigg\{&\sqrt{\frac{2048L^2\sigma_g^2(\max_{l,k}\|A_g^{lk}\|^2/n_g)}{n_g}\log(\frac{10d^2}{\delta})}, \\ &\max_{j\in[d]}\sqrt{\frac{256\sigma_g^2}{n_g}(\mathsf{tr}(\Psi_j) + 2\|\Psi_j\|_F\sqrt{\log(\frac{5d}{\delta})} + 2\|\Psi_j\|\log(\frac{5d}{\delta}))}, \\ &\max_{j\in[d]}\sqrt{\frac{256\sigma_g^2}{n_g}(\mathsf{tr}(\Phi_j) + 2\|\Phi_j\|_F\sqrt{\log(\frac{5d}{\delta})} + 2\|\Phi_j\|\log(\frac{5d}{\delta}))}\Bigg\},\end{aligned}$$

and let c take the value

$$c = \sqrt{\frac{8\beta_p\sigma_p^2}{n_p}(2r(2d+1)\log(36\sqrt{2}) + \log(\frac{10}{\delta}))}.$$

Note that by definition of 1-smoothness(β_g)

$$\frac{1}{n_g}\|A_g^{lk}\|^2 = \langle E_{lk}, \frac{1}{n_g}\mathcal{A}_g^*(\mathcal{A}_g(E_{lk}))\rangle \leq \beta_g,$$

where $E_{lk} \in \mathbb{R}^{d\times d}$ is a basis matrix whose (l,k) entry is 1 and otherwise 0. On the other hand,

$$\|\Psi_j\| = \max_{\|x\|=1, x\in\mathbb{R}^r} x^T U_g^{*T}\frac{A_g^{jT}A_g^j}{n_g}U_g^* x = \max_{\|x\|=1} x^T U_g^{*T}\frac{A_g^{jT}A_g^j}{n_g}U_g^* x.$$

If we define a matrix $E_j(x)$ whose j^{th} row is x^T and otherwise 0, then

$$\|\Psi_j\| = \max_{\|x\|=1}\frac{1}{n_g}\langle E_j(U_g^* x), A_g^*(A_g(E_j(U_g^* x)))\rangle.$$

As $\|x\| = 1$, we have

$$\|E_j(U_g^* x)\|_F = \|U_g^* x\| \leq \sigma_1(U_g^*).$$

Therefore, we have

$$\|\Psi_j\| \leq \max_{\|R\|_F \leq \sigma_1(U_g^*)}\frac{1}{n_g}\langle R, A_g^*(A_g(R)\rangle \leq \beta_g\sigma_1^2(U_g^*).$$

With a similar analysis, we have

$$\|\Phi_j\| \leq \beta_g\sigma_1^2(U_g^*).$$

Given the above results, we can bound the trace and Frobenius norm of Ψ_j and Φ_j proportional to their rank:

$$\mathsf{tr}(\Psi_j) \leq r\|\Psi_j\| \leq r\beta_g\sigma_1^2(U_g^*), \quad \|\Psi_j\|_F \leq \sqrt{r}\|\Psi_j\| \leq \sqrt{r}\beta_g\sigma_1^2(U_g^*)$$
$$\mathsf{tr}(\Phi_j) \leq r\beta_g\sigma_1^2(U_g^*), \quad \|\Phi_j\|_F \leq \sqrt{r}\beta_g\sigma_1^2(U_g^*).$$

Combining all the above results, we can instead set λ as:

$$\lambda = \max\left\{\sqrt{\frac{2048L^2\beta_g\sigma_g^2}{n_g}\log(\frac{10d^2}{\delta})}, \right.$$
$$\left.\sqrt{\frac{256\beta_g\sigma_g^2\sigma_1^2(U_g^*)}{n_g}(r + 2\sqrt{r\log(\frac{5d}{\delta})} + 2\log(\frac{5d}{\delta}))}\right\}.$$

Therefore, with the above choice of λ and with n_p and d such that

$$\sqrt{\frac{8\beta_p\sigma_p^2}{n_p}(2r(2d+1)\log(36\sqrt{2}) + \log(\frac{10}{\delta}))} \leq \frac{L\sigma_r(U_p^*)(3\alpha_p - 2\beta_p)}{8\sqrt{d}},$$

we obtain the following bound for estimation error of $\widetilde{\Delta}_U$:

$$\|\widehat{\Delta}_U - \widetilde{\Delta}_U\|_{2,1} = \mathcal{O}\left(\sqrt{\frac{\sigma_g^2 s^2(r + \log(\frac{d^2}{\delta}))}{n_g}} + \sqrt{\frac{\sigma_p^2(rd^2 + d\log(\frac{1}{\delta}))}{n_p}}\right),$$

with probability greater than $1-\delta$. Consequently,

$$\ell(\widehat{U}_g^{TL}, U_g^*) = \mathcal{O}\left(\sqrt{\frac{\sigma_g^2 s^2(r + \log(\frac{d^2}{\delta}))}{n_g}} + \sqrt{\frac{\sigma_p^2(rd^2 + d\log(\frac{1}{\delta}))}{n_p}}\right),$$

with probability at least $1-\delta$. $\square$

Appendix C: Local Minima

Proof of Proposition 4 By definition,

$$\mathbb{E}_{X_g|\mathcal{A}_g}[\langle\nabla f(\widetilde{\Delta}_U+\Delta)-\nabla f(\widetilde{\Delta}_U),\Delta\rangle]=\frac{2}{n_g}\mathcal{A}_g(\Delta U_g^{*T}+U_g^*\Delta^T+\Delta\Delta^T)^T\mathcal{A}_g(\Delta U_g^{*T}+U_g^*\Delta^T+2\Delta\Delta^T).$$

As $\mathcal{A}_g$ satisfies $\mathsf{RSC}(\sqrt{\frac{2}{3}}U_g^*,\eta,\tau)$,

$$\begin{aligned}&\frac{2}{n_g}\mathcal{A}_g(\Delta U_g^{*T}+U_g^*\Delta^T+\Delta\Delta^T)^T\mathcal{A}_g(\Delta U_g^{*T}+U_g^*\Delta^T+2\Delta\Delta^T)\\=&\frac{2}{n_g}(\|\mathcal{A}_g(\Delta U_g^{*T}+U_g^*\Delta^T+\frac{3}{2}\Delta\Delta^T)\|^2-\frac{1}{4}\|\mathcal{A}_g(\Delta\Delta^T)\|^2)\\\geq&2\eta\|\Delta U_g^{*T}+U_g^*\Delta^T+\frac{3}{2}\Delta\Delta^T\|_F^2-\frac{\beta_g}{2}\|\Delta\Delta^T\|_F^2-\frac{3}{2}\tau(n,d,r)\|\Delta\|_{2,1}^2.\end{aligned}$$

