NAIST-IS-DD1461004

Doctoral Dissertation

Improving Nearest Neighbor Methods from the Perspective of Hubness Phenomenon

Yutaro Shigeto

March 15, 2017

Department of Information Processing
Graduate School of Information Science
Nara Institute of Science and Technology

A Doctoral Dissertation
submitted to Graduate School of Information Science,
Nara Institute of Science and Technology
in partial fulfillment of the requirements for the degree of
Doctor of ENGINEERING

Yutaro Shigeto

Thesis Committee:

Professor Yuji Matsumoto	(Supervisor)
Professor Kazushi Ikeda	(Co-supervisor)
Associate Professor Masashi Shimbo	(Co-supervisor)
Assistant Professor Hiroyuki Shindo	(Co-supervisor)
Assistant Professor Hiroshi Noji	(Co-supervisor)

Improving Nearest Neighbor Methods from the Perspective of Hubness Phenomenon*

Yutaro Shigeto

Abstract

Recently, *hubness phenomenon* has attracted attention in machine learning. It states that a small number of objects, called *hubs*, in the dataset, may occur as the nearest neighbor of many objects. The presence of these hubs will diminish the utility of nearest-neighbor methods, because the lists of nearest neighbors frequently contain the same hub objects regardless of the query. Although researchers have studied hubness in the ordinary setting in which the query and objects are represented in a feature space, it is not certain whether hubs are harmful to the multi-domain setting in which the query and objects are represented in different vector spaces.

In this thesis, we tackle the hubness phenomenon in multi-domain setting. Concretely, we first investigate the influence of hubs in bilingual lexicon extraction, which is a typical task of multi-domain matching. Our experiments show that the emergence of hub words emerge in this task, and it deteriorates the performance of bilingual lexicon extraction. We then discuss why hubs emerge in such task. To understand this, we introduce the degree of bias in the dataset, which causes hub formation. Based on this analysis, we propose a method that alleviates the influence of hubs. We also empirically show that the proposed approach outperforms the baseline methods. Moreover the presented analysis can apply to ordinary k-nearest neighbor classification problem, and thus we can extend the proposed method to k-nearest neighbor classification. In our experiments, we show that the proposed method surely reduces the emergence of hubs, and thus improving the classification accuracies accordingly, In addition, its training time is significantly faster than the existing distance metric learning methods.

*Doctoral Dissertation, Department of Information Processing, Graduate School of Information Science, Nara Institute of Science and Technology, NAIST-IS-DD1461004, March 15, 2017.

Keywords:

Nearest Neighbor Method, Hubness Phenomenon, Zero-Shot Learning, Distance Metric Learning

Acknowledgments

まずはじめに, 博士前期・後期課程あわせて 5 年間ご指導して頂いた松本裕治教授, 新保仁准教授に深く感謝致します. 研究の方法, 進め方など, 研究者としての生き方をご教授頂きました. また, 研究に関することだけではなく, 人生の歩き方についても相談に乗って頂き, 多くの助言をくださりました.

池田和司教授には, お忙しいなか副指導教員を引き受けて頂きました. 博士前期課程のゼミナール発表から, 博士後期課程最終審査まで, 研究に関する多くの助言を頂きました. ありがとうございました.

進藤裕之助教, 能地宏助教ならびに首都大学東京の小町守准教授, Johns Hopkins University の Kevin Duh さんにも, 研究に関する有益なコメントを頂きました. ありがとうございました.

秘書の北川祐子さんには, 事務手続きをはじめ大学での生活を助けて頂きました. ありがとうございました.

研究室の皆様にも, 研究に関する助言を頂きました. また, 日々の雑談など, 楽しい時間を共有させて頂きました. ありがとうございました.

研究室外の方にも大変お世話になりました. ハブ研究に関して様々な助言をしてくださった山形大学の鈴木郁美助教, 国立遺伝学研究所の原一夫さん. インターンシップを実施してくださった千葉工業大学人工知能・ソフトウェアセンター技術研究センターの皆様. ありがとうございました.

I am sincerely grateful to Professor Marco Saerens for accepting me as a visiting researcher in Université Catholique de Louvain.

最後に, これまで私を支えてくれた親族一同に心より感謝致します.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Background

Nearest neighbor methods are a fundamental technique in machine learning and data mining [Dasarathy, 1991; Duda et al., 2000; Wu et al., 2008]. Given a query and a set of objects, nearest neighbor search is to find a object in the set, which is closest to the given query; nearest neighbor classification is to predict the class label of query by its nearest neighbors.

Recently, Radovanović et al. [2010a] suggested that hubness phenomenon diminishes the utility of nearest neighbor methods. This phenomenon is concerned with nearest neighbor methods in high-dimensional space, and states that a small number of objects in the dataset, or *hubs*, may occur as the nearest neighbor of many objects. The emergence of these hubs will diminish the utility of nearest neighbor methods, because the list of nearest neighbors often contain the same hub objects regardless of the query object for which the list is computed.

Guided by this motivation, a surge of recent research [Radovanović et al., 2010a; Schnitzer et al., 2012; Suzuki et al., 2013] has tackled hubness phenomenon. Radovanović et al. was the first to present the mechanism of hubness phenomenon, which explains why hubs emerge. More recently, some researchers proposed the methods that reduce the emergence of hubs, and empirically showed that their methods improved the performance of nearest neighbor method [Schnitzer et al., 2012; Suzuki et al., 2013].

Although there have been increasing research activities in the ordinary setting in which the query and objects are represented in a vector space, it is not certain whether hubs are harmful to the multi-domain setting in which the query and objects are represented in different vector spaces.

1.2 Research Objective and Contributions

To investigate the behavior of hubs on multi-domain data, we first tackle bilingual lexicon extraction: given a word in a source language and a set of words in a target language, the goal of this task is to find the translation word in the target language for the given source word. This task is a typical multi-domain matching problem. In our experiments (Section 3.5), we observe the emergence of hubs in bilingual lexicon extraction: i.e., there exist specific words in the target language that are often chosen as the translation of many source words. This observation clearly shows that the emergence of hub words deteriorates the performance of bilingual lexicon extraction. To mitigate the effect of hubs, we extend the existing hubness reduction methods to bilingual lexicon extraction. These methods indeed reduce hubs, and hence obtain better results compared with baseline methods.

Next, we analyze why hubs emerge in the multi-domain setting. As mentioned earlier, the mechanism of hubs in the ordinary setting, i.e., data being single domain, was presented; on the other hand, the mechanism in the multi-domain setting is still unclear. In this analysis, we present a degree of bias in the data, which causes hub formation, as a function of the dimension of the space and the variance of object distribution, when the feature values of query and object follow zero-mean Gaussian distributions with different variances.

Based on our analysis, we develop a method, which can reduce hubs, and then evaluate it on the task of zero-shot learning. The zero-shot task is a type of classification problem, and its goal is to predict the unseen label of test object, from a training dataset which does not include objects related to unseen label. To predict unseen labels, zero-shot learning assumes that labels are embedded in a vector space. In other words, objects and labels are embedded in different vector spaces, and hence this task can be casted as the multi-domain matching problem. Indeed, many multi-domain matching tasks, including bilingual lexicon extraction, can be formulated as a task of zero-shot learning. As shown in Section 4.6, our proposed method outperforms the existing zero-shot learning method in an empirical evaluation using both synthetic and real data.

Although the above analysis assumes that the dataset is multi-domain, we can extend the analysis to the ordinary classification problem. To extend our analysis, we need to cast the problem as nearest neighbor search. Obviously, the procedure of k-nearest neighbor classification meets this requirement: Because test (unlabeled) objects are always queries, and labeled objects always plays the role of searched instances when the task is viewed as that of nearest neighbor search. Therefore, our analysis and

proposed method can be applied to nearest neighbor classification. In our experiments (Section 5.5), we demonstrate empirically that the proposed method achieves better k-NN classification accuracy than the metric learning methods on most document and image datasets, and comparable on the rest.

1.3 Structure of the Thesis

The thesis is organized as follows.

First, we give a brief review of two prior works on hubness phenomenon, which explain why hubs emerge in nearest neighbor search in Chapter 2.

Chapter 3 investigates the effect of hubness in bilingual lexicon extraction, which is a task of zero-shot learning. In this chapter, we extend the existing hubness reduction methods to bilingual lexicon extraction, and evaluate the methods empirically.

In Chapter 4, we present theoretical analysis for hubness phenomena in zero-shot learning, and propose a method which can reduce hubs for zero-shot learning. This method is also effective for k-nearest neighbor classification. We show this results in Chapter 5.

Finally, we conclude this thesis in Chapter 6.

Chapter 2

Hubness Phenomenon

2.1 Introduction

Recently, the *hubness phenomenon* [Radovanović et al., 2010a] is attracting attention as a new type of the "curse of dimensionality." This phenomenon is concerned with nearest neighbor methods in high-dimensional space, and states that a small number of objects in the dataset, or *hubs*, may occur as the nearest neighbor of many objects. The utility of nearest neighbor search would be significantly reduced if the same objects were to appear consistently as the search result, irrespective of the query. Radovanović et al. showed that such objects, termed *hubs*, indeed occur in high-dimensional space. Although this phenomenon may seem counter-intuitive, hubness is observed in a variety of real datasets and distance/similarity measures used in combination; [Radovanović et al., 2010a; Schnitzer et al., 2012; Suzuki et al., 2013].

