Uncorrelation and Evenness: a New Diversity-Promoting Regularizer

Pengtao Xie [1 2] **Aarti Singh** [1] **Eric P. Xing** [2]

Abstract

Latent space models (LSMs) provide a principled and effective way to extract hidden patterns from observed data. To cope with two challenges in LSMs: (1) how to capture infrequent patterns when pattern frequency is imbalanced and (2) how to reduce model size without sacrificing their expressiveness, several studies have been proposed to "diversify" LSMs, which design regularizers to encourage the components therein to be "diverse". In light of the limitations of existing approaches, we design a new diversity-promoting regularizer by considering two factors: *uncorrelation* and *evenness*, which encourage the components to be uncorrelated and to play equally important roles in modeling data. Formally, this amounts to encouraging the covariance matrix of the components to have more uniform eigenvalues. We apply the regularizer to two LSMs and develop an efficient optimization algorithm. Experiments on healthcare, image and text data demonstrate the effectiveness of the regularizer.

1. Introduction

A fundamental task in machine learning (ML) is to discover latent patterns underlying data, for instance, extracting *topics* from documents and *communities* from social networks. Latent space models (Bishop, 1998; Knott & Bartholomew, 1999; Blei, 2014) are effective tools to accomplish this task. An LSM contains a collection of learnable components such as *hidden units* in neural networks and *factors* in factor analysis (Harman, 1960). Each component is aimed at capturing a hidden pattern. In most LSMs, components are parameterized by vectors.

Among the many challenges encountered in latent space modeling, two of them are of particular interest to us. First, under many circumstances, the frequency of patterns is highly imbalanced. Some patterns have very high frequency while others occur less frequently. As a typical example, in a news corpus, politics and economics are frequent topics (patterns) while furniture and gardening are infrequent. Classic LSMs are sensitive to the skewness of pattern frequency and less capable of capturing the infrequent patterns (Wang et al., 2014). Second, when using LSMs, one needs to carefully balance the tradeoff between model size (precisely, the number of components) and modeling power (Xie, 2015). Larger-sized LSMs are more expressive, but incur higher computational complexity. It is desirable but challenging to achieve sufficient modeling power with a small number of components.

To address these two challenges, recent studies (Zou & Adams, 2012; Cogswell et al., 2015; Xie et al., 2015; 2016) investigate a "diversification" strategy which encourages the components in LSMs to be mutually different, either through frequentist-style regularization (Zou & Adams, 2012; Cogswell et al., 2015; Xie et al., 2015) or Bayesian learning (Xie et al., 2016). They conjecture that: (1) through "diversification", some components that are originally aggregated around frequent patterns can be pushed apart to cover infrequent patterns; (2) "diversified" components bear less redundancy and are mutually complementary; a small number of such components are sufficient to model data well.

Along this line of research, several diversity-promoting regularizers have been proposed, based upon determinantal point process (Kulesza & Taskar, 2012; Zou & Adams, 2012), cosine similarity (Yu et al., 2011; Bao et al., 2013; Xie et al., 2015) and covariance (Malkin & Bilmes, 2008; Cogswell et al., 2015). While these regularizers demonstrate notable efficacy, they have certain limitations, such as sensitivity to vector scaling (Zou & Adams, 2012; Malkin & Bilmes, 2008), inability to measure diversity in a global manner (Yu et al., 2011; Bao et al., 2013; Xie et al., 2015) and computational inefficiency (Cogswell et al., 2015). To address these limitations, we propose a new diversity-promoting regularizer gaining inspiration from principal component analysis (Jolliffe, 2002), biological diversity (Magurran, 2013) and information theory (Cover & Thomas, 2012).

[1]Machine Learning Department, Carnegie Mellon University [2]Petuum Inc. Correspondence to: Pengtao Xie <pengtaox@cs.cmu.edu>, Eric P. Xing <eric.xing@petuum.com>.

Proceedings of the 34^{th} International Conference on Machine Learning, Sydney, Australia, PMLR 70, 2017.

We characterize "diversity" by considering two factors: *uncorrelation* and *evenness*. Uncorrelation (Cogswell et al., 2015) encourages the components to be uncorrelated, such that each component can independently capture a unique pattern. Evenness is inspired from biological diversity (Magurran, 2013) where an ecosystem is deemed to be more diverse if different species contribute equally to the maintenance of biological balance. Analogously, when measuring component diversity, we assign an "importance" score to each component and encourage these scores to be even. In the context of latent space modeling, evenness ensures each component plays a significant role in pattern discovery rather than being dominated by others.

We study uncorrelation and evenness from a statistical perspective. The components are considered as random variables and the eigenvalues of their covariance matrix can be leveraged to characterize these two factors. First, according to Principle Component Analysis (Jolliffe, 2002), the disparity of eigenvalues reflects the correlation among components: the more uniform the eigenvalues, the less correlated the components. Second, eigenvalues represent the variance along principal directions and can be used to measure the "importance" of components. Promoting uniform importance amounts to encouraging evenness among eigenvalues.

To promote uniformity among the eigenvalues, we encourage the discrete distribution parametrized by the normalized eigenvalues to have small Kullback-Leibler divergence with the uniform distribution, based on which, we define a *uniform eigenvalue regularizer* (UER) and make a connection with the von Neumann entropy (Bengtsson & Zyczkowski, 2007) and with the von Neumann divergence (Kulis et al., 2009). We apply UER to two LSMs – distance metric learning (DML) (Xing et al., 2002) and long short-term memory (LSTM) network (Hochreiter & Schmidhuber, 1997) – to encourage their components to be diverse and develop an efficient optimization algorithm. Experiments on healthcare, image and text data demonstrate that UER (1) greatly improves the performance of LSMs; (2) better captures infrequent patterns; (3) reduces model size without sacrificing modeling power; (4) outperforms other diversity-promoting regularizers.

The major contributions of this paper are:

- We propose a new diversity-promoting regularizer from the perspectives of uncorrelation and evenness.
- We propose to simultaneously promote uncorrelation and evenness by encouraging uniformity among the eigenvalues of the covariance matrix of components.
- We develop an efficient projected gradient descent algorithm to solve UE regularized LSM problems.
- In experiments, we demonstrate the effectiveness of this regularizer on two LSMs: DML and LSTM.

The rest of the paper is organized as follows. Section 2 reviews related works. Section 3 introduces the uniform eigenvalue regularizer. Section 4 presents experimental results and Section 5 concludes the paper.

2. Related Works

Diversity promoting regularization has been widely used in classification (Malkin & Bilmes, 2008), ensemble learning (Yu et al., 2011) and latent space modeling (Zou & Adams, 2012; Xie et al., 2015; 2017). In the sequel, we present a brief review of existing diversity-promoting regularizers. Several regularizers (Yu et al., 2011; Bao et al., 2013; Xie et al., 2015; 2017) are based on pairwise dissimilarity of components: if every two components are dissimilar, then overall the set of components are "diverse". Given the weight vectors $\{\mathbf{a}_j\}_{j=1}^m$ of m components, Yu et al. (2011) define the regularizer as $\sum_{1\leq j<k\leq m}(1-c_{jk})$, where c_{jk} is the cosine similarity between component j and k. In (Bao et al., 2013), the score is defined as $-\log(\frac{1}{m(m-1)}\sum_{1\leq j<k\leq m}\beta|c_{jk}|)^{\frac{1}{\beta}}$ where $\beta>0$. In (Xie et al., 2015), the score is defined as mean of $\{\arccos(|c_{jk}|)\}$ minus the variance of $\{\arccos(|c_{jk}|)\}$, where the variance term is utilized to encourage the dissimilarity scores $\{\arccos(|c_{jk}|)\}$ to be even. Xie et al. (2017) define the regularizer as $\sum_{1\leq i<j\leq m}k(\mathbf{a}_i,\mathbf{a}_j)$ where $k(\cdot,\cdot)$ is a kernel function. These regularizers are applied to classifiers ensemble, neural network and restricted Boltzmann machine. While these regularizers can capture pairwise dissimilarities between components, they are unable to capture higher-order "diversity".