The first two terms of the above have

$$\begin{aligned}&2\eta\|\Delta U_g^{*T}+U_g^*\Delta^T+\frac{3}{2}\Delta\Delta^T\|_F^2-\frac{\beta_g}{2}\|\Delta\Delta^T\|_F^2\\\geq&8\eta\|\Delta U_g^{*T}\|_F^2-12\eta\langle\Delta U_g^{*T},\Delta\Delta^T\rangle+\frac{9\eta}{2}\|\Delta\Delta^T\|_F^2-\frac{\beta_g}{2}\|\Delta\Delta^T\|_F^2\\\geq&8\eta\|\Delta U_g^{*T}\|_F^2-12\eta\sup_\Delta\frac{\|\Delta\Delta^T\|_F}{\|\Delta U_g^{*T}\|_F}\|\Delta U_g^{*T}\|_F^2+\frac{9\eta}{2}\|\Delta\Delta^T\|_F^2-\frac{\beta_g}{2}\|\Delta\Delta^T\|_F^2.\end{aligned}$$

When $\rho\leq\sigma_r(U_g^*)/3$, it holds that

$$\sup_\Delta\frac{\|\Delta\Delta^T\|_F}{\|\Delta U_g^{*T}\|_F}\leq\frac{\|\Delta\|_F}{\sigma_r(U_g^*)}\leq\frac{1}{3}.$$

Therefore, we can lower bound the first two terms with

$$2\eta\|\Delta U_g^{*T}+U_g^*\Delta^T+\frac{3}{2}\Delta\Delta^T\|_F^2-\frac{\beta_g}{2}\|\Delta\Delta^T\|_F^2\geq4\eta\|\Delta U_g^{*T}\|_F^2,$$

where we use the assumption that $9\eta\geq\beta_g$. Combining all of the above results gives rise to our first statement.

On the other hand, when $\|\Delta\|_F\leq\rho$,

$$\begin{aligned}\mathbb{E}_{X_g|\mathcal{A}_g}[\langle\nabla f(\widetilde{\Delta}_U+\Delta)-\nabla f(\widetilde{\Delta}_U),\Delta\rangle]=&\frac{2}{n_g}(\|\mathcal{A}_g(\Delta U_g^{*T}+U_g^*\Delta^T+\frac{3}{2}\Delta\Delta^T)\|^2-\frac{1}{4}\|\mathcal{A}_g(\Delta\Delta^T)\|^2)\\\geq&\eta_1\|\Delta\|_F^2-\tau_1(n_g,d,r)\|\Delta\|_{2,1}^2.\end{aligned}$$

Therefore, we have

$$\frac{2}{n_g}\|\mathcal{A}_g(\Delta U_g^{*T}+U_g^*\Delta^T+\frac{3}{2}\Delta\Delta^T)\|^2\geq\eta_1\|\Delta\|_F^2-\tau_1(n_g,d,r)\|\Delta\|_{2,1}^2.$$

Note that

$$\begin{aligned}\|\Delta U_g^{*T}+U_g^*\Delta^T+\frac{3}{2}\Delta\Delta^T\|_F^2=&\|\Delta U_g^{*T}+U_g^*\Delta^T\|_F^2+\|\frac{3}{2}\Delta\Delta^T\|_F^2+6\langle U_g^*\Delta^T,\Delta\Delta^T\rangle\\=&4\|\Delta U_g^{*T}\|_F^2+\|\frac{3}{2}\Delta\Delta^T\|_F^2+6\langle U_g^*\Delta^T,\Delta\Delta^T\rangle\\\leq&4\|\Delta U_g^{*T}\|_F^2+\|\frac{3}{2}\Delta\Delta^T\|_F^2+6\|U_g^*\Delta^T\|_F\|\Delta\Delta^T\|_F\\\leq&4\sigma_1(U_g^*)^2\|\Delta\|_F^2+\frac{9\rho^2}{4}\|\Delta\|_F^2+6\sigma_1(U_g^*)\rho\|\Delta\|_F^2\\=&(2\sigma_1(U_g^*)+\frac{3\rho}{2})^2\|\Delta\|_F^2.\end{aligned}$$

Therefore, $\mathcal{A}_g$ satisfies $\mathsf{RSC}(\sqrt{\frac{2}{3}}U_g^*,\eta,\tau)$ with $\eta=\frac{\eta_1}{2(2\sigma_1(U_g^*)+3\rho/2)^2}$ and $\tau=\tau_1/3$. $\square$

Proof of Theorem 2 Let $\Delta = \widehat{\Delta}_U - \widetilde{\Delta}_U$. We first show that the local minima fall within $\|\Delta\|_F \le \rho$ with high probability. If $\|\Delta\|_F \ge \rho$, condition (6b) gives

$$\begin{aligned}\mathbb{E}_{X_g|\mathcal{A}_g}[\langle \nabla f(\widehat{\Delta}_U) - \nabla f(\widetilde{\Delta}_U), \Delta\rangle] &= \langle \nabla f(\widehat{\Delta}_U) - \nabla f(\widetilde{\Delta}_U), \Delta\rangle + \frac{4}{n_g}\epsilon_g^T \mathcal{A}_g(\Delta\Delta^T) \\ &\ge \eta_2\|\Delta\|_F - \tau_2(n_g, d, r)\|\Delta\|_{2,1}. \end{aligned} \tag{19}$$

As $\widehat{\Delta}_U$ is a local minimum, a necessary condition of the optimization problem is

$$\langle \nabla f(\widehat{\Delta}_U) + \lambda \partial \|\widehat{\Delta}_U\|_{2,1}, \Delta_U - \widehat{\Delta}_U\rangle \ge 0, \tag{20}$$

for any Δ_U within the search area. $\partial\|\widehat{\Delta}_U\|_{2,1}$ is the subgradient of the $\ell_{2,1}$ norm at $\widehat{\Delta}_U$, i.e.,

$$\partial\|X\|_{2,1} \begin{cases} = \nabla\|X\|_{2,1}, & \|X^j\| > 0, \forall j \in [d] \\ \in \{Z \mid \|Z\|_{2,\infty} \le 1\}, & \text{otherwise.} \end{cases}$$

The combination of (19) and (20) implies

$$\langle -\nabla f(\widetilde{\Delta}_U) - \lambda\partial\|\widehat{\Delta}_U\|_{2,1}, \Delta\rangle + \frac{4}{n_g}\epsilon_g^T\mathcal{A}_g(\Delta\Delta^T) \ge \eta_2\|\Delta\|_F - \tau_2(n_g,d,r)\|\Delta\|_{2,1}. \tag{21}$$

Using Hölder's inequality, we have

$$\langle \lambda\partial\|\widehat{\Delta}_U\|_{2,1}, \Delta\rangle \le \lambda\|\partial\|\widehat{\Delta}_U\|_{2,1}\|_{2,\infty}\|\Delta\|_{2,1} \le \lambda\|\Delta\|_{2,1},$$

where we use $\|\partial\|\widehat{\Delta}_U\|_{2,1}\|_{2,\infty} \le 1$. Using the above result, replacing $\nabla f(\widetilde{\Delta}_U)$, and rearranging inequality (21) give

$$\eta_2\|\Delta\|_F - \tau_2(\sqrt{\frac{r}{n_g}} + \sqrt{\frac{\log d}{n_g}})\|\Delta\|_{2,1} \le \frac{2}{n_g}\langle \epsilon_g, \mathcal{A}_g(\Delta U_g^{*T} + U_g^*\Delta^T + 2\Delta\Delta^T)\rangle + \lambda\|\Delta\|_{2,1}.$$