Here, we briefly review prior research that explained the mechanism of hubness phenomenon. We introduce Radovanović et al.'s theorem, which was the first to explain why hubs emerge and which objects tend to be hubs, when Euclidean distance is used as the dissimilarity function of nearest neighbor method. We then show an analysis given by Suzuki et al.. It is similar to the theorem of Radovanović et al., but Suzuki et al. focused on the inner product (not Euclidean distance) as the similarity function.

2.2 Radovanović et al.'s Theorem

To understand why hubs emerge in high-dimensional space, Radovanović et al. [2010a] presented the following theorem. Let $\mathrm{E}[\cdot]$ and $\mathrm{Var}[\cdot]$ denote expectation and variance, respectively.

Theorem 1 ([Radovanović et al., 2010a, Theorem 1])**.** *Let $\mathbf{a}_d$ and $\mathbf{b}_d$ be two fixed objects in d-dimensional space, $\mathbf{X}_d$ be a d-dimensional random variable whose components independently follow the standard normal distribution.*

Define the norm of fixed objects which are specified by the random variable:

$$\|\mathbf{a}_d\| = \mathrm{E}[\|\mathbf{X}_d\|] + c_1\sqrt{\mathrm{Var}[\|\mathbf{X}_d\|]},$$
$$\|\mathbf{b}_d\| = \mathrm{E}[\|\mathbf{X}_d\|] + c_2\sqrt{\mathrm{Var}[\|\mathbf{X}_d\|]},$$

and $c_2 < c_1 \leq 0$; meaning $\|\mathbf{a}_d\| > \|\mathbf{b}_d\|$.

Consider the difference between the expected Euclidean distances

$$\Delta_d = \mathrm{E}[\|\mathbf{X}_d - \mathbf{a}_d\|] - \mathrm{E}[\|\mathbf{X}_d - \mathbf{b}_d\|].$$

We have the following inequalities which hold for $d > 2$:

$$\Delta_d = \sqrt{\frac{\pi}{2}} L_{1/2}^{d/2-1}\left(-\frac{\|\mathbf{a}_d\|^2}{2}\right) - \sqrt{\frac{\pi}{2}} L_{1/2}^{d/2-1}\left(-\frac{\|\mathbf{b}_d\|^2}{2}\right) > 0, \tag{2.1}$$

where L is the generalized Laguerre function, and further

$$\Delta_{d+2} > \Delta_d. \tag{2.2}$$

The Δ_d can be interpreted as the degree of the bias present in the data, which causes hub formation. Equation (2.1) shows that Δ_d is always positive.

This implies that an object $\mathbf{X}_d$, whose components independently follow the standard normal distribution, is more likely to be closer to object $\mathbf{b}_d$ than to $\mathbf{a}_d$; i.e., given query object $\mathbf{X}_d$, $\mathbf{b}_d$ is more likely to become its nearest neighbor.

Because this reasoning applies to any pair of objects $\mathbf{a}_d$ and $\mathbf{b}_d$ in the dataset, it can be concluded that the object, which is closer to the mean of the distribution (i.e., the origin in this case), is closer, on average, to all other objects for any value of d. In other

words, the object closest to the data mean tends to be hub. This bias is called *spatial centrality* [Radovanović et al., 2010a].

Moreover, Eq. (2.2) implies that Δ_d increases with increasing d. In other words, when the objects are in high-dimensional space, the degree of bias (spatial centrality) becomes large: Hubs tend to emerge in high-dimensional space.

From the above analysis, Radovanović et al. concluded that the object that is closest to the data mean tends to be hub in high-dimensional space.

2.3 Analysis Presented by Suzuki et al.

In the same vein as Radovanović et al. [2010a], when inner product is used as a similarity measure, Suzuki et al. [2013] argued that the objects which are similar to (i.e., have a high inner product value with) the data centroid tend to be hubs.

Given a dataset $\mathcal{D}$, its centroid $\bar{\mathbf{x}}$ is

$$\bar{\mathbf{x}} = \frac{1}{|\mathcal{D}|} \sum_{\mathbf{x} \in \mathcal{D}} \mathbf{x}.$$

Further, Suzuki et al. considered two fixed objects $\mathbf{a}$ and $\mathbf{b}$, such that

$$\langle \bar{\mathbf{x}}, \mathbf{a} \rangle - \langle \bar{\mathbf{x}}, \mathbf{b} \rangle < 0. \tag{2.3}$$

That is to say, $\mathbf{b}$ is more similar to the centroid $\bar{\mathbf{x}}$ than $\mathbf{a}$.

In this situation, they are interested in *which object,* $\mathbf{a}$ *or* $\mathbf{b}$*, is more likely to be a hub".*

To answer this question, Suzuki et al. considered the difference of the average of inner products, Δ:

$$\begin{aligned}
\Delta &= \frac{1}{|\mathcal{D}|} \sum_{\mathbf{x} \in \mathcal{D}} \langle \mathbf{x}, \mathbf{a} \rangle - \frac{1}{|\mathcal{D}|} \sum_{\mathbf{x} \in \mathcal{D}} \langle \mathbf{x}, \mathbf{b} \rangle \\
&= \left\langle \frac{1}{|\mathcal{D}|} \sum_{\mathbf{x} \in \mathcal{D}} \mathbf{x}, \mathbf{a} \right\rangle - \left\langle \frac{1}{|\mathcal{D}|} \sum_{\mathbf{x} \in \mathcal{D}} \mathbf{x}, \mathbf{b} \right\rangle \\
&= \langle \bar{\mathbf{x}}, \mathbf{a} \rangle - \langle \bar{\mathbf{x}}, \mathbf{b} \rangle
\end{aligned}$$

By combining the last equation with Eq. (2.3), we obtain the following inequality:

$$\Delta = \frac{1}{|\mathcal{D}|} \sum_{\mathbf{x} \in \mathcal{D}} \langle \mathbf{x}, \mathbf{a} \rangle - \frac{1}{|\mathcal{D}|} \sum_{\mathbf{x} \in \mathcal{D}} \langle \mathbf{x}, \mathbf{b} \rangle < 0. \tag{2.4}$$

Here, recall the first question: that is which object, **a** or **b**, is more likely to be a hub. Equation (2.4) states that, on average, **a** is more similar to objects in the dataset than **b**. In other words, the object, whose similarity to the centroid is higher, i.e., **b** in this case, tends to be hub.

Because this tendency holds for any two objects which satisfy the condition (Eq. (2.3)), Suzuki et al. concluded that there exists a bias, spatial centrality, in the dataset: the object, which is most similar to the centroid, tends to be hub.

Chapter 3

Reducing Hub Words Improves the Accuracy of Bilingual Lexicon Extraction

3.1 Introduction

3.1.1 Background

There exist well-established techniques to extract bilingual translation pairs from parallel corpora in statistical machine translation. The problem is that parallel corpora are costly to build, and only available for limited language pairs and domains. Many researchers have hence pursued bilingual lexicon extraction without using parallel corpora [Fung and Yee, 1998; Irvine and Callison-Burch, 2013; Koehn and Knight, 2002; Rapp, 1999; Tamura et al., 2012; Vulić and Moens, 2013a]. Their focus is on how to utilize monolingual (comparable) corpora in source and target languages for this task.

A dominant approach in this area is to make a common vector space for words across two languages, and measure their similarity therein. If a sufficient number of translation word pairs (*seed* lexicon) are available, such a vector space can be built by taking the seed pairs for the bases of the space. To map a word—in either the source or target language—onto this vector space, its similarity to each seed word (of the same language as the word of interest) is calculated over a corpus; the corpus need not be a parallel corpus since the similarity is computed within the language of the word to map. These similarity scores are then used as the components of the feature vector for the word, thus resulting in a vector of dimension equal to the number of

seed translations. Once words are mapped onto such a common vector space, the task of bilingual lexicon extraction reduces to nearest-neighbor search; most similar cross-language word pairs in this space (measured for example by cosine) are extracted as likely translations.

In machine learning community, meanwhile, the hubness phenomenon [Radovanović et al., 2010a] is attracting attention as a new form of the "curse of dimensionality." This phenomenon is concerned with nearest neighbor methods in a high dimensional space, and states that a small number of objects in the data, or *hubs*, may occur in the k-nearest neighbors of many objects; emergence of hubs will render nearest neighbor search less useful, since nearest neighbor lists often contain the same hub objects regardless of the query object for which the list is computed.

3.1.2 Research Objective and Contributions

This chapter investigates the effect of hubs on the common vector space methods for bilingual lexicon extraction. The vector space has a dimension equal to the number of seed translations, which typically is some hundreds, or thousands. While the number is relatively small compared to the size of entire vocabulary, the resulting space is still high-dimensional, and might be prone to hubness. Indeed, as we demonstrate later, there appear a small number of hub translation candidates that are frequently deemed as a possible translation of many source words. The objective of this research is to investigate ways to suppress hubs, and to improve the accuracy of bilingual lexicon extraction.

Our contributions in this chapter are as follows.

- We point out that the hubness phenomenon severely deteriorates the performance of vector space approaches to bilingual lexicon extraction. In bilingual lexicon extraction, hubs correspond to specific words in the target language that are chosen frequently as a translation of many source words.