Determinantal Point Process (DPP) (Kulesza & Taskar, 2012) was used by (Zou & Adams, 2012; Mariet & Sra, 2015) to encourage the topic vectors in Latent Dirichlet Allocation (Blei et al., 2003), Gaussian mean vectors in Gaussian Mixture Model and hidden units in neural network to be "diverse". The DPP regularizer is defined as $-\log\det(\mathbf{L})$, where $\mathbf{L}$ is a $m\times m$ kernel matrix and $\det(\cdot)$ denotes the determinant of the matrix. L_{ij} equals to $k(\mathbf{a}_i,\mathbf{a}_j)$ and $k(\cdot,\cdot)$ is a kernel function. In geometry, $\det(\mathbf{L})$ is the volume of the parallelepiped formed by vectors in the feature space associated with kernel k. Vectors that result in a larger volume are considered to be more "diverse". Since volume depends on all vectors simultaneously, DPP is able to measure diversity in a global way. The drawback of DPP lies in its sensitivity to the scaling of vectors. The volume increases with the ℓ_2 norm of vectors, but "diversity" does not. Malkin & Bilmes (2008) propose to promote diversity by maximizing the determinant of vectors' covariance matrix. Similar to DPP, this regularizer is sensitive to vector scaling.

Unlike the aforementioned regularizers which are defined directly on weight vectors, Cogswell et al. (2015) design a

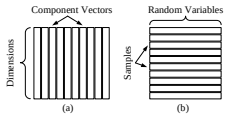

Figure 1. Two views of the component matrix

regularizer on hidden activations in the neural network and influence the parameters indirectly. The number of hidden activations could be much larger than that of weight parameters (like in a convolutional neural network), which may render this regularizer to be computationally inefficient.

3. Method

In this section, we develop a uniform eigenvalue regularizer and apply it to promote "diversity" in two LSMs.

3.1. Uniform Eigenvalue Regularizer

A latent space model (LSM) is equipped with a set of m components and each component is represented with a vector $\mathbf{a} \in \mathbb{R}^d$. To achieve broader coverage of infrequent patterns and reduce model size without sacrificing modeling power, previous works (Zou & Adams, 2012; Xie et al., 2015) propose to "diversify" the components by imposing a regularizer over them.

As a subjective concept, "diversity" has been defined in various ways as reviewed in Section 2. In this paper, we define a new measure of "diversity" by taking two factors into consideration: *uncorrelation* and *evenness*. Uncorrelation is a measure of how uncorrelated the components are. Literally, less correlation is equivalent to more diversity. Evenness is borrowed from biological diversity (Magurran, 2013), which measures how equally important different species are in maintaining the ecological balance within an ecosystem. If no species dominates another, the ecosystem is deemed as more diverse. Likewise, in latent space modeling, we desire the components to play equally important roles and no one dominates another, such that each component contributes significantly to the modeling of data.

We characterize the uncorrelation among components from a statistical perspective: treating the components as random variables and measuring their covariance which is proportional to their correlation. Let $\mathbf{A} \in \mathbb{R}^{d\times m}$ denote the component matrix where in the k-th column is the parameter vector $\mathbf{a}_k$ of component k. Alternatively, we can take a row view (Figure 1(b)) of $\mathbf{A}$: each component is treated as a random variable and each row vector $\tilde{\mathbf{a}}_i^\top$ can be seen as a sample drawn from the random vector formed by the m components. Let $\boldsymbol{\mu} = \frac{1}{d}\sum_{i=1}^d \tilde{\mathbf{a}}_i = \frac{1}{d}\mathbf{A}^\top\mathbf{1}$ be the sample mean, where the elements of $\mathbf{1} \in \mathbb{R}^d$ are all 1. We compute the empirical covariance matrix of the components as

$$\begin{aligned}\mathbf{G} &= \tfrac{1}{d}\textstyle\sum_{i=1}^d(\tilde{\mathbf{a}}_i - \boldsymbol{\mu})(\tilde{\mathbf{a}}_i - \boldsymbol{\mu})^\top \\ &= \tfrac{1}{d}\mathbf{A}^\top\mathbf{A} - (\tfrac{1}{d}\mathbf{A}^\top\mathbf{1})(\tfrac{1}{d}\mathbf{A}^\top\mathbf{1})^\top\end{aligned} \quad (1)$$

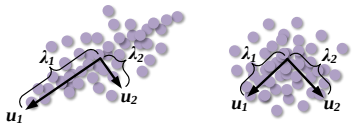

Figure 2. When the principal directions ($\mathbf{u}_1$ and $\mathbf{u}_2$) are not aligned with the coordinate axis, the level of disparity between the eigenvalues (λ_1 and λ_2) indicates the correlation between random variables (components).

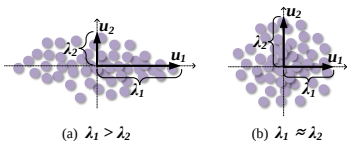

Figure 3. When the principal directions ($\mathbf{u}_1$ and $\mathbf{u}_2$) are aligned with the coordinate axis, the magnitude of eigenvalues represents the importance of components.

Imposing the constraint $\mathbf{A}^\top\mathbf{1} = \mathbf{0}$, we have $\mathbf{G} = \frac{1}{d}\mathbf{A}^\top\mathbf{A}$. Suppose $\mathbf{A}$ is a full rank matrix and $m < d$, then $\mathbf{G}$ is a full-rank matrix with rank m.

For the next step, we show that the eigenvalues of $\mathbf{G}$ play important roles in characterizing the uncorrelation and evenness of components. We start with uncorrelation. Let $\mathbf{G} = \sum_{k=1}^m \lambda_k\mathbf{u}_k\mathbf{u}_k^\top$ be the eigendecomposition where λ_k is an eigenvalue and $\mathbf{u}_k$ is the associated eigenvector. As is well known in Principle Component Analysis (Jolliffe, 2002), an eigenvector $\mathbf{u}_k$ of the covariance matrix $\mathbf{G}$ represents a principal direction of the data points and the associated eigenvalue λ_k tells the variability of points along that direction. As shown in Figure 2(a), the larger λ_k is, the more spread out the points along the direction $\mathbf{u}_k$. When the eigenvectors (principal directions) are not aligned with coordinate axis (as shown in Figure 2), the level of disparity among eigenvalues indicates the level of correlation among the m components (random variables). The more different the eigenvalues are, the higher the correlation is. As shown in Figure 2(a), λ_1 is about three times larger than λ_2 and there is a high correlation along the direction $\mathbf{u}_1$. On the other hand, in Figure 2(b), the two eigenvalues are close to each other and the points evenly spread out in both directions with negligible correlation. In light of this, we would utilize the uniformity among eigenvalues of $\mathbf{G}$ to measure how uncorrelated the components are.

Secondly, we relate the eigenvalues with the other factor of diversity: evenness. When the eigenvectors are aligned with the coordinate axis (as shown in Figure 3(a)), the components are uncorrelated. In this case, we bring in evenness to measure diversity. As stated earlier, we first need to assign each component an importance score. Since the eigenvectors are in parallel to the coordinate axis, the eigenvalues reflect the variance of components. Analogous to PCA which posits that random variables with larger variance are

more important, we use variance to measure importance. As shown in Figure 3(a), component 1 has a larger eigenvalue λ_1 and accordingly larger variability, hence is more important than component 2. According to the evenness criteria, the components are more diverse if their importance match, which motivates us to encourage the eigenvalues to be uniform. As shown in Figure 3(b), the two eigenvalues are close and the two components have roughly the same variability, hence are similarly important.