Consider the same event of $\mathcal{I}$ in (13) and the following events

$$\bar{\mathcal{G}}_1 = \left\{\frac{2}{n_g}\max_{j\in[d]}\|\mathcal{A}_g^*(\epsilon_g)^j U_g^*\| \le \frac{\lambda}{16}\right\}, \quad \bar{\mathcal{G}}_2 = \left\{\frac{2}{n_g}\max_{j\in[d]}\|\mathcal{A}_g^*(\epsilon_g)^{Tj} U_g^*\| \le \frac{\lambda}{16}\right\},$$

and

$$\bar{\mathcal{F}} = \left\{\frac{4}{n_g}|\mathcal{A}_g^*(\epsilon_g)|_\infty \le \frac{\lambda}{32L}\right\}.$$

We know from Lemma 2 these events hold with high probability. On the event $\mathcal{I}$, we further have $\|\Delta\|_{2,1} \le 4L$. Therefore, under all these events and assuming that $\lambda \ge \frac{4}{3}\tau_2(n_g,d,r)$, we have

$$\eta_2\|\Delta\|_F \le 2\lambda\|\Delta\|_{2,1} \le 8\lambda L.$$

With $\lambda \le (\rho\eta_2)/(8L)$, we have $\|\Delta\|_F \le \rho$, which is a contradiction.

Consequently, we only need to consider $\|\Delta\|_F \le \rho$. Condition (6a) gives

$$\langle \nabla f(\widehat{\Delta}_U) - \nabla f(\widetilde{\Delta}_U), \widehat{\Delta}_U - \widetilde{\Delta}_U\rangle + \frac{4}{n_g}\epsilon_g^T\mathcal{A}_g(\Delta\Delta^T) \ge \eta_1\|\Delta\|_F^2 - \tau_1(n_g,d,r)\|\Delta\|_{2,1}^2. \tag{22}$$

Since the $\ell_{2,1}$ norm is convex, we have for any Δ_U

$$\langle \partial\|\widehat{\Delta}_U\|_{2,1}, \Delta_U - \widehat{\Delta}_U\rangle \le \|\Delta_U\|_{2,1} - \|\widehat{\Delta}_U\|_{2,1}. \tag{23}$$

Combining inequalities (22), (20), and (23), we have

$$\eta_1\|\Delta\|_F^2 \;\le\; \frac{2}{n_g}\langle \epsilon_g, \mathcal{A}_g(\Delta U_g^{*T} \;+\; U_g^*\Delta^T \;+\; 2\Delta\Delta^T)\rangle \;+\; \lambda(\|\widetilde{\Delta}_U\|_{2,1} \;-\; \|\widehat{\Delta}_U\|_{2,1}) \;+\; 4L\tau_1(n_g,d,r)\|\Delta\|_{2,1},$$

where we use $\|\Delta\|_{2,1} \le 4L$. Since $\lambda \ge 16L\tau_1(n_g,d,r)$, we can derive that

$$\eta_1\|\Delta\|_F^2 \le \frac{\lambda}{2}\|\widetilde{\Delta}_U - \widehat{\Delta}_U\|_{2,1} + \lambda(\|\widetilde{\Delta}_U\|_{2,1} - \|\widehat{\Delta}_U\|_{2,1}),$$

on the events $\bar{\mathcal{G}}_1$, $\bar{\mathcal{G}}_2$ and $\bar{\mathcal{F}}$. Further arrange the inequality and we have

$$\eta_1\|\widehat{\Delta}_U - \widetilde{\Delta}_U\|_F^2 + \frac{\lambda}{2}\sum_{j\in J^c}\|(\widehat{\Delta}_U - \widetilde{\Delta}_U)^j\| \le \frac{3\lambda}{2}\sum_{j\in J}\|(\widehat{\Delta}_U - \widetilde{\Delta}_U)^j\| + 2\lambda\sum_{j\in J^c}\|\nu^j\|. \tag{24}$$

Inequality (24) gives us

$$\|\widehat{\Delta}_U - \widetilde{\Delta}_U\|_F \le \max\left\{\frac{3\lambda\sqrt{s}}{\eta_1}, \sqrt{\frac{4\lambda\sum_{j\in J^c}\|\nu^j\|}{\eta_1}}\right\},$$

and

$$\|\widehat{\Delta}_U - \widetilde{\Delta}_U\|_{2,1} \le 4\sqrt{s}\|\widehat{\Delta}_U - \widetilde{\Delta}_U\|_F + 4\sum_{j\in J^c}\|\nu^j\|,$$

which implies

$$\|\widehat{\Delta}_U - \widetilde{\Delta}_U\|_{2,1} \le \max\left\{\frac{12\lambda s}{\eta_1} + 4\sum_{j\in J^c}\|\nu^j\|, 8\sqrt{\frac{\lambda s\sum_{j\in J^c}\|\nu^j\|}{\eta_1}} + 4\sum_{j\in J^c}\|\nu^j\|\right\} \le \frac{12\lambda s}{\eta_1} + 6\sum_{j\in J^c}\|\nu^j\|.$$

Under the same event $\mathcal{J}$ in equation (17), we have that

$$\|\widehat{\Delta}_U - \widetilde{\Delta}_U\|_{2,1} \le 12(\frac{\lambda s}{\eta_1} + \frac{4\sqrt{d}c}{(3\alpha_p - 2\beta_p)\sigma_r(U_p^*)}).$$

The final concentration inequality is as follows:

$$\begin{aligned}
&\mathbb{P}\left(\|\widehat{\Delta}_U - \widetilde{\Delta}_U\|_{2,1} \ge 12(\frac{\lambda s}{\eta_1} + \frac{4\sqrt{d}c}{(3\alpha_p - 2\beta_p)\sigma_r(U_p^*)})\right)\\
&\qquad\le \mathbb{P}(\mathcal{I}^c) + \mathbb{P}(\bar{\mathcal{G}}_1^c) + \mathbb{P}(\bar{\mathcal{G}}_2^c) + \mathbb{P}(\bar{\mathcal{F}}^c) + \mathbb{P}(\mathcal{J}^c)\\
&\qquad\le 2(36\sqrt{2})^{2r(2d+1)}\exp(-\frac{L^2\sigma_r^2(U_p^*)(3\alpha_p - 2\beta_p)^2 n_p}{512\beta_p\sigma_p^2 d})\\
&\qquad\quad + 2d^2\exp\left(-\frac{\lambda^2 n_g}{32768L^2\sigma_g^2(\max_{l,k}\|A_g^{lk}\|^2/n_g)}\right)\\
&\quad + d\max_{j\in[d]}\exp\left(-(\sqrt{\frac{\frac{\lambda^2 n_g}{512\sigma_g^2} - (\mathsf{tr}(\Psi_j) - \frac{\|\Psi_j\|_F^2}{2\|\Psi_j\|})}{2\|\Psi_j\|}} - \frac{\|\Psi_j\|_F}{2\|\Psi_j\|})^2\right)\\
&\quad + d\max_{j\in[d]}\exp\left(-(\sqrt{\frac{\frac{\lambda^2 n_g}{512\sigma_g^2} - (\mathsf{tr}(\Phi_j) - \frac{\|\Phi_j\|_F^2}{2\|\Phi_j\|})}{2\|\Phi_j\|}} - \frac{\|\Phi_j\|_F}{2\|\Phi_j\|})^2\right)\\
&\qquad\qquad + 2(36\sqrt{2})^{2r(2d+1)}\exp(-\frac{c^2 n_p}{8\beta_p\sigma_p^2}).
\end{aligned} \tag{25}$$