- We demonstrate that reducing hubness improves the accuracy of bilingual lexicon extraction. In this research, we extend the *centering* transformation and *mutual proximity* to bilingual lexicon extraction. As shown in Section 3.5, both methods outperform an existing method [Tamura et al., 2012]. Centering and mutual proximity have recently been shown as an effective method for hub re-

duction [Schnitzer et al., 2012; Suzuki et al., 2013], but for limited tasks. This work is the first to report its effectiveness in bilingual lexicon extraction.

3.2 Related Work

Bilingual lexicon extraction has been an active research topic in cross-lingual natural language processing. The goal of this task is to find words that are translations of each other. To find translations, previous methods assumed that words that are translation of each other have similar properties (e.g., context, frequency, spelling, and topic) across languages. Similarity cannot be computed directly, since source language words and target language words represent each language vector space.

To represent words in a common vector space, many researchers proposed various effective methods.

These methods are broadly divided into two approaches: One approach uses the existing bilingual dictionary, and another does not use the bilingual dictionary.

In the former, most of previous methods use context distribution that is expressed by co-occurring words around the target word in each language. Since feature vectors in different languages are in each language vector space, seed bilingual lexicon are used by projecting context in the common feature space. This is usually done with the help of seed lexicon, and this work also follows this approach.

Fung [1995] and Rapp [1999] were the first to use seed lexicon as the axes of the common feature space over the source and target languages. Some researchers [Andrade et al., 2011; Bouamor et al., 2013; Hazem and Morin, 2013; Morin and Prochasson, 2011] are more recent work along this line.

Tamura et al. [2012] presented a common space method based on graph-based semi-supervised learning. In particular, they used a label propagation algorithm [Zhu et al., 2003] to associate to each word a vector of higher-order distributional similarity with seed words.

Some researchers compose a common vector space without using seed lexicon. The method proposed by Fung [1995] used context heterogeneity, and Yu and Tsujii [2009] used dependency heterogeneity to find probable translation pairs to use as the axes of the vector space. Koehn and Knight [2002] proposed a bootstrap approach which uses orthographically identical words as the initial seed lexicon, and then source and target words with the highest similarity is added to the seeds in the subsequent trials. Haghighi et al. [2008] proposed a generative model that projects source and tar-

get words into a common latent space of topics. Some researchers [Liu et al., 2013; Mimno et al., 2009; Vulić and Moens, 2013a; Vulić et al., 2011] are approaches that also make a common feature space using topic models.

Some recent methods [Aker et al., 2013; Irvine and Callison-Burch, 2013; Prochasson and Fung, 2011] did not make a common feature space, but treated bilingual lexicon extraction as a pairwise binary classification problem; these methods directly predicted whether a pair of source and target words is a translation or not.

More recently, some researchers [Dinu and Baroni, 2015; Mikolov et al., 2013b] proposed the regression based approaches to construct the feature space where similarity measure can be calculated. We will discuss these approaches in Section 4.

Although, their research objective, in most cases, were how to utilize the comparable corpus for extraction, our objectives are to investigate the effect of hubness on bilingual lexicon extraction, and how to mitigate the effect.

3.3 A Vector Space Approach to Bilingual Lexicon Extraction

In this section, we describe a common vector space for words across languages, built from seed lexicon. This vector space is used in our experiments (Section 3.5).

The vector representation of words, which we describe below, is basically the one called *similarity vectors* in [Koehn and Knight, 2002, Chapter 3], except that we do not convert the scores of distributional similarity into ranks; see also [Diab and Finch, 2000].

In this vector space, a word is represented by a vector holding the scores of monolingual distributional similarity between the word and the individual seed words (of the same language); the resulting vector is of dimension equal to the seed size, and each seed translation pair corresponds to an axis of the space.

Let $\{(s^{(i)}, t^{(i)}) \mid i = 1, \ldots, n\}$ be a seed lexicon, where $s^{(i)}$ is a word in the source language, and $t^{(i)}$ its translation in the target language.[1] For an arbitrary word s (not necessarily a seed word) in the source language, its feature vector $\mathbf{s} = [s_1, \ldots, s_n]^{\mathrm{T}} \in \mathbb{R}^n$ is such that the j-th component s_j is given by the distributional similarity between s and $s^{(j)}$, the source word in the j-th seed pair. Both words are in the same (source)

[1] In this chapter, we use parenthesized superscripts to denote sample (seed) indices, and subscripts to denote component indices within a feature vector.

language, so the distributional similarity can be readily computed over a corpus of the source language.

Specifically in this chapter, vector component s_j holds the cosine of the context vector of the word and that of the j-th seed word; i.e., if we let $v(s)$ be the context (co-occurrence) vector of the word s, whose components correspond to various context patterns (usually words in a vicinity) around words in a corpus, and let $v(s^{(j)})$ be the context vector of a seed word $s^{(j)}$, then

$$s_j = \frac{\left\langle v(s), v(s^{(j)}) \right\rangle}{\|v(s)\| \cdot \|v(s^{(j)})\|}. \tag{3.1}$$

Similar computation can be done for any word t in the target language. Its feature vector $\mathbf{t} \in \mathbb{R}^n$ holds as the j-th component the distributional similarity of words t and the target word $t^{(j)}$ of the j-th seed pair, this time computed over a target language corpus.

Once we obtain such a common vector representation of words, similarity of words across two languages can be computed, for instance, by cosine similarity in this vector space. Hence, given a source word s, or rather, its feature vector $\mathbf{s}$, the task of finding its most likely translations reduces to that of ranking candidate words in the target language by the similarity between $\mathbf{s}$ and their feature vectors in this space.

3.4 Reducing the Effect of Hubness

3.4.1 Hubness Phenomenon in High-Dimensional Space

The hubness phenomenon [Radovanović et al., 2010a] states that in high-dimensional space, a small number of objects in the data, or *hubs*, may occur in the nearest neighbor of many objects; this means that the same objects may appear frequently in the nearest neighbor list of many other objects, regardless of the object for which the list is computed. Consequently, hubs will significantly reduce the utility of nearest neighbor search.

The hubness phenomenon is relevant to bilingual lexicon extraction as well; as we saw in Section 3.3, its task is basically a nearest-neighbor search in a high dimensional space. The space is high-dimensional because the number of seed translation pairs can typically exceed some hundreds or thousands, and this number determines the dimension of the vector space in bilingual lexicon extraction.

Below we present two approaches that we use to reduce hubs in bilingual lexicon extraction: *centering*, and *mutual proximity*. Both methods as means to suppress hubs were proposed, but in other limited application domains.

3.4.2 Centering: Reducing the Bias in the Dataset

Centering

Centering [Eriksson et al., 2006; Fisher and Lenz, 1996; Mardia et al., 1979] moves the origin of the vector space to the centroid $\bar{\mathbf{x}}$ of the dataset. Given a dataset $\mathcal{D}$, its centroid $\bar{\mathbf{x}}$ is

$$\bar{\mathbf{x}} = \frac{1}{|\mathcal{D}|} \sum_{\mathbf{x} \in \mathcal{D}} \mathbf{x}$$

After centering, each object $\mathbf{x} \in \mathcal{D}$ is transformed to

$$\mathbf{x}_{\text{cent}} = \mathbf{x} - \bar{\mathbf{x}}$$

and all subsequent evaluation of object similarity takes place in this centered space.

Centering is a classic method for removing bias in the data. Recently, it was rediscovered as an effective way to mitigate the effect of hubness [Suzuki et al., 2013], when the similarity is measured by inner product (which also subsumes cosine similarity).

Centering moves the origin of the space to the data centroid, effectively reducing the inner product between objects similar to the centroid in the original space and other objects; i.e., these objects are less similar to other objects after centering. And according to Radovanović et al. [2010a]; Suzuki et al. [2013], these objects most similar to the centroid are the ones that tend to become hubs, as discussed in Chapter 2. It follows that objects that were hubs in the original space are now less similar to other objects in the data, which also reduces the chance for these objects to make hubs.

To understand the above analysis, we recall the discussion in Section 2.3. Further, we define $\mathbf{a}_{\text{cent}}$, $\mathbf{b}_{\text{cent}}$, and $\mathcal{D}_{\text{cent}}$ as $\mathbf{a} - \bar{\mathbf{x}}$, $\mathbf{b} - \bar{\mathbf{x}}$, and the centered dataset, respectively.

After centering, the degree of bias, i.e., spatial centrality, can be rewritten as:

$$\begin{aligned}\Delta_{\text{cent}} &= \frac{1}{|\mathcal{D}_{\text{cent}}|}\sum_{\mathbf{x}_{\text{cent}}\in\mathcal{D}_{\text{cent}}}\langle\mathbf{x}_{\text{cent}},\mathbf{a}_{\text{cent}}\rangle - \frac{1}{|\mathcal{D}_{\text{cent}}|}\sum_{\mathbf{x}_{\text{cent}}\in\mathcal{D}_{\text{cent}}}\langle\mathbf{x}_{\text{cent}},\mathbf{b}_{\text{cent}}\rangle \\ &= \left\langle\frac{1}{|\mathcal{D}_{\text{cent}}|}\sum_{\mathbf{x}_{\text{cent}}\in\mathcal{D}_{\text{cent}}}\mathbf{x}_{\text{cent}},\mathbf{a}_{\text{cent}}\right\rangle - \left\langle\frac{1}{|\mathcal{D}_{\text{cent}}|}\sum_{\mathbf{x}_{\text{cent}}\in\mathcal{D}_{\text{cent}}}\mathbf{x}_{\text{cent}},\mathbf{b}_{\text{cent}}\right\rangle \\ &= \langle\bar{\mathbf{x}}_{\text{cent}},\mathbf{a}_{\text{cent}}\rangle - \langle\bar{\mathbf{x}}_{\text{cent}},\mathbf{b}_{\text{cent}}\rangle\,.\end{aligned}$$

Since the centroid of centered dataset is at the origin of space: i.e., $\bar{\mathbf{x}}_{\text{cent}} = \mathbf{0}$, we have

$$\begin{aligned}\Delta_{\text{cent}} &= \langle\bar{\mathbf{x}}_{\text{cent}},\mathbf{a}_{\text{cent}}\rangle - \langle\bar{\mathbf{x}}_{\text{cent}},\mathbf{b}_{\text{cent}}\rangle \\ &= \langle\mathbf{0},\mathbf{a}_{\text{cent}}\rangle - \langle\mathbf{0},\mathbf{b}_{\text{cent}}\rangle \\ &= 0.\end{aligned}$$

This equality implies that the average of inner product of $\mathbf{a}_{\text{cent}}$ is equivalent to that of $\mathbf{b}_{\text{cent}}$. Because this holds for any pair of objects $\mathbf{a}$ and $\mathbf{b}$ in this space, the object, which is most similar to the centroid, does not exist. In other words, there is no bias in the dataset. Suzuki et al. concluded that the bias is vanished after centering, thus reducing hubs.