To sum up, we desire to encourage the eigenvalues to be even in both cases: (1) when the eigenvectors are not aligned with the coordinate axis, they are preferred to be even to reduce the correlation of components; (2) when the eigenvectors are aligned with the coordinate axis, they are encouraged to be even such that different components contribute equally in modeling data. Previously, encouraging evenness among variances (eigenvalues) is investigated in other problems, such as learning compact representations for efficient hashing (Kong & Li, 2012; Ge et al., 2013).

Next, we discuss how to promote uniformity among eigenvalues. The basic idea is: we normalize the eigenvalues into a probability simplex and encourage the discrete distribution parameterized by the normalized eigenvalues to have small Kullback-Leibler (KL) divergence with the uniform distribution. Given the eigenvalues $\{\lambda_k\}_{k=1}^m$, we first normalize them into a probability simplex $\widehat{\lambda}_k = \frac{\lambda_k}{\sum_{j=1}^m \lambda_j}$ based on which we define a distribution on a discrete random variable $X = 1, \cdots, m$ where $p(X = k) = \widehat{\lambda}_k$. In addition, to guarantee the eigenvalues are strictly positive, we require $\mathbf{A}^\top\mathbf{A}$ to be positive definite. To encourage $\{\widehat{\lambda}_k\}_{k=1}^m$ to be uniform, we encourage the distribution $p(X)$ to be "close" to a uniform distribution $q(X = k) = \frac{1}{m}$, where the "closeness" is measured using KL divergence $KL(p||q)$: $\sum_{k=1}^m \widehat{\lambda}_k \log \frac{\widehat{\lambda}_k}{1/m} = \frac{\sum_{k=1}^m \lambda_k \log \lambda_k}{\sum_{j=1}^m \lambda_j} - \log \sum_{j=1}^m \lambda_j + \log m$. In this equation, $\sum_{k=1}^m \lambda_k \log \lambda_k$ is equivalent to $\mathrm{tr}((\frac{1}{d}\mathbf{A}^\top\mathbf{A})\log(\frac{1}{d}\mathbf{A}^\top\mathbf{A}))$, where $\log(\cdot)$ denotes matrix logarithm. To show this, note that $\log(\frac{1}{d}\mathbf{A}^\top\mathbf{A}) = \sum_{k=1}^m \log(\lambda_k)\mathbf{u}_k\mathbf{u}_k^\top$, according to the property of matrix logarithm. Then we have $\mathrm{tr}((\frac{1}{d}\mathbf{A}^\top\mathbf{A})\log(\frac{1}{d}\mathbf{A}^\top\mathbf{A}))$ equals to $\mathrm{tr}((\sum_{k=1}^m \lambda_k\mathbf{u}_k\mathbf{u}_k^\top)(\sum_{k=1}^m \log(\lambda_k)\mathbf{u}_k\mathbf{u}_k^\top))$ which equals to $\sum_{k=1}^m \lambda_k \log \lambda_k$. According to the property of trace, we have $\mathrm{tr}(\frac{1}{d}\mathbf{A}^\top\mathbf{A}) = \sum_{k=1}^m \lambda_k$. Then the KL divergence can be turned into a diversity-promoting uniform eigenvalue regularizer (UER):

$$\frac{\mathrm{tr}((\frac{1}{d}\mathbf{A}^\top\mathbf{A})\log(\frac{1}{d}\mathbf{A}^\top\mathbf{A}))}{\mathrm{tr}(\frac{1}{d}\mathbf{A}^\top\mathbf{A})} - \log\mathrm{tr}(\frac{1}{d}\mathbf{A}^\top\mathbf{A}) \quad (2)$$

subject to $\mathbf{A}^\top\mathbf{A} \succ 0$ and $\mathbf{A}^\top\mathbf{1} = 0$. Compared with previous diversity-promoting regularizers, UER has the following benefits: (1) It measures the diversity of all components in a holistic way, rather than reducing to pairwise dissimilarities as other regularizers (Yu et al., 2011; Bao et al., 2013; Xie et al., 2015) do. This enables UER to capture global relations among components. (2) Unlike determinant-based regularizers (Malkin & Bilmes, 2008; Zou & Adams, 2012) that are sensitive to vector scaling, UER is derived from normalized eigenvalues where the normalization effectively removes scaling. (3) UER is amenable for computation. First, unlike DoCev (Cogswell et al., 2015) that is defined over data-dependent intermediate variables incurring computational inefficiency, UER is directly defined on model parameters independent of data. Second, unlike the regularizers proposed in (Bao et al., 2013; Xie et al., 2015) that are non-smooth, UER is a smooth function. The dominating computation in UER is the matrix logarithm. It does not substantially increase computational overhead as long as the number of components is not too large (e.g., less than 1000).

We apply UER to promote diversity in LSMs. Let $\mathcal{L}(\mathbf{A})$ denote the objective function of an LSM, then an UE-regularized LSM problem can be defined as

$$\begin{aligned} \min_{\mathbf{A}} \quad & \mathcal{L}(\mathbf{A}) + \lambda(\frac{\mathrm{tr}((\frac{1}{d}\mathbf{A}^\top\mathbf{A})\log(\frac{1}{d}\mathbf{A}^\top\mathbf{A}))}{\mathrm{tr}(\frac{1}{d}\mathbf{A}^\top\mathbf{A})} - \log\mathrm{tr}(\frac{1}{d}\mathbf{A}^\top\mathbf{A})) \\ s.t. \quad & \mathbf{A}^\top\mathbf{1} = \mathbf{0},\ \mathbf{A}^\top\mathbf{A} \succ 0 \end{aligned}$$

where λ is the regularization parameter. Similar to other diversity-promoting regularizers, UER is non-convex. Since $\mathcal{L}(\mathbf{A})$ in most LSMs is non-convex, adding UER does not substantially increase difficulty for optimization.

Connection with von Neumann Entropy In this section, we make a connection between UER and von Neumann entropy. A matrix $\mathbf{M}$ is referred to as a density matrix (Bengtsson & Zyczkowski, 2007) if its eigenvalues are strictly positive and sum to one, equivalently, $\mathbf{M} \succ 0$ and $\mathrm{tr}(\mathbf{M}) = 1$. The von Neumann entropy (Bengtsson & Zyczkowski, 2007) of $\mathbf{M}$ is defined as $S(\mathbf{M}) = -\mathrm{tr}(\mathbf{M}\log\mathbf{M})$, which is essentially the Shannon entropy of its eigenvalues. If the covariance matrix $\mathbf{G}$ of components is a density matrix, then we can use its von Neumann entropy to define a UER. To encourage the eigenvalues $\{\lambda_k\}_{k=1}^m$ of $\mathbf{G}$ to be even, we directly encourage the KL divergence between the distribution parameterized by the eigenvalues (without normalization) and the uniform distribution to be small: $\sum_{k=1}^m \lambda_k \log \frac{\lambda_k}{1/m} = \sum_{k=1}^m \lambda_k \log \lambda_k + \log m$, which is equivalent to encouraging the Shannon entropy of the eigenvalues $-\sum_{k=1}^m \lambda_k \log \lambda_k$, i.e., the von Neumann entropy of $\mathbf{G}$ to be large. Then a new UER can be defined as the negative von Neumann entropy of $\mathbf{G}$: $\mathrm{tr}((\frac{1}{d}\mathbf{A}^\top\mathbf{A})\log(\frac{1}{d}\mathbf{A}^\top\mathbf{A}))$, subject to the constraints: (1) $\mathbf{A}^\top\mathbf{A} \succ 0$; (2) $\mathrm{tr}(\frac{1}{d}\mathbf{A}^\top\mathbf{A}) = 1$; (3) $\mathbf{A}^\top\mathbf{1} = \mathbf{0}$. This new UER is a special case of the previous one (Eq.(2)).