Following a similar analysis in the proof of Theorem 1, set λ as

$$\lambda = \max\left\{\sqrt{\frac{32768L^2\beta_g\sigma_g^2}{n_g}\log(\frac{10d^2}{\delta})}, \sqrt{\frac{512\beta_g\sigma_g^2\sigma_1^2(U_g^*)}{n_g}(r+2\sqrt{r\log(\frac{5d}{\delta})}+2\log(\frac{5d}{\delta}))}, \right.$$
$$\left. \frac{4}{3}\tau_2(n_g,d,r), 16L\tau_1(n_g,d,r)\right\},$$

and take

$$c = \sqrt{\frac{8\beta_p\sigma_p^2}{n_p}(2r(2d+1)\log(36\sqrt{2})+\log(\frac{10}{\delta}))}.$$

Then, given

$$\sqrt{\frac{8\beta_p\sigma_p^2}{n_p}(2r(2d+1)\log(36\sqrt{2})+\log(\frac{10}{\delta}))} \le \frac{L\sigma_r(U_p^*)(3\alpha_p-2\beta_p)}{8\sqrt{d}},$$

we obtain statistical guarantees on all local minima

$$\ell(\widehat{U}_g^{TL}, U_g^*) = \mathcal{O}\left(\sqrt{\frac{\sigma_g^2 s^2(r+\log(\frac{d^2}{\delta}))}{n_g}} + \sqrt{\frac{\sigma_p^2(rd^2+d\log(\frac{1}{\delta}))}{n_p}}\right),$$

with probability at least $1-\delta$. This provides us with a same bound as the global minimum. □

Appendix D: Error Bound of Gold Estimator

Before discussing the estimation error bound of the gold estimator, we first introduce a lemma that helps with our proof.

LEMMA 3. *Let $\mathcal{Z} \subset \mathbb{R}^{d\times d}$ be the subspace of matrices with rank at most r. The operator $\mathcal{A}$ is r-smooth(β) in $\mathcal{Z}$ and ϵ is σ-subgaussian. Then, we have*

$$\mathbb{P}(\sup_{Z\in\mathcal{Z}} |\frac{1}{n}\sum_{i\in[n]}\epsilon_i\langle A_i, Z\rangle| \le c\|Z\|_F) \ge 1 - 2(36\sqrt{2})^{r(2d+1)}\exp\left(-\frac{c^2 n}{8\beta\sigma^2}\right).$$

Proof of Lemma 3 Without loss of generality, consider $\mathcal{Z} = \{Z \in \mathbb{R}^{d\times d} | \,\mathsf{rank}(Z) \le r, \|Z\|_F = 1\}$. Define $\mathcal{N}$ to be a $\frac{1}{4\sqrt{2}}$-net of $\mathcal{Z}$. Lemma 4 gives the covering number for the set $\mathcal{Z}$:

$$|\mathcal{N}| \le (36\sqrt{2})^{r(2d+1)}.$$

For any $Z \in \mathcal{Z}$, there exists $Z' \in \mathcal{N}$ with $\|Z - Z'\|_F \le \frac{1}{4\sqrt{2}}$, such that

$$|\sum_{i\in[n]}\epsilon_i\langle A_i, Z\rangle| \le |\sum_{i\in[n]}\epsilon_i\langle A_i, Z'\rangle| + |\sum_{i\in[n]}\epsilon_i\langle A_i, Z-Z'\rangle|. \tag{26}$$

Set $\Delta_Z = Z - Z'$ and note that $\mathsf{rank}(\Delta_Z) \le 2r$. We decompose Δ_Z into two matrices, $\Delta_Z = \Delta_{Z,1} + \Delta_{Z,2}$, that satisfy $\mathsf{rank}(\Delta_{Z,j}) \le r$ for $j = 1, 2$ and $\langle \Delta_{Z,1}, \Delta_{Z,2}\rangle = 0$ (e.g. through SVD). As $\|\Delta_{Z,1}\|_F + \|\Delta_{Z,2}\|_F \le \sqrt{2}\|\Delta_Z\|_F$, we have $\|\Delta_{Z,j}\|_F \le \frac{1}{4}$, $j = 1, 2$. Combined with inequality (26), we have

$$|\sum_{i\in[n]}\epsilon_i\langle A_i, Z\rangle| \le \sup_{Z'\in\mathcal{N}} |\sum_{i\in[n]}\epsilon_i\langle A_i, Z'\rangle| + \frac{1}{2}\sup_{Z\in\mathcal{Z}} |\sum_{i\in[n]}\epsilon_i\langle A_i, Z\rangle|.$$

Since the above holds for any $Z \in \mathcal{Z}$, the following holds:

$$\sup_{Z\in\mathcal{Z}} |\sum_{i\in[n]}\epsilon_i\langle A_i, Z\rangle| \le 2\sup_{Z'\in\mathcal{N}} |\sum_{i\in[n]}\epsilon_i\langle A_i, Z'\rangle|.$$

Then it follows from the union bound that

$$\begin{aligned}\mathbb{P}(\sup_{Z\in\mathcal{Z}}|\frac{1}{n}\sum_{i\in[n]}\epsilon_i\langle A_i,Z\rangle|\geq c) &\leq \mathbb{P}(\sup_{Z\in\mathcal{N}}|\frac{1}{n}\sum_{i\in[n]}\epsilon_i\langle A_i,Z\rangle|\geq \frac{c}{2})\\ &\leq |\mathcal{N}|\max_{Z\in\mathcal{N}}\mathbb{P}(|\frac{1}{n}\sum_{i\in[n]}\epsilon_i\langle A_i,Z\rangle|\geq\frac{c}{2})\\ &\leq 2|\mathcal{N}|\exp\left(-\frac{c^2n}{8\beta\sigma^2}\right)\\ &= 2(36\sqrt{2})^{r(2d+1)}\exp\left(-\frac{c^2n}{8\beta\sigma^2}\right).\end{aligned}$$

The last inequality uses r-smoothness(β) of $\mathcal{A}$ and a tail inequality of σ-subgaussian random variables. □