Centering for bilingual lexicon extraction

We expect that centering can reduce the emergence of hub words in bilingual lexicon extraction, and thus the performance is improved. In this section, we explain how to apply centering to bilingual lexicon extraction.

We first define two sets $\mathcal{S}$ and $\mathcal{T}$: $\mathcal{S}$ is the set of source words (queries), and $\mathcal{T}$ is the set of target words which are the targets (not queries) of nearest neighbor search. Given a source word $\mathbf{s}\in\mathcal{S}$, a target word $\mathbf{t}\in\mathcal{T}$, and certain centroid $\bar{\mathbf{x}}$, centering transforms them to new points:

$$\mathbf{s}_{\text{cent}} = \mathbf{s} - \bar{\mathbf{x}} \tag{3.2}$$

$$\mathbf{t}_{\text{cent}} = \mathbf{t} - \bar{\mathbf{x}}. \tag{3.3}$$

In fact, the procedure of centering for target word, Eq. (3.3) is not required, because of the following reason. In bilingual lexicon extraction, system finds the most similar target object for a given source object.

For example, given two centered target words $\mathbf{t}_{\text{cent}}^{(1)}$ and $\mathbf{t}_{\text{cent}}^{(1)}$, if we want to decide which target word is more similar to a source word $\mathbf{s}_{\text{cent}}$. To decide this, we first compute inner product between them: $\langle \mathbf{s}_{\text{cent}}, \mathbf{t}_{\text{cent}}^{(1)} \rangle$ and $\langle \mathbf{s}_{\text{cent}}, \mathbf{t}_{\text{cent}}^{(2)} \rangle$. We then compare these inner products:

$$\begin{aligned}
\langle \mathbf{s}_{\text{cent}}, \mathbf{t}_{\text{cent}}^{(1)} \rangle - \langle \mathbf{s}_{\text{cent}}, \mathbf{t}_{\text{cent}}^{(2)} \rangle &= \langle \mathbf{s}_{\text{cent}}, \mathbf{t}_{\text{cent}}^{(1)} - \mathbf{t}_{\text{cent}}^{(2)} \rangle \\
&= \langle \mathbf{s}_{\text{cent}}, (\mathbf{t}^{(1)} - \bar{\mathbf{x}}) - (\mathbf{t}^{(2)} - \bar{\mathbf{x}}) \rangle \\
&= \langle \mathbf{s}_{\text{cent}}, \mathbf{t}^{(1)} - \mathbf{t}^{(2)} \rangle \\
&= \langle \mathbf{s}_{\text{cent}}, \mathbf{t}^{(1)} \rangle - \langle \mathbf{s}_{\text{cent}}, \mathbf{t}^{(2)} \rangle.
\end{aligned}$$

Thus, inner product needs to be measured only between the centered source word and target words without centering.

In the nearest neighbor search phase, we first conduct centering only for source words by Eq. (3.2). Then we carry out nearest neighbor search by regarding with target words without centering.

Centroids in bilingual lexicon extraction

In previous work on hubness, there was only a single source of data, $\mathcal{D}$. Thus, data mean $\bar{\mathbf{x}} = 1/|\mathcal{D}| \sum_{\mathbf{x} \in \mathcal{D}} \mathbf{x}$ was used as the centroid. However, in bilingual lexicon extraction, data mean can be computed in three ways:

- The average of source words:

$$\bar{\mathbf{s}} = \frac{1}{|\mathcal{S}|} \sum_{\mathbf{s} \in \mathcal{S}} \mathbf{s}. \tag{3.4}$$

- The average of target words:

$$\bar{\mathbf{t}} = \frac{1}{|\mathcal{T}|} \sum_{\mathbf{t} \in \mathcal{T}} \mathbf{t}. \tag{3.5}$$

- The average of source and target words:

$$\mathbf{c} = \frac{1}{|\mathcal{S}| + |\mathcal{T}|} \left(\sum_{\mathbf{s} \in \mathcal{S}} \mathbf{s} + \sum_{\mathbf{t} \in \mathcal{T}} \mathbf{t} \right). \tag{3.6}$$

In the context of nearest neighbor search, source word is always query, and target word is always the searched instance. As mentioned in Section 3.4.2, hubs in nearest neighbor search can be reduced by centering with query's centroid. Thus, centering, which moves the origin to $\bar{\mathbf{s}}$, can alleviate the emergence of hub target words. Indeed, we observed that centering with $\bar{\mathbf{s}}$ reduced the hubs in our experiments (Section 3.5).

3.4.3 Mutual Proximity: Breaking the Asymmetric Neighbor Relation

Mutual proximity

When hubs emerge in the dataset, hubs become the nearest neighbor of many objects.[2] In contrast, the nearest neighbor of such hub object is a single object in the dataset, and the others can not become the nearest neighbor. As a consequence, the nearest neighbor relations might be asymmetric: hub object x is the nearest neighbors of many objects but not vice versa.

Schnitzer et al. [2012] argued this observation causes the hubness phenomenon, and thus, proposed the scaling method, called mutual proximity, which attempts to symmetrize nearest neighbor relations for alleviating the emergence of hubs. They defined the mutual proximity between $\mathbf{x}$ and $\mathbf{y}$, $\mathrm{MP}(\mathbf{x},\mathbf{y})$, as:

> When an object $\mathbf{z}$ is sampled from a distribution, mutual proximity represents a joint probability, that is $\mathbf{y}$ is the closer than $\mathbf{z}$ from $\mathbf{x}$, and $\mathbf{x}$ is the closer than $\mathbf{z}$ from $\mathbf{y}$.

Hence, mutual proximity can be represented by the following equation:

$$\mathrm{MP}(\mathbf{x},\mathbf{y}) = P\left((X > \|\mathbf{x}-\mathbf{y}\|) \wedge (Y > \|\mathbf{x}-\mathbf{y}\|)\right), \tag{3.7}$$

where random variables X and Y depict the distances from $\mathbf{z}$ to $\mathbf{x}$ and $\mathbf{y}$, $\|\mathbf{x}-\mathbf{z}\|$ and $\|\mathbf{y}-\mathbf{z}\|$, respectively.[3]

[2] To simplify the discussion, we consider 1-nearest neighbor. However the same discussion applies to k-nearest neighbor with $k > 1$.

[3] We use the Euclidean distance as the distance measure. The original paper by Schnitzer et al. [2012] provided a more general framework with arbitrary distance.

To compute mutual proximity, Eq. (3.7) can be rewritten as:

$$\begin{aligned}\mathrm{MP}(\mathbf{x},\mathbf{y}) &= P\left((X > \|\mathbf{x}-\mathbf{y}\|) \wedge (Y > \|\mathbf{x}-\mathbf{y}\|)\right) \\ &= 1 - P((X \leq \|\mathbf{x}-\mathbf{y}\|) \vee (Y \leq \|\mathbf{x}-\mathbf{y}\|)) \\ &= 1 - [P(X \leq \|\mathbf{x}-\mathbf{y}\|) + P(Y \leq \|\mathbf{x}-\mathbf{y}\|) \\ &\qquad - P((X \leq \|\mathbf{x}-\mathbf{y}\|) \wedge (Y \leq \|\mathbf{x}-\mathbf{y}\|))]\,.\end{aligned}$$

If we can assume that the distances $\|\mathbf{x}-\mathbf{z}\|$ and $\|\mathbf{y}-\mathbf{z}\|$ follow a certain probability distribution, mutual proximity can be straightforwardly obtained from the cumulative distribution function:

$$\mathrm{MP}(\mathbf{x},\mathbf{y}) = 1 - [F_X(\|\mathbf{x}-\mathbf{y}\|) + F_Y(\|\mathbf{x}-\mathbf{y}\|) - F_{X,Y}(\|\mathbf{x}-\mathbf{y}\|, \|\mathbf{x}-\mathbf{y}\|)]\,,$$

where F_X and F_Y represent the cumulative distribution function of the random variables X and Y, and $F_{X,Y}$ is the joint cumulative distribution function.