Connection with von Neumann Divergence Next we make a connection between the UER and von Neumann divergence (Kulis et al., 2009). Given two positive defi-

nite matrices $\mathbf{X}$ and $\mathbf{Y}$, their von Neumann divergence is defined as $\text{tr}(\mathbf{X}\log\mathbf{X} - \mathbf{X}\log\mathbf{Y} - \mathbf{X} + \mathbf{Y})$, which measures the closeness between the two matrices. Given two vectors $\mathbf{x}, \mathbf{y} \in \mathbb{R}^m$, their generalized KL divergence can be defined as $\sum_{k=1}^m x_k \log(\frac{x_k}{y_k}) - (x_k - y_k)$, which measures the closeness between two vectors. To encourage uniformity among the eigenvalues of the covariance matrix $\mathbf{G}$, we can decrease the generalized KL divergence between these eigenvalues and an all-1 vector:

$$\begin{aligned}&\textstyle\sum_{k=1}^m \lambda_k \log(\frac{\lambda_k}{1}) - (\lambda_k - 1)\\ &= \text{tr}((\tfrac{1}{d}\mathbf{A}^\top\mathbf{A})\log(\tfrac{1}{d}\mathbf{A}^\top\mathbf{A})) - \text{tr}(\tfrac{1}{d}\mathbf{A}^\top\mathbf{A}) + m\end{aligned} \tag{3}$$

which is the von Neumann divergence between $\mathbf{G}$ and an identity matrix. Hence, encouraging uniformity among eigenvalues can be achieved by making $\mathbf{G}$ to be close to an identity matrix based on the von Neumann divergence.

3.2. Case Studies

In this section, we apply the uniform eigenvalue regularizer to promote diversity in two latent space models: DML and LSTM. We also applied it to latent Dirichlet allocation (Blei et al., 2003) and classifier ensemble (Yu et al., 2011). Due to space limit, the results of the latter two are deferred to the supplements.

Distance Metric Learning (DML) Given data pairs either labeled as "similar" or "dissimilar", DML (Xing et al., 2002; Davis et al., 2007; Guillaumin et al., 2009) aims to learn a distance metric under which similar pairs would be placed close to each other and dissimilar pairs are separated apart. The learned distance can benefit a wide range of tasks, including retrieval, clustering and classification. Following (Weinberger & Saul, 2009), we define the distance metric between $\mathbf{x}, \mathbf{y} \in \mathbb{R}^d$ as $\|\mathbf{A}^\top\mathbf{x} - \mathbf{A}^\top\mathbf{y}\|_2^2$ where $\mathbf{A} \in \mathbb{R}^{d\times m}$ is a parameter matrix whose column vectors are components. Built upon the DML formulation in (Xie, 2015), an uniform-eigenvalue regularized DML (DML-UE) problem can be formulated as

$$\begin{aligned}\min_{\mathbf{A}} \quad & \sum_{(\mathbf{x},\mathbf{y})\in\mathcal{S}} \|\mathbf{A}^\top\mathbf{x} - \mathbf{A}^\top\mathbf{y}\|_2^2\\ & + \sum_{(\mathbf{x},\mathbf{y})\in\mathcal{D}} \max(0, 1 - \|\mathbf{A}^\top\mathbf{x} - \mathbf{A}^\top\mathbf{y}\|_2^2)\\ & + \lambda(\frac{\text{tr}((\frac{1}{d}\mathbf{A}^\top\mathbf{A})\log(\frac{1}{d}\mathbf{A}^\top\mathbf{A}))}{\text{tr}(\frac{1}{d}\mathbf{A}^\top\mathbf{A})} - \log\text{tr}(\tfrac{1}{d}\mathbf{A}^\top\mathbf{A}))\\ s.t. \quad & \mathbf{A}^\top\mathbf{1} = \mathbf{0},\ \mathbf{A}^\top\mathbf{A} \succ 0\end{aligned} \tag{4}$$

where $\mathcal{S}$ and $\mathcal{D}$ are the set of similar and dissimilar pairs respectively. The first and second term in the objective function encourage similar pairs to have small distance and dissimilar pairs to have large distance respectively. The learned metrics are applied for information retrieval.

Long Short-Term Memory (LSTM) Network LSTM (Hochreiter & Schmidhuber, 1997) is a type of recurrent neural network, that is better at capturing long-term dependency in sequential modeling. At each time step t where the input is $\mathbf{x}_t$, there is an input gate $\mathbf{i}_t$, a forget gate $\mathbf{f}_t$, an output gate $\mathbf{o}_t$, a memory cell $\mathbf{c}_t$ and a hidden state $\mathbf{h}_t$. The transition equations among them are

$$\begin{aligned}&\mathbf{i}_t = \sigma(\mathbf{W}^{(i)}\mathbf{x}_t + \mathbf{U}^{(i)}\mathbf{h}_{t-1} + \mathbf{b}^{(i)})\\ &\mathbf{f}_t = \sigma(\mathbf{W}^{(f)}\mathbf{x}_t + \mathbf{U}^{(f)}\mathbf{h}_{t-1} + \mathbf{b}^{(f)})\\ &\mathbf{o}_t = \sigma(\mathbf{W}^{(o)}\mathbf{x}_t + \mathbf{U}^{(o)}\mathbf{h}_{t-1} + \mathbf{b}^{(o)})\\ &\mathbf{c}_t = \mathbf{i}_t \odot \tanh(\mathbf{W}^{(c)}\mathbf{x}_t + \mathbf{U}^{(c)}\mathbf{h}_{t-1} + \mathbf{b}^{(c)}) + \mathbf{f}_t \odot \mathbf{c}_{t-1}\\ &\mathbf{h}_t = \mathbf{o}_t \odot \tanh(\mathbf{c}_t)\end{aligned}$$

where $\mathcal{W} = \{\mathbf{W}^{(s)}|s \in S = \{i, f, o, c\}\}$ and $\mathcal{U} = \{\mathbf{U}^{(s)}|s \in S\}$ are gate-specific weight matrices and $\mathcal{B} = \{\mathbf{b}^{(s)}|s \in S\}$ are bias vectors. The row vectors in $\mathbf{W}$ and $\mathbf{U}$ are treated as components. Let $\mathcal{L}(\mathcal{W}, \mathcal{U}, \mathcal{B})$ denote the loss function of an LSTM network and $\mathcal{R}(\cdot)$ denote the UER (including constraints), then a UE-regularized LSTM problem can be defined as

$$\min_{\mathcal{W},\mathcal{U},\mathcal{B}} \quad \mathcal{L}(\mathcal{W}, \mathcal{U}, \mathcal{B}) + \lambda\textstyle\sum_{s\in S}(\mathcal{R}(\mathbf{W}^{(s)}) + \mathcal{R}(\mathbf{U}^{(s)})) \tag{5}$$

The LSTM network is applied for cloze-style reading comprehension (CSRC). The network architecture follows that in (Seo et al., 2017), which achieves the state of the art performance on CSRC.