Proof of Theorem 3 The proof mainly follows Theorem 8 and Theorem 31 of Ge et al. (2017) but we also provide here for completeness. As in Ge et al. (2017), we use the notation $U:\mathcal{H}:V$ to denote the inner product $\langle U,\mathcal{H}(V)\rangle$ for $U,V\in\mathbb{R}^{d_1\times d_2}$. The linear operator $\mathcal{H}$ can be viewed as a $d_1d_2\times d_1d_2$ matrix. In our problem (9), we define

$$\Theta:\mathcal{H}:\Theta=\frac{1}{n_g}\|\mathcal{A}_g(\Theta)\|^2$$

for any $\Theta\in\mathbb{R}^{d\times d}$. We can rewrite problem (9) as

$$\min_{U_g} f(U_g)=\frac{1}{n_g}\|X_g-\mathcal{A}_g(U_gU_g^T)\|^2.$$

Rearrange the objective function with $\mathcal{H}$ and we have

$$f(U_g)=(U_gU_g^T-\Theta_g^*):\mathcal{H}:(U_gU_g^T-\Theta_g^*)+Q(U_g),$$

with

$$Q(U_g)=-\frac{2}{n_g}\sum_{i\in[n_g]}\langle A_{g,i},U_gU_g^T-\Theta_g^*\rangle\epsilon_{g,i}+\frac{1}{n_g}\sum_{i\in[n_g]}\epsilon_{g,i}^2.$$

Define $\Delta=\widehat{U}_g-U_g^*R_{(\widehat{U}_g,U_g^*)}$. By Lemma 7 from Ge et al. (2017), we have for the Hessian $\nabla^2f(\widehat{U}_g)$ with $\nabla f(\widehat{U}_g)=0$

$$\Delta:\nabla^2f(\widehat{U}_g):\Delta=2\Delta\Delta^T:\mathcal{H}:\Delta\Delta^T-6(\widehat{\Theta}_g-\Theta_g^*):\mathcal{H}:(\widehat{\Theta}_g-\Theta_g^*)+\Delta:\nabla^2Q(\widehat{U}_g):\Delta-4\langle\nabla Q(\widehat{U}_g),\Delta\rangle.$$

Using Lemma 5 and the $2r$-RWC assumption, the above inequality can be simplified as

$$\Delta:\nabla^2f(\widehat{U}_g):\Delta\leq-2(3\alpha_g-2\beta_g)\|\widehat{\Theta}_g-\Theta_g^*\|_F^2+\Delta:\nabla^2Q(\widehat{U}_g):\Delta-4\langle\nabla Q(\widehat{U}_g),\Delta\rangle.$$

We then bound the terms related to function Q. Note that

$$\Delta:\nabla^2Q(\widehat{U}_g):\Delta-4\langle\nabla Q(\widehat{U}_g),\Delta\rangle=\frac{4}{n_g}\sum_{i\in[n_g]}\langle A_{g,i},\widehat{\Theta}_g-\Theta_g^*\rangle\epsilon_{g,i}+\frac{4}{n_g}\sum_{i\in[n_g]}\langle A_{g,i},\widehat{U}_g\Delta^T-\Delta\widehat{U}_g^T\rangle\epsilon_{g,i}$$

Define $\mathcal{Z}=\{Z\in\mathbb{R}^{d\times d}\,|\,\mathsf{rank}(Z)\leq 2r\}$. On the event

$$\mathcal{E}_g=\left\{\sup_{Z\in\mathcal{Z}}|\frac{1}{n_g}\sum_{i\in[n_g]}\epsilon_{g,i}\langle A_{g,i},Z\rangle|\leq c\|Z\|_F\right\},$$

it holds that

$$\frac{4}{n_g}\sum_{i\in[n_g]}\langle A_{g,i},\widehat{\Theta}_g-\Theta_g^*\rangle\epsilon_{g,i}\le 4c\|\widehat{\Theta}_g-\Theta_g^*\|_F$$
$$\frac{4}{n_g}\sum_{i\in[n_g]}\langle A_{g,i},\widehat{U}_g\Delta^T-\Delta\widehat{U}_g^T\rangle\epsilon_{g,i}\le 4(1+\sqrt{2})c\|\widehat{\Theta}_g-\Theta_g^*\|_F,$$

where the second inequality uses Lemma 5. Therefore, we have

$$\Delta:\nabla^2 f(\widehat{U}_g):\Delta\le -2(3\alpha_g-2\beta_g)\|\widehat{\Theta}_g-\Theta_g^*\|_F^2+(8+4\sqrt{2})c\|\widehat{\Theta}_g-\Theta_g^*\|_F.$$

Since $\widehat{U}_g$ is a local minimum, we must have

$$-2(3\alpha_g-2\beta_g)\|\widehat{\Theta}_g-\Theta_g^*\|_F^2+(8+4\sqrt{2})c\|\widehat{\Theta}_g-\Theta_g^*\|_F\ge 0,$$

that is, $\widehat{\Theta}_g$ satisfies

$$\|\widehat{\Theta}_g-\Theta_g^*\|_F\le\frac{(4+2\sqrt{2})c}{3\alpha_g-2\beta_g}.$$

Again using Lemma 5 gives

$$\|\widehat{U}_g-U_g^*R_{(\widehat{U}_g,U_g^*)}\|_F\le\frac{1}{\sqrt{2(\sqrt{2}-1)\sigma_r(\Theta_g^*)}}\|\widehat{\Theta}_g-\Theta_g^*\|_F\le\frac{8c}{(3\alpha_g-2\beta_g)\sigma_r(U_g^*)}.$$

Further by Cauchy-Schwarz, we have

$$\|\widehat{U}_g-U_g^*R_{(\widehat{U}_g,U_g^*)}\|_{2,1}\le\frac{8c\sqrt{d}}{(3\alpha_g-2\beta_g)\sigma_r(U_g^*)}.$$

Lemma 3 shows with high probability $\mathcal{E}_g$ holds:

$$\mathbb{P}(\mathcal{E}_g)\ge 1-2(36\sqrt{2})^{2r(2d+1)}\exp\left(-\frac{c^2n_g}{8\beta_g\sigma_g^2}\right).$$

The result follows by taking

$$c=\sqrt{\frac{8\beta_g\sigma_g^2(2r(2d+1)\log(36\sqrt{2})+\log(\frac{2}{\delta}))}{n_g}}.\quad\square$$

Appendix E: Error Bound of Proxy Estimator

Proof of Theorem 4 Same as the proof of Theorem 3, we get

$$\|\widehat{U}_p-U_p^*R_{(\widehat{U}_p,U_p^*)}\|_{2,1}\le\frac{8c\sqrt{d}}{(3\alpha_p-2\beta_p)\sigma_r(U_p^*)}.$$

On the event

$$\mathcal{E}_p=\left\{\sup_{Z\in\mathcal{Z}}|\frac{1}{n_p}\sum_{i\in[n_p]}\epsilon_{p,i}\langle A_{p,i},Z\rangle|\le c\|Z\|_F\right\}.$$