However, probability distribution is unknown in most cases. Schnitzer et al. therefore proposed an empirical way of computing mutual proximity: given a set of objects $\mathcal{D}$ sampled from an unknown distribution, mutual proximity is estimated from the empirical distribution on $\mathcal{D}$. Thus empirical computation of mutual proximity is to count the number of objects $\mathbf{z}$ whose distances to $\mathbf{x}$ and $\mathbf{y}$ are greater than the distance between $\mathbf{x}$ and $\mathbf{y}$:

$$\mathrm{MP}(\mathbf{x},\mathbf{y}) = \frac{|\{\mathbf{z} \in \mathcal{D} \mid (\|\mathbf{x}-\mathbf{z}\| > \|\mathbf{x}-\mathbf{y}\|) \wedge (\|\mathbf{z}-\mathbf{y}\| > \|\mathbf{x}-\mathbf{y}\|)\}|}{|\mathcal{D}|}. \tag{3.8}$$

This equation implies that mutual proximity reduces the emergence of hubs. When either $\mathbf{x}$ or $\mathbf{y}$ is a hub, the object $\mathbf{z}$, which satisfies the stated condition in Eq. (3.8), may not appear frequently Since mutual proximity increases with the number of such objects $\mathbf{z}$, and vice versa, mutual proximity between hub and non hub objects tends to be smaller. Therefore, the object that is a hub in the original metric, i.e., Euclidean distance in this case, will not be nearest neighbor of many objects. In other words, mutual proximity reduces the emergence of hub.

The computational cost of counting $\mathbf{z}$ is expensive. To reduce computational cost, Schnitzer et al. also proposed an approximation method of mutual proximity [Schnitzer et al., 2012, Section 3.2.2].

Although, in general, two random variables X and Y are not independent, assuming independence simplifies Eq. (3.7) as follows:

$$\mathrm{MP}_I(\mathbf{x},\mathbf{y}) = P(X > \|\mathbf{x}-\mathbf{y}\|)\,P(Y > \|\mathbf{x}-\mathbf{y}\|).$$

In the case of empirical distribution, mutual proximity can be computed by:

$$\mathrm{MP}_I(\mathbf{x},\mathbf{y}) = \frac{|\{\mathbf{z}\in\mathcal{D} \mid \|\mathbf{x}-\mathbf{z}\| > \|\mathbf{x}-\mathbf{y}\|\}|}{|\mathcal{D}|} \times \frac{|\{\mathbf{z}\in\mathcal{D} \mid \|\mathbf{z}-\mathbf{y}\| > \|\mathbf{x}-\mathbf{y}\|\}|}{|\mathcal{D}|}. \quad (3.9)$$

Mutual proximity for bilingual lexicon extraction

We extend the mutual proximity in Eq. (3.9) to bilingual lexicon extraction.

We recall the definition of two sets $\mathcal{S}$ and $\mathcal{T}$: $\mathcal{S}$ is the set of source words (queries), and $\mathcal{T}$ is the set of target words which are the targets (not queries) of nearest neighbor search.

In this case, computation of mutual proximity is to substitute $\mathbf{x}$ and $\mathbf{y}$ in Eq. (3.9) for $\mathbf{s}\in\mathcal{S}$ and $\mathbf{t}\in\mathcal{T}$ respectively. Since (dis)similarity, in bilingual lexicon extraction, needs to be computed only between source and target words, $\mathbf{z}$ in the first term of Eq. (3.9) is the word of the set of target words, and the other is in the source words. Thus, mutual proximity for bilingual lexicon extraction can be computed by:

$$\mathrm{MP}_I(\mathbf{s},\mathbf{t}) = \frac{|\{\mathbf{z}\in\mathcal{T} \mid \|\mathbf{s}-\mathbf{z}\| > \|\mathbf{s}-\mathbf{t}\|\}|}{|\mathcal{T}|} \times \frac{|\{\mathbf{z}\in\mathcal{S} \mid \|\mathbf{z}-\mathbf{t}\| > \|\mathbf{s}-\mathbf{t}\|\}|}{|\mathcal{S}|}. \quad (3.10)$$

3.5 Experiments

The objective of this experiments is to investigate the effect of hubs on bilingual lexicon extraction, and whether the proposed methods actually reduce hubs, or not.

Given a source word, the goal of bilingual lexicon extraction is to rank its gold translation (the one listed in an existing bilingual lexicon as the translation of the source word) higher than other non-translation words (decoys). In this experiment, English is the source language, and Japanese is the target: The task is to find the Japanese words which are the translations of given English words.

3.5.1 Experimental Setups

Datasets

We prepared two sets of English-Japanese lexicons and comparable corpora for evaluation. Lexicons are used to extract gold translations, which are then used as the seed sets and test sets. As will be shown later, corpora are used only for computing monolingual distributional similarity.

Table 3.1: Corpus statistics. "#noun" indicates the number of words, which also exist on the dictionary, in the corpus. "#pair" depicts the number of bilexicon that we used in this experiments. "#unique" is the number of unique words in the bilexicon.

Corpus (language)	dictionary	#sentences	#nouns	#pairs	#unique words
MEDLINE (En)	LSD	139,404	2633	1213	1213
PNE (Ja)		512,504	2579		1212
Wikipedia (En)	EDR	334,886	6916	2102	2086
Wikipedia (Ja)		162,138	5474		2012

- MEDLINE-PNE: The English corpus in this dataset is a portion of the MEDLINE abstracts of the articles published in 2006.[4] As the Japanese corpus, we used the full text content of articles published from 1985 through 2006 in Japanese bio-science journal PNE.[5] The Life Science Dictionary is used as the bilingual lexicon.[6]

- Wikipedia: We used interlinked Wikipedia articles, 5,000 each from English and Japanese Wikipedia, as the corpora.[7] The EDR Japanese-to-English dictionary is used as the bilingual lexicon.[8]

Table 3.1 gives the summary of the data set.

We ran part-of-speech taggers on these corpora, and removed functional words. The GENIA tagger[9] and hunpos[10] were used for assigning part-of-speech tags to MEDLINE abstracts and English Wikipedia pages, respectively. For Japanese corpora, PNE and Japanese Wikipedia, we used MeCab[11], a Japanese morphological analyzer. When two or more nouns appear consecutively in a Japanese sentence, we treated them as a single compound noun.

From the Life Science Dictionary, we found 1213 translation pairs of which both the English and Japanese words occur at least 10 times in MEDLINE and PNE, re-

[4] http://www.ncbi.nlm.nih.gov/pubmed
[5] http://lifesciencedb.jp/pne/
[6] http://lsd.pharm.kyoto-u.ac.jp/ja/index.html
[7] http://en.wikipedia.org
[8] http://www2.nict.go.jp/out-promotion/techtransfer/EDR/J_index.html
[9] http://www.nactem.ac.uk/GENIA/tagger/
[10] https://code.google.com/p/hunpos/
[11] http://taku910.github.io/mecab/

spectively. From the EDR Lexicon, we found 2012 such translation pairs occurring in Wikipedia. These translations make the gold translations used in our evaluation; these translations are further split into the seed, test and development sets.

From the remaining Japanese words in each lexicon, we also extracted those which occur in the Japanese corpus at least 10 times but its the English counterpart does not, to use them as decoys (incorrect translation candidates).

The candidate translation set consists of all these decoys, plus all the Japanese words in the gold test set. Hence in the evaluation, for each source (English) word chosen from the gold test set, the system must rank its gold translation higher than all the decoy translations as well as those in the gold test set that are not the translation of the given source word (i.e., remaining translation pairs in the gold test set are also used as additional decoys for this specific source word).

For our experiments, gold translation pairs must be split into seed, development, and test sets. The development set is necessary because some of the compared methods have parameters that need to be tuned; see Section 3.5.1 for detail. We retained 60% of the entire data for the seed set, and 20% each for the development and test sets. We made different splits at random four times, and report the averaged results over these four trials.

In addition, to evaluate how much the performance depends on the number of seed translation pairs, we report the results when the number of seeds is changed; we tested with 20%, and 40% of the entire data as the seed set. The exact number of seed translation pairs for each seed size (20%, 40%, 60%) are 243, 486, and 727 in MEDLINE-PNE, and 420, 840, and 1262 in Wikipedia, respectively.

Feature vector construction

We followed the feature vector construction described in Sections 3.3. To make the context vector $v(s)$ in Eq. (3.1), we counted the frequency of co-occurring content words (not just nouns) in the 4-word window on each side in each corpus. The same words occurring on the left or right of the term are counted as distinct features of $v(s)$, whereas they are treated as an identical feature if both occur on the same side of the term (i.e., distance from the term is ignored as long as they are on the same side). Following related works [Tamura et al., 2012; Vulić and Moens, 2013b], the frequency is converted to the positive pointwise mutual information between the term and the feature, which is then used as the component of the context vector $v(s)$ of the

term. Finally, we construct the distributional similarity vectors computed by Eq. (3.1).

Compared methods

We compared the following methods.

- cos: Cosine similarity of raw feature vectors. This was often used in bilingual lexicon extraction as a baseline.
- ip: Inner product of raw feature vectors.
- centering: Inner product of centered feature vectors. This is the proposed approach which we discussed in Section 3.4.2. Following Suzuki et al. [2013], we normalized feature vectors by ℓ_2 norm before centering. For centering, we used the centroid of source words (Eq. (3.4)).
- mp: Mutual proximity of raw feature vectors with cosine distance. As described in Section 3.4.3, we used Eq. (3.10) as the similarity measurement.
- lp: Our reimplementation of label propagation-based method [Tamura et al., 2012]. This method makes two graphs, one for the source language and the other for the target language. Nodes of the graphs represent words, and edge weights are determined by the monolingual distributional similarity between words. It then runs a label propagation algorithm [Zhu et al., 2003] on these graphs, regarding seed words as labels. This results in a label distribution assigned to words, which in effect can be taken as a feature vector (of dimension of the seed size). The final similarity measurement is done with cosine of these feature vectors. To construct graphs, we used k-nearest neighbor graph which retains only the largest k edges in each node.