3.3. Algorithm

We develop a projected gradient descent (PGD) algorithm to solve the UE-regularized LSM problem in Eq.(5). The constraint $\mathbf{A}^\top\mathbf{A} \succ 0$ ensures the eigenvalues of $\mathbf{A}^\top\mathbf{A}$ are positive, such that $\log(\mathbf{A}^\top\mathbf{A})$ is well-defined. However, it makes optimization very nasty. To address this issue, we add a small perturbation $\epsilon\mathbf{I}$ over $\mathbf{A}^\top\mathbf{A}$ where ϵ is a close-to-zero positive scalar and $\mathbf{I}$ is an identity matrix, to ensure $\log(\mathbf{A}^\top\mathbf{A} + \epsilon\mathbf{I})$ is always well-defined. Accordingly, the constraint $\mathbf{A}^\top\mathbf{A} \succ 0$ can be eliminated. The PGD algorithm iteratively performs three steps: (1) compute (sub)gradient $\triangle\mathbf{A}$ of the objective function; (2) update $\mathbf{A}$ using gradient descent: $\widetilde{\mathbf{A}} \leftarrow \mathbf{A} - \eta \triangle \mathbf{A}$; (3) project $\widetilde{\mathbf{A}}$ to the constraint set $\{\mathbf{A}|\mathbf{A}^\top\mathbf{1} = \mathbf{0}\}$. In step (1), the derivative of $\text{tr}((\frac{1}{d}\mathbf{A}^\top\mathbf{A} + \epsilon\mathbf{I})\log(\frac{1}{d}\mathbf{A}^\top\mathbf{A} + \epsilon\mathbf{I}))$ is $\frac{2}{d}\mathbf{A}(\log(\frac{1}{d}\mathbf{A}^\top\mathbf{A} + \epsilon\mathbf{I}) + \mathbf{I})$. To compute the logarithm of $\frac{1}{d}\mathbf{A}^\top\mathbf{A} + \epsilon\mathbf{I}$, we perform an eigen-decomposition of this matrix into $\mathbf{U}\mathbf{\Lambda}\mathbf{U}^\top$, transform $\mathbf{\Lambda}$ into another diagonal matrix $\widetilde{\mathbf{\Lambda}}$ where $\widetilde{\Lambda}_{jj} = \log(\Lambda_{jj})$ and then compute $\log(\frac{1}{d}\mathbf{A}^\top\mathbf{A} + \epsilon\mathbf{I})$ as $\mathbf{U}\widetilde{\mathbf{\Lambda}}\mathbf{U}^\top$. The complexity of eigen-decomposing this m-by-m matrix is $O(m^3)$. In our applications, m is no more than 500, so $O(m^3)$ is not a big bottleneck. In addition, this matrix is symmetric and the symmetry can be leveraged for fast eigen-decomposition. In implementation, we use the MAGMA library that supports efficient eigen-decomposition of symmetric matrices on both CPUs and GPUs. In step (3), the projection operation amounts to solving the following problem: $\min_{\mathbf{A}} \frac{1}{2}\|\mathbf{A} - \widetilde{\mathbf{A}}\|_F^2$ subject to $\mathbf{A}^\top\mathbf{1} = \mathbf{0}$. According to KKT conditions (Boyd & Vandenberghe, 2004), we have

	#Train	#Test	Dim.	#Class
MIMIC	40K	18K	7207	2833
Cars	8144	8041	4096	196
Birds	9000	2788	4096	200
CNN	380K	3198	–	–
DailyMail	879K	53K	–	–

Table 1. Dataset Statistics

$\mathbf{A} - \widetilde{\mathbf{A}} + \mathbf{1}\boldsymbol{\lambda}^\top = 0$ and $\mathbf{A}^\top \mathbf{1} = \mathbf{0}$. Solving this system of equations, we get $\mathbf{A} = (\mathbf{I} - \frac{1}{d}\mathbf{1}\mathbf{1}^\top)\widetilde{\mathbf{A}}$, which centers the row vectors in $\widetilde{\mathbf{A}}$ to have zero mean.

4. Experiments

In this section, we present experimental results.

Dataset We used five datasets in the experiments: an electronic health record dataset MIMIC-III (Johnson et al., 2016); two image datasets Stanford-Cars (Krause et al., 2013) and Caltech-UCSD-Birds (Welinder et al., 2010); two question answering (QA) datasets CNN and DailyMail (Hermann et al., 2015). The first three were used for DML and the last two for LSTM. Their statistics are summarized in Table 1. MIMIC-III contains hospital admissions of patients. The class label of each admission is the primarily diagnosed disease. For Stanford-Cars, CNN and DailyMail, we use a single train/test split specified by the data providers; for the other two, five random splits are performed and the results are averaged over the five runs. For the MIMIC-III dataset, we extract 7207-dimensional features: (1) 2 dimensions from demographics, including age and gender; (2) 5300 dimensions from clinical notes, including 5000-dimensional bag-of-words (weighted using tf-idf) and 300-dimensional Word2Vec (Mikolov et al., 2013); (3) 1905-dimensions from lab tests where the zero-order, first-order and second-order temporal features are extracted for each of the 635 lab items. For bag-of-words, we remove stop words, then select the 5000 words with largest document frequency. For Word2Vec, we train 300-dimensional embeddings for each word; to represent a document, we average the embeddings of all words in this document. For the two image datasets, we use the VGG16 (Simonyan & Zisserman, 2014) convolutional neural network trained on the ImageNet (Deng et al., 2009) dataset to extract features, which are the 4096-dimensional outputs of the second fully-connected layer. In the two QA datasets, each instance consists of a passage, a question and an answer. The question is a cloze-style task where an entity is replaced by a placeholder and the goal is to infer this missing entity (answer) from all the possible entities appearing in the passage.

Experimental Setup In DML experiments, two samples are labeled as similar if belonging to the same class and dissimilar otherwise. The learned distance metrics are applied for retrieval whose performance is evaluated using precision@K. We compare with two sets of regularizers: (1) diversity-promoting regularizers based on determinant of covariance (DC) (Malkin & Bilmes, 2008), cosine similarity (CS) (Yu et al., 2011), determinantal point process (DPP) (Kulesza & Taskar, 2012; Zou & Adams, 2012), InCoherence (IC) (Bao et al., 2013), mutual angles (MA) (Xie et al., 2015), and decorrelation (DeCov) (Cogswell et al., 2015); (2) regularizers that are designed for other purposes, including L2 norm for small norm, L1 norm for sparsity, low-rankness (Recht et al., 2010) and Dropout (Srivastava et al., 2014). All these regularizers are applied to the same DML formulation (Eq.(4) without the regularizer). In addition, we compare with vanilla Euclidean distance (EUC) and other distance learning methods including information theoretic metric learning (ITML) (Davis et al., 2007), logistic discriminant metric learning (LDML) (Guillaumin et al., 2009), and geometric mean metric learning (GMML) (Zadeh et al., 2016). We use 5-fold cross validation to tune the regularization parameter in $\{10^{-5}, 10^{-4}, \cdots, 10^{5}\}$ and the number of components in $\{50, 100, 200, \cdots, 500\}$. The best tuned regularization parameters of UER are: 0.001 for MIMIC, 0.01 for Cars and Birds. The best tuned component numbers are: 200 for MIMIC, 100 for Cars and 200 for Birds. The learning rate of the PGD algorithm is set to 0.001.

	MIMIC	Cars	Birds
DML	72.5 ± 0.3	53.1 ± 0.0	55.9 ± 0.5
EUC	58.3 ± 0.1	37.8 ± 0.0	43.2 ± 0.0
ITML	69.3 ± 0.4	50.1 ± 0.0	52.9 ± 0.3
LDML	70.9 ± 0.9	51.3 ± 0.0	52.1 ± 0.2
GMML	71.2 ± 0.3	54.2 ± 0.0	53.7 ± 0.6
DML-L2	72.9 ± 0.1	53.4 ± 0.0	57.1 ± 0.4
DML-L1	72.6 ± 0.6	53.7 ± 0.0	56.4 ± 0.2
DML-LowRank	72.5 ± 0.7	53.3 ± 0.0	56.1 ± 0.6
DML-Dropout	73.1 ± 0.3	53.5 ± 0.0	56.6 ± 0.3
DML-DC	73.7 ± 0.4	57.1 ± 0.0	56.5 ± 0.4
DML-CS	73.5 ± 0.5	55.7 ± 0.0	57.4 ± 0.2
DML-DPP	74.2 ± 0.3	55.9 ± 0.0	56.9 ± 0.7
DML-IC	74.3 ± 0.2	56.3 ± 0.0	57.8 ± 0.2
DML-MA	73.6 ± 0.4	55.8 ± 0.0	58.2 ± 0.1
DML-DeCov	72.6 ± 0.1	56.2 ± 0.0	56.2 ± 0.8
DML-UE	$\mathbf{75.4} \pm 0.3$	$\mathbf{58.2} \pm 0.0$	$\mathbf{59.4} \pm 0.2$

Table 2. Precision@10 (%) on three datasets. The Cars dataset has a single train/test split, hence the standard error is 0.