To measure the estimation error of $\widehat{U}_p$ for U_g^*, we need to align $\widehat{U}_p$ with U_g^*. The estimation error of using proxy estimator for gold data is

$$\begin{aligned}\|\widehat{U}_p-U_g^*R_{(\widehat{U}_p,U_g^*)}\|_{2,1}=&\|\widehat{U}_p-U_p^*R_{(\widehat{U}_p,U_p^*)}+U_p^*R_{(\widehat{U}_p,U_p^*)}-(U_p^*+\Delta_U^*)R_{(\widehat{U}_p,U_g^*)}\|_{2,1}\\ \le&\|\widehat{U}_p-U_p^*R_{(\widehat{U}_p,U_p^*)}\|_{2,1}+\|U_p^*(R_{(\widehat{U}_p,U_p^*)}-R_{(\widehat{U}_p,U_g^*)})\|_{2,1}+\|\Delta_U^*\|_{2,1}.\end{aligned}$$

Therefore, we have

$$\|\widehat{U}_p - U_g^* R_{(\widehat{U}_p, U_g^*)}\|_{2,1} \le \|\Delta_U^*\| + \|U_p^*(R_{(\widehat{U}_p, U_p^*)} - R_{(\widehat{U}_p, U_g^*)})\|_{2,1} + \frac{8c\sqrt{d}}{(3\alpha_p - 2\beta_p)\sigma_r(U_p^*)}.$$

By Lemma 3,

$$\mathbb{P}(\mathcal{E}_p) \ge 1 - 2(36\sqrt{2})^{2r(2d+1)} \exp\left(-\frac{c^2 n_p}{8\beta_p \sigma_p^2}\right).$$

Similarly, the result follows by taking

$$\omega = \|U_p^*(R_{(\widehat{U}_p, U_p^*)} - R_{(\widehat{U}_p, U_g^*)})\|_{2,1},$$

and

$$c = \sqrt{\frac{8\beta_p \sigma_p^2 (2r(2d+1)\log(36\sqrt{2}) + \log(\frac{2}{\delta}))}{n_p}}. \quad \square$$

Appendix F: Useful Lemmas

LEMMA 4. *Let $\mathcal{Z} = \{Z \in \mathbb{R}^{d_1 \times d_2} | \, \mathsf{rank}(Z) \le r, \|Z\|_F = 1\}$. Then there exists an ϵ-net $\mathcal{N} \subseteq \mathcal{Z}$ with respect to the Frobenius norm obeying*

$$|\mathcal{N}| \le (9/\epsilon)^{(d_1+d_2+1)r}.$$

Proof of Lemma 4 See Lemma 3.1 of Candes and Plan (2011). $\square$

LEMMA 5. *Let $\Delta = \widehat{U} - U^* R_{(\widehat{U}, U^*)}$, $\Theta^* = U^* U^{*T}$ and $\widehat{\Theta} = \widehat{U}\widehat{U}^T$, where $R_{(\widehat{U}, U^*)}$ is defined in Definition 2. Then,*

$$\|\Delta\Delta^T\|_F^2 \le 2\|\widehat{\Theta} - \Theta^*\|_F^2$$
$$\sigma_r(\Theta^*)\|\Delta\|_F^2 \le \frac{1}{2(\sqrt{2}-1)}\|\widehat{\Theta} - \Theta^*\|_F^2.$$

Proof of Lemma 5 See Lemma 6 of Ge et al. (2017). $\square$

LEMMA 6. *Let $X \in \mathbb{R}^n$ be a σ-subgaussian random vector, $A \in \mathbb{R}^{m \times n}$ and $\Sigma = A^T A$. Then, for any $t > 0$,*

$$\mathbb{P}(\|AX\|^2 > \sigma^2(\mathsf{tr}(\Sigma) + 2\|\Sigma\|_F\sqrt{t} + 2\|\Sigma\|t)) \le \exp(-t).$$

Proof of Lemma 6 See Theorem 1 of Hsu et al. (2012). $\square$

LEMMA 7. *Let gaussian random vector $X = \begin{bmatrix} X_1 & \cdots & X_n \end{bmatrix}^T \in \mathbb{R}^n$ with i.i.d. $X_i \sim N(0,1)$, and $f : \mathbb{R}^n \to \mathbb{R}$ an L-Lipschitz function, i.e., $|f(x) - f(y)| \le L\|x - y\|$ for any $x, y \in \mathbb{R}^n$. Then, for any $t > 0$*

$$\mathbb{P}(f(X) - \mathbb{E}[f(X)] > t) \le \exp(-\frac{t^2}{2L^2}).$$

Proof of Lemma 7 See Theorem 5.6 in Boucheron et al. (2013). $\square$

LEMMA 8. *For gaussian random vector $X = \begin{bmatrix} X_1 & \cdots & X_n \end{bmatrix}^T \in \mathbb{R}^n$ with $X \sim N(0, \Sigma)$, the demeaned $\|X\|$ is subgaussian, i.e., for any $t > 0$,*

$$\mathbb{P}(\|X\| - \mathbb{E}[\|X\|] > t) \le \exp(-\frac{t^2}{2\|\Sigma\|}).$$

Proof of Lemma 8 It is a direct application of Lemma 7. $\square$

LEMMA 9. *For random variables $X_i, i \in [n]$ with X_i drawn from σ_i-subgaussian,*

$$\mathbb{E}[\max_{i\in[n]} X_i] \le \max_{i\in[n]} \sigma_i \sqrt{2\log n}, \quad \mathbb{E}[\max_{i\in[n]} |X_i|] \le \max_{i\in[n]} \sigma_i \sqrt{2\log(2n)}.$$

Proof of Lemma 9 It is a simple extension of Theorem 1.14 of Rigollet (2015). □

LEMMA 10. *For Γ function and any integer n, we have*

$$\frac{n}{\sqrt{n+1}} \le \sqrt{2}\frac{\Gamma(\frac{n+1}{2})}{\Gamma(\frac{n}{2})} \le \sqrt{n}.$$

Proof of Lemma 10 It is easy to prove by induction. □

LEMMA 11. *Let $Z_1, \cdots, Z_n$ be independent matrices in $\mathbb{R}^{d_1\times d_2}$ s.t. $\mathbb{E}[Z_i] = 0$ and $\|Z_i\| \le D$ almost surely for all $i \in [n]$. Let σ_Z be such that*

$$\sigma_Z^2 \ge \max\left\{ \|\frac{1}{n}\sum_{i\in[n]} \mathbb{E}[Z_i^T Z_i]\|, \|\frac{1}{n}\sum_{i\in[n]} \mathbb{E}[Z_i Z_i^T]\| \right\}.$$

Then for any $t > 0$,

$$\mathbb{P}(\|\frac{1}{n}\sum_{i\in[n]} Z_i\| \ge t) \le (d_1+d_2)\exp(\frac{-nt^2}{2\sigma_Z^2 + (2Dt)/3}).$$

Proof. This is a simple extension of Proposition 1 of Athey et al. (2021). □

Appendix G: Experimental Details

This section provides details on the setup of experiments described in §5 on both synthetic and real data.