Parameters in lp (the number k of retained components in graph construction, and the number of iterations t for label propagation) are optimized with the development translation pairs. We computed the mean reciprocal rank over the development set for the range of parameters $k \in \{1, 10, 50, 100, 200, 300\}$, and chose the best one to apply for the test data.

Evaluation criteria

Since we formulate bilingual lexicon extraction as a task of ranking possible translation words, we use the mean reciprocal rank (MRR) as the main evaluation criterion. MRR is one of the standard evaluation criteria for ranking methods, and is defined as

$$\mathrm{MRR} = \frac{1}{|\mathcal{S}|}\sum_{s\in\mathcal{S}}\frac{1}{\mathrm{rank}(s)},$$

where $\mathcal{S}$ is the set of source words (queries) in the test set. and $\mathrm{rank}(s)$ is the similarity rank of the gold translation for the source word.

We also report the top j accuracies for $j = 1$ and 10. The top j accuracy represents the frequency of gold translation words present in the top j candidates:

$$\mathrm{Acc}_j = \frac{|\{s \in \mathcal{S} \mid \mathrm{rank}(s) \leq j\}|}{|\mathcal{S}|}.$$

As mentioned earlier, we report the average of these performance indices over four different random splits of the gold translations.

We verify our claim that hubs in the data degrade the accuracy of bilingual lexicon extraction. To this end, following the literature of hubness research [Radovanović et al., 2010a; Suzuki et al., 2013], we used the *skewness* of the N_j distribution as the indicator of how much a method suffers from the hubness effect. The N_j distribution is the empirical distribution of the number $N_j(t)$ of times each target (Japanese) word t occurs in the top j ranking over all source (English) words. Let $\mathcal{T}$ be the set of translation candidates (target words in the test set), and $N_j(t)$ be the N_j count of the translation candidate t (target word). The skewness of N_j distribution is defined as follows:

$$(N_j \text{ skewness}) = \frac{1}{\sigma^3}\frac{\sum_{t\in\mathcal{T}}\left(N_j(t)-\mu\right)^3}{|\mathcal{T}|}.$$

Here, μ and σ are respectively the (empirical) mean and standard deviation of N_j distribution. A high N_j skewness indicates a strong bias in the frequency of objects appearing in the j-nearest neighbors of other objects; i.e., emergence of hub words in target language (Japanese). We compute the N_j distribution of the translation candidates in the test set in our data splits. Again, we report the average over four trials.

Table 3.2: Experimental results. Mean reciprocal rank (MRR), top j accuracies (Acc_j), and N_j skewness. The bold figure indicates the best performers in each of the performance indices (the higher the better) and the N_j skewness (the lower the better).

(a) MEDLINE-PNE

	MRR	Acc_1	Acc_{10}	N_1 skewness	N_{10} skewness
cos	0.179	0.119	0.286	10.89	5.82
ip	0.084	0.042	0.148	17.99	8.35
centering	**0.291**	**0.199**	**0.459**	4.34	**2.16**
mp	0.281	0.192	0.447	**3.21**	2.23
lp	0.266	0.182	0.417	4.74	3.80

(b) Wikipedia

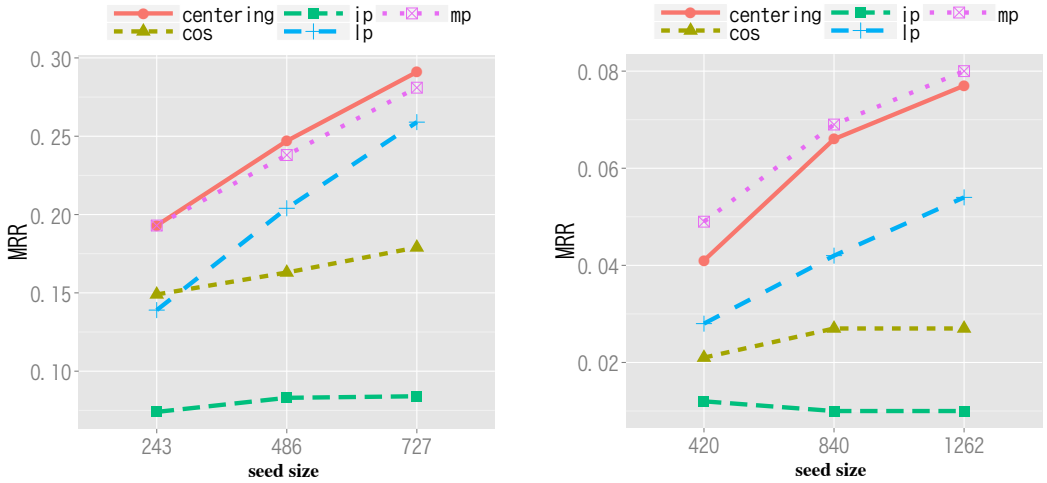
	MRR	Acc_1	Acc_{10}	N_1 skewness	N_{10} skewness
cos	0.027	0.013	0.049	20.40	12.79
ip	0.010	0.001	0.021	44.35	14.41
centering	0.077	0.039	**0.150**	5.43	3.39
mp	**0.080**	**0.043**	0.148	**4.05**	**2.73**
lp	0.054	0.027	0.104	13.84	14.27

3.5.2 Experimental Results and Discussion

The effect of hubs

Table 3.2 shows the results of bilingual lexicon extraction. As the table shows, centering performed best in terms of MRR on MEDLINE-PNE, and mutual proximity (mp) came close second. On Wikipedia, mutual proximity performed best, followed by centering. Label propagation (lp) comes third on both data sets. Centering and mutual proximity also performed best in terms of top j accuracies.

In terms of N_j skewness, baseline methods (cos and ip) had the relatively high value on both datasets. That is, the hub target words are emerged, and hence, the methods performed poorly. Centering and mutual proximity both reduced the N_j skewness dramatically compared with baseline methods, meaning that they effectively suppressed hub translation candidates. This results in improving MRR and accuracies.



(a) MEDLINE-PNE (b) Wikipedia

Figure 3.1: Dependency on the number of seed translation pairs.

Robustness with the limited number of seeds

Next we reduced the size of the seed set, to see the robustness of the approaches when the number of available seeds is limited. The results are shown in Figure 3.1. Centering and mutual proximity performed better than other methods over all seed sizes.

Robustness with word frequency

To investigate the effect of word frequency, following Tamura et al. [2012], we first split test data to two sets: the set of high frequency words whose frequencies are more than 50 (MEDLINE-PNE: 76.25 words, Wikipedia: 206.25 words), and the set of low frequency words whose frequencies are less than or equal to 50 (MEDLINE-PNE: 166.75 words, Wikipedia: 213.75 words). And then, we individually evaluated two sets.

Table 3.3 shows the results in this setting. In both datasets, centering and mutual proximity reduced the emergence of hubs, and hence, obtained the better MRR and accuracies compared with the others.

Table 3.3: The effect of word frequency. High frequency words (freq. $>$ 50) and low frequency words (freq. $\leq$ 50).

(a) MEDLINE-PNE

	Freq. $>$ 50			Freq. $\leq$ 50		
method	MRR	Acc_1	Acc_{10}	MRR	Acc_1	Acc_{10}
cos	0.362	0.255	0.560	0.040	0.019	0.075
ip	0.141	0.064	0.281	0.015	0.001	0.035
centering	**0.536**	**0.412**	**0.759**	**0.109**	**0.060**	**0.206**
mp	0.506	0.382	0.722	0.107	**0.060**	0.189
lp	0.505	0.382	0.702	0.077	0.043	0.139

(b) Wikipedia

	Freq. $>$ 50			Freq. $\leq$ 50		
method	MRR	Acc_1	Acc_{10}	MRR	Acc_1	Acc_{10}
cos	0.097	0.059	0.164	0.014	0.007	0.025
ip	0.059	0.033	0.090	0.004	0.000	0.007
centering	**0.182**	0.105	**0.325**	0.046	0.019	0.095
mp	**0.182**	**0.108**	0.324	**0.055**	**0.026**	**0.107**
lp	0.161	0.095	0.291	0.031	0.011	0.069

Similarity between the centroid and hub words

Table 3.4 shows that hub words which were frequently extracted as the translation words with cosine similarity (cos in Table 3.2). The table also shows the similarity between hub words and the centroid and its ranking.

We observed that hub words (i.e., objects with higher N_{10} value) entirely consisted of words with higher cosine similarity to the centroid, as anticipated by the theory of hubness [Radovanović et al., 2010a,b; Suzuki et al., 2013].

In MEDLINE-PNE, "細菌染色体 (*bacterial chromosome*)" was extracted 124 times. That is, the word was decided as the translation of roughly half of the source words. We also observed similar result in Wikipedia (see "多数 (*acres*)" in Table 3.4).

From these observations, we can conclude that the emergence of hub words affects

the performance of bilingual lexicon extraction.