In LSTM experiments, the model architecture and experimental settings follow the Bidirectional Attention Flow (BIDAF) (Seo et al., 2017) model, which consists of the following layers: character embedding, word embedding, contextual embedding, attention flow, modeling and output. The contextual and modeling layers use long short-term memory (LSTM) networks (Seo et al., 2017). In char-

	MIMIC	Cars	Birds	Average
DML	300	300	500	367
DML-L2	300	300	500	367
DML-L1	300	300	500	367
DML-LowRank	400	300	400	367
DML-Dropout	300	300	400	333
DML-DC	200	400	400	333
DML-CS	300	100	300	233
DML-DPP	200	300	300	267
DML-IC	400	300	200	300
DML-MA	300	200	300	267
DML-DeCov	300	400	300	333
DML-UE	200	100	200	**167**

Table 3. Optimal number of components.

acter embedding based on convolutional neural network, 100 1D filters are used, each with a width of 5. The hidden state size is set to 100. AdaDelta (Zeiler, 2012) is used for optimization with a minibatch size of 48. Dropout (Srivastava et al., 2014) with probability 0.2 is used for all LSTM layers. The model is trained for 8 epochs with early stop when the validation accuracy starts to drop. We compare UER with other diversity-promoting regularizers including DC, CS, DPP, IC, MA and DeCov.

Results Table 2 shows the retrieval precision ($K = 10$) on three datasets, where we observe: (1) DML-UE achieves much better precision than DML, proving that UER is an effective regularizer in improving generalization performance; (2) UER outperforms other diversity-promoting regularizers possibly due to its capability to capture global relations among all components and insensitivity to vector scaling; (3) diversity-promoting regularizers perform better than other types of regularizers such as L2, L1, low rank and Dropout, corroborating the efficacy of inducing diversity; (4) DML-UE outperforms other popular distance learning methods such as ITML, LDML and GMML.

Table 3 shows the number of components that achieves the precision in Table 2. Compared with DML, DML-UE uses much fewer components to achieve better precision. For example, on the Cars dataset, DML-UE achieves 58.2% precision with 100 components. In contrast, with more components (300), DML achieves a much lower precision (53.1%). This demonstrates that by encouraging the components to be diverse, UER is able to reduce model size without sacrificing modeling power. UER encourages equal "importance" among components such that each component plays a significant role in modeling data. As a result, it suffices to use a small number of components to achieve larger modeling power. Compared with other diversity-promoting regularizers, UER achieves better precision with fewer components, demonstrating its ability to better promote diversity.

	Frequent	Infrequent
DML	77.6 ± 0.2	64.2 ± 0.3
EUC	58.7 ± 0.1	57.6 ± 0.2
ITML	74.2 ± 0.6	61.3 ± 0.3
LDML	76.1 ± 0.8	62.3 ± 0.9
GMML	75.9 ± 0.1	63.5 ± 0.4
DML-L2	77.5 ± 0.3	65.4 ± 0.1
DML-L1	77.4 ± 0.5	64.8 ± 0.8
DML-LowRank	77.7 ± 0.5	64.0 ± 0.8
DML-Dropout	78.1 ± 0.2	64.9 ± 0.4
DML-DC	77.9 ± 0.4	66.8 ± 0.2
DML-CS	78.0 ± 0.5	66.2 ± 0.7
DML-DPP	77.3 ± 0.2	69.1 ± 0.5
DML-IC	78.5 ± 0.3	67.4 ± 0.2
DML-MA	76.8 ± 0.2	68.4 ± 0.4
DML-DeCov	77.1 ± 0.1	65.3 ± 0.1
DML-UE	78.3 ± 0.3	70.7 ± 0.4

Table 4. Precision@10 (%) on frequent and infrequent diseases of the MIMIC-III dataset.

Next, we verify whether "diversifying" the components in DML can better capture infrequent patterns. In the MIMIC-III dataset, we consider diseases as patterns and consider a disease as "frequent" if more than 1000 hospital admissions are diagnosed with this disease and "infrequent" if otherwise. Table 4 shows the retrieval precision on frequent diseases and infrequent diseases. As can be seen, compared with the baselines, DML-UE achieves more improvement on infrequent diseases than on frequent diseases. This indicates that by encouraging the components to diversely spread out, UER is able to better capture infrequent patterns (diseases in this case) without compromising the performance on frequent patterns. On infrequent diseases, DML-UE outperforms other diversity-promoting methods, showing the advantage of UER over other diversity-promoting regularizers. To further verify this, we select 3 most frequent diseases (hypertension, AFib, CAD) and randomly select 5 infrequent ones (helicobacter pylori, acute cholecystitis, joint pain-shlder, dysarthria, pressure ulcer), and show the precision@10 on each individual disease in Table 5. As can be seen, on the five infrequent diseases, DML-UE achieves higher precision than baselines while on the three frequent diseases, DML-UE achieves comparable precision.

We empirically verify whether UER can promote uncorrelation and evenness. Given m component vectors, we compute the empirical correlation (cosine similarity) of every two vectors, then average these pairwise correlation scores to measure the overall correlation of m vectors. We perform the study by learning distance metrics that have 200 components, on the MIMIC-III dataset. The average correlation under unregularized DML and DML-UE is 0.73 and 0.57 respectively. This shows that UER can reduce corre-