G.1. Synthetic Data

Experimental Details. We consider a low-data regime where the gold and proxy sample sizes are $n_g = 50$ and $n_p = 5{,}000$ respectively. The observation matrices $A_{p,i}, A_{g,i} \in \mathbb{R}^{d\times d}$ are independent gaussian random matrices whose entries are i.i.d. $N(0,1)$. The parameter $\Theta_p^* \in \mathbb{R}^{d\times d}$ is created by choosing $U_p^* \in \mathbb{R}^{d\times r}$ with i.i.d. $N(0,1)$ entries. Then, we generate Θ_g^* by setting the row sparsity of Δ_U^* to s, randomly picking s rows our of d, and drawing the value of each entry from a uniform distribution $\texttt{Uniform}[-1,1]$. Additionally, we sample noise terms $\epsilon_{p,i}, \epsilon_{g,i}$ independently from a gaussian distribution $N(0,1)$.

Cross Validation. We compute the gold, proxy, and transfer learning estimators by solving optimization problems (9), (10), and (3), respectively. To construct the transfer learning estimator, we also need to pick a proper value for the hyperparameter λ to balance bias and variance. Although we provide a theoretically justified expression for λ in Theorem 1, this choice depends on problem-dependent parameters that are typically not known in practice.

Rather, in practice, the hyperparameters are typically chosen using the popular cross-validation method (Kohavi et al. 1995, Hastie et al. 2009); specifically, in the context of low-rank matrix factorization and group-sparse regression, k-fold cross validation is typically used to tune hyperparameters (Huang and Zhang 2010, Chen et al. 2013, Cai et al. 2016). Here we use a 5-fold cross validation to tune λ on a pre-specified grid. In particular, we split the full gold sample into 5 parts; for each of the 5 times, we use 4 parts (i.e., 80% of the gold data) as the training set and calculate the Frobenius error on the remaining one part (i.e., 20% of the gold data), which is the validation set. Note that our estimates of the embeddings U_g^* are invariant under an orthogonal change-of-basis (see discussion in §2.1), so we use the rotation-invariant Frobenius norm to measure estimation error of Θ_g^*. We pick the value of λ that gives the lowest average Frobenius error across the 5 runs.

G.2. Wikipedia

Data Pre-processing. All the Wikipedia text data were downloaded from the English Wikipedia database dumps[6] in January 2020. We preprocess the text using a standard approach—i.e., splitting and tokenizing sentences, removing short sentences that contain less than 20 characters or 5 tokens, and removing stop words. We download the pre-trained word embeddings from GloVe's official website.[7] In our experiment, we use the pre-trained vectors trained from the 2014 Wikipedia dump and Gigaword 5, which contains around 6 billion tokens and 400K vocabulary words.

Experimental Details. We implement our transfer learning method based on (3). Note that the goal of our transfer learning approach is to efficiently use publicly available pre-trained word embeddings together with domain textual data. Since it's computationally costly to train pre-trained word embeddings based on our method from the whole Wikipedia dump, we use the GloVe pre-trained word embedding as our proxy estimator. Then, to extend our approach to the GloVe objective, we solve the optimization problem (8). The Mittens word embeddings are obtained solving a similar problem as (8), but with the Frobenius norm penalty—i.e.,

$$\sum_{i\in[d]} \|(U_i + V_i) - \widehat{U}_p^i\|^2.$$

We follow the experimental setting of GloVe. We create the co-occurrence matrix using a symmetric context window of length 5. We choose the dimension of the word embedding to be 100. We tune the hyperparameters for all methods and take $\lambda = 0.5$ for our estimator, and $\lambda = 0.05$ for Mittens estimator and our estimator adjusted to GloVe objective. We found our results to be robust to the choice of hyperparameter.

To identify domain-specific words for each article, we score each word i by the ℓ_2 distance between its new embedding (e.g, our transfer learning estimator or Mittens) and its pre-trained embedding; a higher score indicates a higher likelihood of being a domain-specific word. To evaluate the accuracy of domain-specific word identification, we choose a threshold of 10%, and select and compare the top 10% of words according to this score for each estimator. In other words, we treat all words in the top 10% as positives identified by each estimator, and accordingly the rest 90% of the words are negatives identified by each estimator. Then, we will be able to calculate an $F1$ score of each estimator for each article. To compare different estimators, we target a domain-level $F1$ score, which is an average across article-level $F1$ scores normalized by article length of articles within the corresponding domain.

Appendix H: Additional Experiments

This section details additional experiments to complement and extend the main experimental results in §5.

[6] `https://dumps.wikimedia.org/enwiki/latest/`

[7] `https://nlp.stanford.edu/projects/glove/`

H.1. Synthetic Data

We conduct additional experiments to show how our estimator's performance varies based on problem-specific parameters, as well as to assess the robustness of performance with respect to the choice of hyperparameters. The experimental results are shown in Figure 4 and 5. In these experiments, we inherit the setting (a) in Figure 3 with $d = 20$, $r = 5$, and $s = 2$, except for Figure 5 (a) and (b) where we take $d = 40$, $r = 5$, and $s = 2$.

First, aligned with our theory, Figure 4 shows the estimation error declines with the gold sample size for both the gold estimator and our transfer learning estimator. Note that the proxy estimator only depends on the proxy sample size and hence its performance does not vary as a function of the gold sample size. These results supports our theoretical finding that our estimator performs consistently better than other benchmarks in the low-data regime where $n_p \gg d^2$ and $n_g \ll d^2$ (in this example, $n_p = 5,000$ and $d = 20$). Intuitively, if the gold estimator has low estimation error, we can always set the hyperparameter λ to be 0 so that our transfer learning estimator equals the gold estimator. Therefore, with flexibility in tuning hyperparameters, our estimator should always weakly outperform the gold estimator in practice. More interestingly, our results also suggest that our estimator performs relatively well even in the regime with moderate gold sample size (in this example, when $n_g \geq d^2 = 400$).

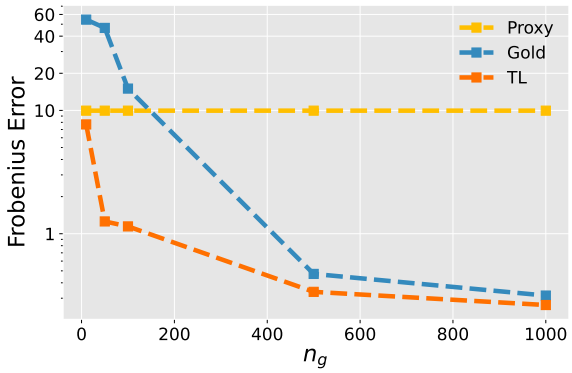
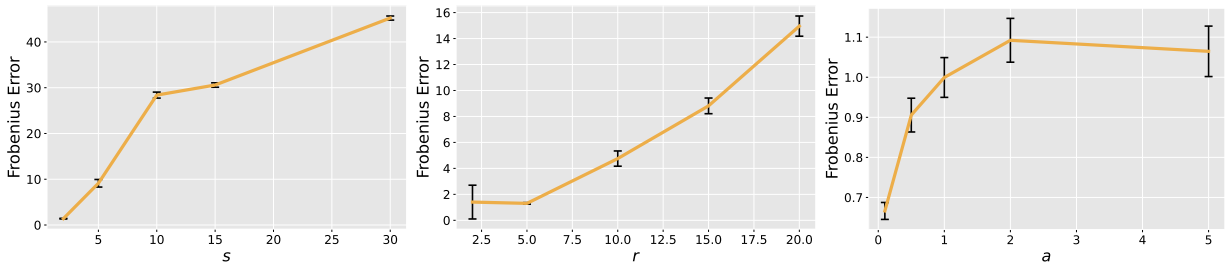