The effect of centroid

As already mentioned in Section 3.4.2, we can consider three centroids: the average of source words $\bar{\mathbf{s}}$ in Eq. (3.4), the average of target words $\bar{\mathbf{t}}$ in Eq. (3.5), and the average of target and source words $\mathbf{c}$ in Eq. (3.6).

Table 3.5 shows the effect of centering with three centroids. As expected, $\bar{\mathbf{s}}$ obtained the most lowest N_j skewness compared with the others: i.e., centering with $\bar{\mathbf{s}}$ reduced the emergence of hubs effectively. In terms of MRR and Acc_j, on MEDLINE-PNE, $\bar{\mathbf{s}}$ also obtained the better results. On the other hand, $\mathbf{c}$ had the best results on Wikipedia. Note, however, that the difference of results was small, and $\bar{\mathbf{s}}$ obtained the better results with respect to Acc_{10}.

3.6 Summary

We have shown that the hubness phenomenon severely harms the performance of vector space-based methods for bilingual lexicon extraction. To reduce the effect of this phenomenon, we have used centering and mutual proximity to reduce hubs, and shown that they consequently improved the performance.

In future work, we plan to investigate the influence of hubs in other methods, such as those based on topic models [Liu et al., 2013; Mimno et al., 2009; Vulić and Moens, 2013a] and linear classifier [Aker et al., 2013; Irvine and Callison-Burch, 2013; Laws et al., 2010].

Table 3.4: The list of top 15 hub words. The hub words have the top 15 largest N_{10} value which represents the number of times each words is found in the nearest neighbors. The "Sim." indicates the similarity between hub word and data mean.

(a) MEDLINE-PNE

Ranking of N_{10}	Hub word	N_{10}	Sim.	Ranking of sim.
1	細菌染色体 (*bacterial chromosome*)	124	0.835	1
2	カルシウム (*calcium*)	114	0.816	2
3	肝細胞 (*stem cell*)	62	0.805	3
4	アデニル酸シクラーゼ (*adenylate cyclase*)	60	0.790	5
5	ガングリオシド (*ganglioside*)	52	0.789	6
6	腫瘍 (*neoplasm*)	50	0.768	26
7	造血幹細胞 (*hematopoietic stem cell*)	48	0.784	8
8	トランスフェリン (*transferrin*)	47	0.781	10
9	実験動物 (*laboratory animal*)	47	0.792	4
10	カスパーゼ (*caspase*)	41	0.781	11
11	ユビキチン (*ubiquitin*)	41	0.772	21
12	膜脂質 (*membrane lipid*)	41	0.776	15
13	転写制御因子 (*transcription factor*)	41	0.784	7
14	酸素 (*oxygen*)	40	0.739	70
15	オカダ酸 (*okadaic acid*)	39	0.759	38

(b) Wikipedia

Ranking of N_{10}	Hub word	N_{10}	Sim.	Ranking of sim.
1	多数 (*acres*)	296	0.775	1
2	場所 (*ll*)	235	0.770	2
3	以下 (*below*)	191	0.766	3
4	最近 (*late*)	184	0.765	4
5	仕事 (*toil*)	178	0.763	5
6	一部 (*lith*)	156	0.762	6
7	建物 (*bigging*)	133	0.759	7
8	デザイン (*design*)	131	0.759	8
9	将来 (*fut.*)	119	0.758	9
10	少数 (*decimal*)	95	0.753	10
11	従来 (*erenow*)	90	0.752	11
12	当初 (*primitively*)	72	0.752	12
13	過去 (*past*)	69	0.749	15
14	手 (*leg-up*)	65	0.750	14
15	意味 (*circumstances*)	60	0.746	19

Table 3.5: The results with three data means.

(a) MEDLINE-PNE

centroid	MRR	Acc_1	Acc_{10}	N_1 skew.	N_{10} skewness
$\bar{\mathbf{s}}$	**0.291**	**0.199**	**0.459**	**4.34**	**2.16**
$\bar{\mathbf{t}}$	0.275	0.183	0.441	6.31	3.31
$\mathbf{c}$	0.280	0.186	0.454	6.13	2.99

(b) Wikipedia

centroid	MRR	Acc_1	Acc_{10}	N_1 skew.	N_{10} skewness
$\bar{\mathbf{s}}$	0.077	0.039	**0.150**	**5.43**	**3.39**
$\bar{\mathbf{t}}$	0.076	0.041	0.143	11.99	6.39
$\mathbf{c}$	**0.078**	**0.043**	0.146	11.25	5.54

Chapter 4

Zero-Shot Learning with Hubness Reduction

4.1 Introduction

4.1.1 Background

In recent years, *zero-shot learning* (ZSL) [Farhadi et al., 2009; Lampert et al., 2009; Larochelle et al., 2008; Palatucci et al., 2009] has been an active research topic in machine learning, computer vision, and natural language processing. Many practical applications can be formulated as a ZSL task: drug discovery [Larochelle et al., 2008], bilingual lexicon extraction [Dinu and Baroni, 2014, 2015; Mikolov et al., 2013b], and image labeling [Akata et al., 2014; Frome et al., 2013; Norouzi et al., 2014; Palatucci et al., 2009; Socher et al., 2013], to name a few. Cross-lingual information retrieval [Vinokourov et al., 2002] can also be viewed as a ZSL task.

ZSL can be regarded as a type of (multi-class) classification problem, in the sense that the classifier is given a set of known example-class label pairs (training set), with the goal to predict the unknown labels of new examples (test set). However, ZSL differs from the standard classification in that the labels for the test examples are not present in the training set. In standard settings, the classifier chooses, for each test example, a label among those observed in the training set, but this is not the case in ZSL. Moreover, the number of class labels can be huge in ZSL; indeed, in bilingual lexicon extraction, labels correspond to possible translation words, which can range over entire vocabulary of the target language.

Obviously, the task would be intractable without further assumptions. Labels are

thus assumed to be embedded in a metric space (*label space*), and their distance (or similarity) can be measured in this space[1]. Such a label space can be built with the help of background knowledge or external resources; in image labeling tasks, for example, labels correspond to annotation keywords, which can be readily represented as vectors in a Euclidean space, either by using corpus statistics in a standard way, or by using the more recent techniques for learning word representations, such as the continuous bag-of-words or skip-gram models [Mikolov et al., 2013a].

After a label space is established, one natural approach would be to use a regression technique on the training set to obtain a mapping function from the example space to the label space. This function could then be used for mapping unlabeled examples into the label space, where nearest neighbor search is carried out to find the label closest to the mapped example. Finally, this label would be output as the prediction for the example.

To find the mapping function, some researchers use the standard linear ridge regression [Dinu and Baroni, 2014, 2015; Mikolov et al., 2013b; Palatucci et al., 2009], whereas others use neural networks [Frome et al., 2013; Norouzi et al., 2014; Socher et al., 2013].

In the machine learning community, meanwhile, the *hubness phenomenon* [Radovanović et al., 2010a] is attracting attention as a new type of the "curse of dimensionality." This phenomenon is concerned with nearest neighbor methods in high-dimensional space, and states that a small number of objects in the dataset, or *hubs*, may occur as the nearest neighbor of many objects. The emergence of these hubs will diminish the utility of nearest neighbor search, because the list of nearest neighbors often contain the same hub objects regardless of the query object for which the list is computed.

4.1.2 Research Objective and Contributions

In this chapter, we show that the interaction between the regression step in ZSL and the subsequent nearest neighbor step has a non-negligible effect on the prediction accuracy.

In ZSL, examples and labels are represented as vectors in high-dimensional space, of which the dimensionality is typically a few hundred. As demonstrated by Dinu and Baroni [2015] (see also Section 4.6), when ZSL is formulated as a problem of ridge regression from examples to labels, "hub" labels emerge, which are simultaneously the nearest

[1] Throughout the chapter, we assume both the example and label spaces are Euclidean.

neighbors of many mapped examples. This has the consequence of incurring bias in the prediction, as these labels are output as the predicted labels for these examples. The presence of hubs are not necessarily disadvantageous in standard classification settings; there may be "good" hubs as well as "bad" hubs [Radovanović et al., 2010a]. However, in typical ZSL tasks in which the label set is fine-grained and huge, hubs are nearly always harmful to the prediction accuracy.

Therefore, the objective of this study is to investigate ways to suppress hubs, and to improve the ZSL accuracy. Our contributions can be summarized as follows.

1. We analyze the mechanism behind the emergence of hubs in ZSL, both with ridge regression and ordinary least squares. It is established that hubness occurs in ZSL not only because of high-dimensional space, but also because ridge regression has conventionally been used in ZSL in a way that *promotes* hubness. To be precise, the distributions of the mapped examples and the labels are different such that hubs are likely to emerge.

2. Drawing on the above analysis, we propose using ridge regression to map labels into the space of examples. This approach is contrary to that followed in existing work on ZSL, in which examples are mapped into label space. Our proposal is therefore to change the mapping direction.

   As shown in Section 4.6, our proposed approach outperformed the existing approach in an empirical evaluation using both synthetic and real data.

3. In terms of contributions to the research on hubness, this research is the first to provide in-depth analysis of the situation in which the query and data follow different distributions, and to show that the variance of the data distribution matters to hubness. In particular, in Section 4.3, we provide a proposition in which the degree of bias present in the data, which causes hub formation, is expressed as a function of the data variance. In Section 4.4, this proposition serves as the main tool for analyzing hubness in ZSL.