	1 (3566)	2 (3498)	3 (2757)	4 (204)	5 (176)	6 (148)	7 (131)	8 (121)
DML	80.2 ± 0.5	79.4 ± 0.8	80.9 ± 0.7	5.6 ± 1.6	6.1 ± 0.8	4.9 ± 0.8	4.3 ± 1.7	5.2 ± 0.6
EUC	66.4 ± 1.2	69.6 ± 1.2	61.8 ± 0.4	7.2 ± 1.0	6.9 ± 0.6	3.1 ± 1.4	6.8 ± 1.2	2.4 ± 0.7
ITML	75.4 ± 1.0	76.3 ± 0.9	79.3 ± 0.9	5.3 ± 1.2	3.7 ± 1.3	6.0 ± 0.9	3.3 ± 0.9	5.0 ± 1.0
LDML	77.0 ± 0.8	75.7 ± 1.0	78.1 ± 0.6	3.2 ± 1.3	5.6 ± 1.8	3.8 ± 1.1	6.7 ± 1.4	5.0 ± 1.3
GMML	76.3 ± 0.4	78.7 ± 0.8	79.8 ± 0.8	3.9 ± 1.8	5.9 ± 1.5	**11.9** ± 1.4	6.1 ± 0.6	6.3 ± 1.5
DML-L2	81.0 ± 0.5	79.3 ± 0.8	77.6 ± 0.4	4.4 ± 1.1	5.6 ± 0.9	3.7 ± 0.9	4.9 ± 1.2	6.2 ± 1.1
DML-L1	78.2 ± 1.0	79.9 ± 1.1	80.8 ± 1.2	6.3 ± 1.9	4.8 ± 1.1	9.5 ± 1.0	7.7 ± 1.0	5.6 ± 1.7
DML-LowRank	79.6 ± 0.4	79.7 ± 0.5	75.4 ± 0.5	3.1 ± 1.0	9.2 ± 1.2	5.5 ± 1.4	4.8 ± 0.6	4.5 ± 1.5
DML-Dropout	**81.6** ± 0.5	78.7 ± 0.7	80.7 ± 0.5	3.2 ± 1.5	4.2 ± 1.9	6.1 ± 0.9	4.2 ± 0.8	6.2 ± 1.9
DML-DC	77.9 ± 1.0	77.3 ± 0.9	80.3 ± 1.2	7.1 ± 0.8	8.9 ± 0.9	9.7 ± 1.4	11.9 ± 0.7	9.0 ± 1.6
DML-CS	80.0 ± 0.5	80.3 ± 0.7	80.8 ± 0.6	9.4 ± 1.3	4.8 ± 1.7	8.9 ± 1.9	9.7 ± 0.7	9.0 ± 1.0
DML-DPP	79.8 ± 0.8	77.6 ± 0.2	77.4 ± 0.7	10.1 ± 1.1	10.3 ± 0.8	8.8 ± 1.7	11.7 ± 1.2	8.4 ± 1.3
DML-IC	78.8 ± 1.3	79.2 ± 1.1	77.0 ± 0.8	11.8 ± 0.6	9.2 ± 1.4	5.7 ± 1.6	8.7 ± 1.4	9.6 ± 0.7
DML-MA	77.3 ± 1.1	80.1 ± 1.0	**81.0** ± 0.7	11.5 ± 1.1	9.9 ± 1.1	4.9 ± 1.1	7.6 ± 1.2	10.4 ± 1.4
DML-DeCov	80.7 ± 0.5	78.8 ± 0.7	80.5 ± 1.1	10.5 ± 0.8	**11.4** ± 1.2	9.2 ± 0.7	9.8 ± 1.2	10.4 ± 1.2
DML-UE	81.4 ± 0.9	**82.4** ± 0.8	80.5 ± 0.4	**14.3** ± 0.9	11.2 ± 1.3	10.7 ± 1.8	**15.8** ± 1.4	**13.2** ± 0.7

Table 5. Precision@10 (%) on three frequent and five infrequent diseases. The number next to a disease ID is its frequency.

	MIMIC	Cars	Birds
DML	20.5	9.1	10.1
DML-DC	22.3	10.9	11.7
DML-CS	20.9	9.7	10.5
DML-DPP	22.6	10.6	11.2
DML-IC	21.1	9.7	10.5
DML-MA	21.3	9.4	10.6
DML-DeCov	21.7	10.1	10.8
DML-UE	22.8	10.5	11.5

Table 6. Average runtime (hours) of DML methods

lation. To measure evenness, we first measure the "importance" of components. For each component with parameter vector $\mathbf{a}$, we project the training examples $\{\mathbf{x}_i\}_{i=1}^N$ onto $\mathbf{a}$: $\{\mathbf{x}_i^\top \mathbf{a}\}_{i=1}^N$, then use the variance of $\{\mathbf{x}_i^\top \mathbf{a}\}_{i=1}^N$ to measure the importance of this component. After that, we map these importance scores into a probabilistic simplex using softmax. Finally, the evenness is measured by the KL divergence between the discrete distribution parameterized by these probabilities and a uniform distribution. A smaller KL divergence indicates larger evenness. On MIMIC-III with 200 components, the KL divergence under unregularized DML and DML-UE is 3.54 and 2.92 respectively. This suggests that our regularizer is able to encourage evenness.

Table 6 shows the runtime taken by DML methods to reach convergence. Compared with unregularized DML, DML-UE does not increase the training time substantially. The relative increase is 11.2% on MIMIC, 15.4% on Cars and 13.9% on Birds. The runtime of DML-UE is close to DML regularized by other diversity-promoting regularizers.

In the LSTM experiments, Table 7 shows state of the art accuracy on the two QA datasets. Compared with the original BIDAF (Seo et al., 2017), our method BIDAF-UE achieves better accuracy, further demonstrating UER's ability to improve generalization performance. Besides, UER outperforms other regularizers.

	CNN		DailyMail	
	Dev	Test	Dev	Test
Kadlec et al. (2016)	68.6	69.5	75.0	73.9
Kobayashi et al. (2016)	71.3	72.9	–	–
Sordoni et al. (2016)	72.6	73.3	–	–
Trischler et al. (2016)	73.4	74.0	–	–
Chen et al. (2016)	73.8	73.6	77.6	76.6
Dhingra et al. (2016)	73.0	73.8	76.7	75.7
Cui et al. (2016)	73.1	74.4	–	–
Shen et al. (2016)	72.9	74.7	77.6	76.6
BIDAF	76.31	76.94	80.33	79.63
BIDAF-DC	76.36	76.98	80.51	79.68
BIDAF-CS	76.43	77.10	80.37	79.71
BIDAF-DPP	76.32	77.04	80.45	79.77
BIDAF-IC	76.41	77.21	80.49	79.83
BIDAF-MA	76.49	77.09	80.42	79.74
BIDAF-DeCov	76.35	77.15	80.38	79.67
BIDAF-UE	76.58	77.27	80.63	79.86
Dhingra et al. (2016)	77.9	77.9	81.5	80.9
Dhingra et al. (2017)	79.2	78.6	–	–

Table 7. Accuracy (%) on the two QA datasets

5. Conclusions

We propose a new diversity-promoting regularizer from the perspectives of *uncorrelation* which prefers the components in LSMs to be uncorrelated and *evenness* which encourages the components to contribute equally to the modeling of data. Gaining insight from PCA, promoting uncorrelation and evenness both amount to encouraging the covariance matrix of components to have uniform eigenvalues, which leads to a uniform eigenvalue regularizer (UER). The UER is applied to DML and LSTM. Experimental studies reveal that UER greatly boosts the performance of LSMs, better captures infrequent patterns, reduces model size without compromising modeling power and outperforms other diversity-promoting regularizers.

Acknowledgements

We would like to thank the anonymous reviewers for the helpful suggestions and comments. P.X and E.X are supported by National Institutes of Health P30DA035778, R01GM114311, National Science Foundation IIS1617583, DARPA FA872105C0003 and Pennsylvania Department of Health BD4BH4100070287.

References

Bao, Yebo, Jiang, Hui, Dai, Lirong, and Liu, Cong. Incoherent training of deep neural networks to decorrelate bottleneck features for speech recognition. 2013.

Bengtsson, Ingemar and Zyczkowski, Karol. *Geometry of quantum states: an introduction to quantum entanglement*. Cambridge University Press, 2007.

Bishop, Christopher M. Latent variable models. In *Learning in graphical models*, pp. 371–403. Springer, 1998.

Blei, David M. Build, compute, critique, repeat: Data analysis with latent variable models. *Annual Review of Statistics and Its Application*, 2014.

Blei, David M, Ng, Andrew Y, and Jordan, Michael I. Latent dirichlet allocation. *the Journal of machine Learning research*, 2003.

Boyd, Stephen and Vandenberghe, Lieven. *Convex optimization*. Cambridge university press, 2004.

Chen, Danqi, Bolton, Jason, and Manning, Christopher D. A thorough examination of the cnn/daily mail reading comprehension task. *arXiv preprint arXiv:1606.02858*, 2016.

Cogswell, Michael, Ahmed, Faruk, Girshick, Ross, Zitnick, Larry, and Batra, Dhruv. Reducing overfitting in deep networks by decorrelating representations. *arXiv preprint arXiv:1511.06068*, 2015.

Cover, Thomas M and Thomas, Joy A. *Elements of information theory*. John Wiley & Sons, 2012.