Figure 4 **Lines depict the Frobenius norm estimation errors averaged over 100 trials. The 95% confidence intervals are too small to be observed and hence not shown. 'TL' denotes our transfer learning estimator.**

Next, also consistent with our theory, we show that the estimation error of our estimator increases with the group sparsity level s, the matrix rank r, and the magnitude of Δ_U^*, i.e., L, (remember $\|\Delta_U^*\|_{2,1} \leq L$). Figure 5 (a) shows the estimation error of our method increases with the group sparsity level s. Intuitively, when the gold and proxy tasks become more heterogeneous—e.g., a higher sparsity level implying that less information can be shared—transfer learning becomes harder. Figure 5 (b) analyzes the performance of our method with different values of matrix rank r. Intuitively, higher matrix rank r means more within-group parameters to learn, and thus will increase the learning difficulty for all estimators. Figure 5 (c) shows our estimation error for different values of L. Specifically, we draw the value of each entry of the s nonzero rows of Δ_U^* from a uniform distribution $\texttt{Uniform}[-a, a]$. Thus, L becomes larger when a takes larger values. Again, this result shows that transfer learning becomes harder when the gold and proxy problems are more



(a) Varying s (b) Varying r (c) Varying a

Figure 5 Lines depict the Frobenius norm estimation errors of our transfer learning estimator averaged over 100 trials, with error bars the corresponding 95% confidence intervals.

different. Note that when a takes smaller values, Δ_U^* is easier to estimate, and the result here is consistent with that in Figure 5 (a).

Lastly, we study the robustness of our estimator towards the hyperparameters. Figure 6 shows the Frobenius norm estimation error of our transfer learning estimator with varying values of the hyperparameter λ, compared with the benchmark errors of proxy and gold estimators. We find that the Frobenius error of our method is not substantially affected by varying values of the hyperparameter; particularly, our method still dominates the two other benchmarks consistently over different values of λ. This suggests that our algorithm is robust, which is important especially in empirical applications where these hyperparameters might not be well specified.

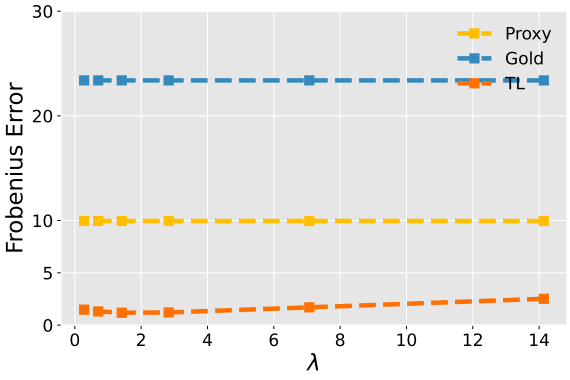


Figure 6 Lines depict the Frobenius norm estimation errors averaged over 100 trials. The 95% confidence intervals are too small to be observed and hence not shown. 'TL' denotes our transfer learning estimator.

H.2. Wikipedia

As discussed in the main text, we also compare our algorithm with two other benchmarks that combine domain-specific word embeddings with pre-trained ones through Canonical Correlation Analysis (CCA) or the closely related kernelized version KCCA (Sarma et al. 2018). In the following, without raising any ambiguity, we will call these two benchmarks CCA and KCCA estimators. We show our approach outperforms these two benchmarks as well.

Specifically, the CCA estimator implements CCA to align the domain specific word embedding $\widehat{U}_{g,i}$ and the pre-trained word embedding $\widehat{U}_{p,i}$ for each word i, transforming them to $\bar{U}_{g,i}$ and $\bar{U}_{p,i}$ respectively. The final proposed CCA estimator (Sarma et al. 2018) is computed as $\frac{1}{2}\bar{U}_{g,i} + \frac{1}{2}\bar{U}_{p,i}$ for each word i, i.e., the average of the aligned domain-specific word embeddings and the pre-trained word embeddings. In contrast, the KCCA estimator transforms the embeddings $\widehat{U}_{g,i}$ and $\widehat{U}_{p,i}$ for word i through a kernel CCA, instead of CCA. We follow Sarma et al. (2018) setting the hyperparameter σ of the gaussian kernel to be the median of pairwise distances between domain-specific word embeddings and pre-trained word embeddings. See Sarma et al. (2018) for details of the implementations of both estimators.

As illustrated in Table H.1, our approach outperforms these two algorithms at the task of identifying domain-specific words across different types of Wikipedia articles.

Domain	TL	Mittens	CCA	KCCA	Random
Finance	**0.2320**	0.1829	0.1347	0.1633	0.1376
Math	**0.2660**	0.2175	0.2385	0.1690	0.1543
Computing	**0.2527**	0.1963	0.1980	0.2319	0.1430
Politics	**0.1873**	0.1571	0.0602	0.1373	0.0640

Table H.1 Average $F1$-score of domain word identification (weighted by article length) for four domains respectively. "TL" represents our transfer learning approach.

In addition, we also evaluate how our transfer learning estimator performs when varying the value of selection threshold, which determines the criteria for domain-specific words; in particular, we consider 10%, 20%, and 30% (note our main experimental result uses a threshold of 10%). Figure 7 shows the weighted F_1-score versus the top percentage set for the threshold in the finance domain. Our approach consistently outperforms all baselines including CCA and KCCA over different selection thresholds, illustrating that it is robust to how we define domain-specific words.

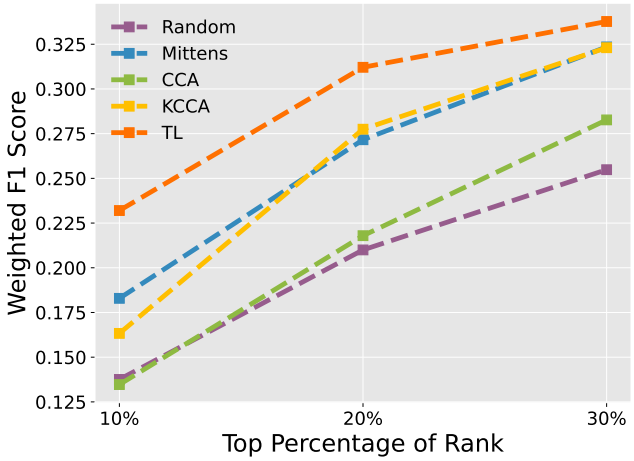


Figure 7 Average $F1$ score (weighted by article length) versus top percentage of the rank set for the threshold in the finance domain. "TL" represents our transfer learning approach.