Cui, Yiming, Chen, Zhipeng, Wei, Si, Wang, Shijin, Liu, Ting, and Hu, Guoping. Attention-over-attention neural networks for reading comprehension. *arXiv preprint arXiv:1607.04423*, 2016.

Davis, Jason V, Kulis, Brian, Jain, Prateek, Sra, Suvrit, and Dhillon, Inderjit S. Information-theoretic metric learning. In *Proceedings of the 24th international conference on Machine learning*. ACM, 2007.

Deng, Jia, Dong, Wei, Socher, Richard, Li, Li-Jia, Li, Kai, and Fei-Fei, Li. Imagenet: A large-scale hierarchical image database. In *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, pp. 248–255. IEEE, 2009.

Dhingra, Bhuwan, Liu, Hanxiao, Cohen, William W, and Salakhutdinov, Ruslan. Gated-attention readers for text comprehension. *arXiv preprint arXiv:1606.01549*, 2016.

Dhingra, Bhuwan, Yang, Zhilin, Cohen, William W, and Salakhutdinov, Ruslan. Linguistic knowledge as memory for recurrent neural networks. *arXiv preprint arXiv:1703.02620*, 2017.

Ge, Tiezheng, He, Kaiming, Ke, Qifa, and Sun, Jian. Optimized product quantization for approximate nearest neighbor search. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2946–2953, 2013.

Guillaumin, Matthieu, Verbeek, Jakob, and Schmid, Cordelia. Is that you? metric learning approaches for face identification. In *IEEE International Conference on Computer Vision*. IEEE, 2009.

Harman, Harry H. Modern factor analysis. 1960.

Hermann, Karl Moritz, Kocisky, Tomas, Grefenstette, Edward, Espeholt, Lasse, Kay, Will, Suleyman, Mustafa, and Blunsom, Phil. Teaching machines to read and comprehend. In *Advances in Neural Information Processing Systems*, pp. 1693–1701, 2015.

Hochreiter, Sepp and Schmidhuber, Jürgen. Long short-term memory. *Neural computation*, 9(8):1735–1780, 1997.

Johnson, Alistair EW, Pollard, Tom J, Shen, Lu, Lehman, Li-wei H, Feng, Mengling, Ghassemi, Mohammad, Moody, Benjamin, Szolovits, Peter, Celi, Leo Anthony, and Mark, Roger G. Mimic-iii, a freely accessible critical care database. *Scientific data*, 3, 2016.

Jolliffe, Ian. *Principal component analysis*. Wiley Online Library, 2002.

Kadlec, Rudolf, Schmid, Martin, Bajgar, Ondrej, and Kleindienst, Jan. Text understanding with the attention sum reader network. *ACL*, 2016.

Knott, Martin and Bartholomew, David J. *Latent variable models and factor analysis*. Number 7. Edward Arnold, 1999.

Kobayashi, Sosuke, Tian, Ran, Okazaki, Naoaki, and Inui, Kentaro. Dynamic entity representation with max-pooling improves machine reading. In *Proceedings of NAACL-HLT*, pp. 850–855, 2016.

Kong, Weihao and Li, Wu-Jun. Isotropic hashing. In *Advances in Neural Information Processing Systems*, pp. 1646–1654, 2012.

Krause, Jonathan, Stark, Michael, Deng, Jia, and Fei-Fei, Li. 3d object representations for fine-grained categorization. In *Proceedings of the IEEE International Conference on Computer Vision Workshops*, pp. 554–561, 2013.

Kulesza, Alex and Taskar, Ben. Determinantal point processes for machine learning. *arXiv preprint arXiv:1207.6083*, 2012.

Kulis, Brian, Sustik, Mátyás A, and Dhillon, Inderjit S. Low-rank kernel learning with bregman matrix divergences. *Journal of Machine Learning Research*, 10 (Feb):341–376, 2009.

Magurran, Anne E. *Measuring biological diversity*. John Wiley & Sons, 2013.

Malkin, Jonathan and Bilmes, Jeff. Ratio semi-definite classifiers. In *2008 IEEE International Conference on Acoustics, Speech and Signal Processing*, pp. 4113–4116. IEEE, 2008.

Mariet, Zelda and Sra, Suvrit. Diversity networks. *arXiv preprint arXiv:1511.05077*, 2015.

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

Recht, Benjamin, Fazel, Maryam, and Parrilo, Pablo A. Guaranteed minimum-rank solutions of linear matrix equations via nuclear norm minimization. *SIAM review*, 52(3):471–501, 2010.

Seo, Minjoon, Kembhavi, Aniruddha, Farhadi, Ali, and Hajishirzi, Hannaneh. Bidirectional attention flow for machine comprehension. *ICLR*, 2017.

Shen, Yelong, Huang, Po-Sen, Gao, Jianfeng, and Chen, Weizhu. Reasonet: Learning to stop reading in machine comprehension. *arXiv preprint arXiv:1609.05284*, 2016.

Simonyan, Karen and Zisserman, Andrew. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

Sordoni, Alessandro, Bachman, Philip, Trischler, Adam, and Bengio, Yoshua. Iterative alternating neural attention for machine reading. *arXiv preprint arXiv:1606.02245*, 2016.

Srivastava, Nitish, Hinton, Geoffrey E, Krizhevsky, Alex, Sutskever, Ilya, and Salakhutdinov, Ruslan. Dropout: a simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research*, 15(1): 1929–1958, 2014.

Trischler, Adam, Ye, Zheng, Yuan, Xingdi, and Suleman, Kaheer. Natural language comprehension with the epireader. *arXiv preprint arXiv:1606.02270*, 2016.

Wang, Yi, Zhao, Xuemin, Sun, Zhenlong, Yan, Hao, Wang, Lifeng, Jin, Zhihui, Wang, Liubin, Gao, Yang, Law, Ching, and Zeng, Jia. Peacock: Learning long-tail topic features for industrial applications. *ACM Transactions on Intelligent Systems and Technology*, 2014.

Weinberger, Kilian Q and Saul, Lawrence K. Distance metric learning for large margin nearest neighbor classification. *Journal of Machine Learning Research*, 10(Feb): 207–244, 2009.

Welinder, Peter, Branson, Steve, Mita, Takeshi, Wah, Catherine, Schroff, Florian, Belongie, Serge, and Perona, Pietro. Caltech-ucsd birds 200. 2010.

Xie, Bo, Liang, Yingyu, and Song, Le. Diversity leads to generalization in neural networks. *AISTATS*, 2017.

Xie, Pengtao. Learning compact and effective distance metrics with diversity regularization. In *European Conference on Machine Learning*, 2015.

Xie, Pengtao, Deng, Yuntian, and Xing, Eric P. Diversifying restricted boltzmann machine for document modeling. In *ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, 2015.

Xie, Pengtao, Zhu, Jun, and Xing, Eric. Diversity-promoting bayesian learning of latent variable models. In *Proceedings of The 33rd International Conference on Machine Learning*, pp. 59–68, 2016.

Xing, Eric P, Jordan, Michael I, Russell, Stuart, and Ng, Andrew Y. Distance metric learning with application to clustering with side-information. In *Advances in neural information processing systems*, 2002.

Yu, Yang, Li, Yu-Feng, and Zhou, Zhi-Hua. Diversity regularized machine. 2011.

Zadeh, Pourya Habib, Hosseini, Reshad, and Sra, Suvrit. Geometric mean metric learning. 2016.

Zeiler, Matthew D. Adadelta: an adaptive learning rate method. *arXiv preprint arXiv:1212.5701*, 2012.

Zou, James Y and Adams, Ryan P. Priors for diversity in generative latent variable models. In *Advances in Neural Information Processing Systems*, pp. 2996–3004, 2012.