4.2 Zero-Shot Learning as a Regression Problem

Let X be a set of examples, and Y be a set of class labels. In ZSL, not only examples but also labels are assumed to be vectors. For this reason, examples are sometimes referred to as *source objects*, and labels as *target objects*. In the subsequent sections

of this chapter, we mostly follow this terminology when referring to the members of X and Y.

Let $X \subset \mathbb{R}^c$ and $Y \subset \mathbb{R}^d$. These spaces, $\mathbb{R}^c$ and $\mathbb{R}^d$, are called *source space* and *target space*, respectively. Although X can be the entire space $\mathbb{R}^c$, Y is usually a finite set of points in $\mathbb{R}^d$, even though its size may be enormous in some problems.

Let $X_{\text{train}} = \{\mathbf{x}_i \mid i = 1, \ldots, n\}$ be the training examples (training source objects), and $Y_{\text{train}} = \{\mathbf{y}_i \mid i = 1, \ldots, n\}$ be their labels (training target objects); i.e., the class label of example $\mathbf{x}_i$ is $\mathbf{y}_i$, for each $i = 1, \ldots, n$. In a standard classification setting, the labels in the training set are equal to the entire set of labels; i.e., $Y_{\text{train}} = Y$. In contrast, this assumption is not made in ZSL, and Y_{train} is a strict subset of Y. Moreover, it is assumed that the true class labels of test examples do not belong to Y_{train}; i.e., they belong to $Y \backslash Y_{\text{train}}$.

In such a situation, it is difficult to find a function f that maps $\mathbf{x} \in X$ directly to a label in Y. Therefore, a popular (and also natural) approach is to learn a projection $m : \mathbb{R}^c \to \mathbb{R}^d$, which can be done with a regression technique. With a projection function m at hand, the label of a new source object $\mathbf{x} \in \mathbb{R}^c$ is predicted to be the one closest to the mapped point $m(\mathbf{x})$ in the target space. The prediction function f is thus given by

$$f(\mathbf{x}) = \arg\min_{\mathbf{y} \in Y} \|m(\mathbf{x}) - \mathbf{y}\|.$$

After a source object $\mathbf{x}$ is projected to $m(\mathbf{x})$, the task is reduced to that of nearest neighbor search in the target space.

4.3 Hubness Phenomenon and the Variance of Data

The utility of nearest neighbor search would be significantly reduced if the same objects were to appear consistently as the search result, irrespective of the query. Radovanović et al. [2010a] showed that such objects, termed *hubs*, indeed occur in high-dimensional space. Although this phenomenon may seem counter-intuitive, hubness is observed in a variety of real datasets and distance/similarity measures used in combination [Radovanović et al., 2010a; Schnitzer et al., 2012; Suzuki et al., 2013].

The aim of this study is to analyze the hubness phenomenon in ZSL, which involves nearest neighbor search in high-dimensional space as the last step. However, as a tool for analyzing ZSL, the existing theory on hubness [Radovanović et al., 2010a] is inadequate, as it was mainly developed for comparing the emergence of hubness in spaces of different dimensionalities.

In the analysis of ZSL in Section 4.4.2, we aim to compare two distributions in the same space, but which differ in terms of *variance*. To this end, we first present a proposition below, which is similar in spirit to the main theorem of Radovanović et al. [2010a, Theorem 1], but which distinguishes the query and data distributions, and also expresses the expected difference between the squared distances from queries to database objects in terms of their variance.

The proposition is concerned with nearest neighbor search, in which $\mathbf{s}$ is a query, and $\mathbf{t}_1$ and $\mathbf{t}_2$ are two objects in a dataset. In the context of ZSL as formulated in Section 4.2, $\mathbf{s}$ represents the image of a source object in the target space (through the learned regression function m), and $\mathbf{t}_1$ and $\mathbf{t}_2$ are target objects (labels) lying at different distances from the origin. We are interested in which of $\mathbf{t}_1$ and $\mathbf{t}_2$ are more likely to be closer to $\mathbf{s}$, when $\mathbf{s}$ is sampled from a distribution $\mathcal{S}$ with zero mean.

Let $E_{\mathcal{X}}[\cdot]$ and $\mathrm{Var}_{\mathcal{X}}[\cdot]$ respectively denote the expectation and variance under a distribution $\mathcal{X}$, and let $\mathcal{N}(\mu,\Sigma)$ be a multivariate normal distribution with mean μ and covariance matrix Σ.

Proposition 1. *Let $\mathbf{t} = [t_1,\ldots,t_d]^{\mathrm{T}}$ be a d-dimensional random vector, with components t_i $(i = 1,\ldots,d)$ sampled i.i.d. from a normal distribution with zero mean and variance σ^2; i.e., $\mathbf{t} \sim \mathcal{T}$, where $\mathcal{T} = \mathcal{N}(\mathbf{0}, s^2\mathbf{I})$. Further let $\eta = \sqrt{\mathrm{Var}_{\mathcal{T}}[\|\mathbf{t}\|^2]}$ be the standard deviation of the squared norm $\|\mathbf{t}\|^2$.*

Consider two fixed samples $\mathbf{t}_1$ and $\mathbf{t}_2$ of random vector $\mathbf{t}$, such that the squared norms of $\mathbf{t}_1$ and $\mathbf{t}_2$ are $\gamma\eta$ apart. In other words,

$$\|\mathbf{t}_2\|^2 - \|\mathbf{t}_1\|^2 = \gamma\eta.$$

Let $\mathbf{s}$ be a point sampled from a distribution $\mathcal{S}$ with zero mean. Then, the expected difference Δ between the squared distances from $\mathbf{t}_1$ and $\mathbf{t}_2$ to $\mathbf{s}$, i.e.,

$$\Delta = \mathrm{E}_{\mathcal{S}}\left[\|\mathbf{s}-\mathbf{t}_2\|^2\right] - \mathrm{E}_{\mathcal{S}}\left[\|\mathbf{s}-\mathbf{t}_1\|^2\right] \tag{4.1}$$

is given by

$$\Delta = \sqrt{2}\gamma d^{1/2}\sigma^2. \tag{4.2}$$

Proof. For $i = 1,2$, the distance between a point $\mathbf{s}$ and $\mathbf{t}_i$ is given by

$$\|\mathbf{s}-\mathbf{t}_i\|^2 = \|\mathbf{s}\|^2 + \|\mathbf{t}_i\|^2 - 2\mathbf{s}^{\mathrm{T}}\mathbf{t}_i,$$

and its expected value is

$$\mathrm{E}_{\mathcal{S}}\left[\|\mathbf{s}-\mathbf{t}_i\|^2\right] = \mathrm{E}_{\mathcal{S}}\left[\|\mathbf{s}\|^2\right] + \|\mathbf{t}_i\|^2 - 2\,\mathrm{E}_{\mathcal{S}}\left[\mathbf{s}\right]^{\mathrm{T}}\mathbf{t}_i = \mathrm{E}_{\mathcal{S}}\left[\|\mathbf{s}\|^2\right] + \|\mathbf{t}_i\|^2,$$

since $\mathrm{E}_{\mathcal{S}}\left[\mathbf{s}\right] = 0$ by assumption. Substituting this equality in Eq. (4.1) yields

$$\Delta = \overbrace{\left(\mathrm{E}_{\mathcal{S}}\left[\|\mathbf{s}\|^2\right] + \|\mathbf{t}_2\|^2\right)}^{\mathrm{E}_{\mathcal{S}}\left[\|\mathbf{s}-\mathbf{t}_2\|^2\right]} - \overbrace{\left(\mathrm{E}_{\mathcal{S}}[\|\mathbf{s}\|^2] + \|\mathbf{t}_1\|^2\right)}^{\mathrm{E}_{\mathcal{S}}\left[\|\mathbf{s}-\mathbf{t}_1\|^2\right]} = \|\mathbf{t}_2\|^2 - \|\mathbf{t}_1\|^2 = \gamma\eta. \tag{4.3}$$

Now, it is well known that if a d-dimensional random vector $\mathbf{z}$ follows the multivariate standard normal distribution $\mathcal{N}(\mathbf{0},\mathbf{I})$, then its squared norm $\|\mathbf{z}\|^2$ follows the chi-squared distribution with d degrees of freedom, and its variance is $2d$. Since $\mathbf{t} = \sigma\mathbf{z}$, the variance η^2 of the squared norm $\|\mathbf{t}\|^2$ is

$$\eta^2 = \mathrm{Var}_{\mathcal{T}}\left[\|\mathbf{t}\|^2\right] = \mathrm{Var}_{\mathcal{Z}}\left[\sigma^2\|\mathbf{z}\|^2\right] = \sigma^4\,\mathrm{Var}_{\mathcal{Z}}\left[\|\mathbf{z}\|^2\right] = 2d\sigma^4. \tag{4.4}$$

From (4.3) and (4.4), we obtain $\Delta = \gamma\sigma^2\sqrt{2d}$. □

Note that in Proposition 1, the standard deviation σ is used as a yardstick of measurement to allow for comparison of "similarly" located object pairs across different distributions; two object pairs in different distributions are regarded as similar if objects in each pair are $\gamma\sigma$ apart as measured by the σ for the respective distributions, but has an equal factor γ. This technique is due to Radovanović et al. [2010a].

Now, Δ represents the expected difference between the squared distances from $\mathbf{s}$ to $\mathbf{t}_1$ and $\mathbf{t}_2$. Equation (4.2) shows that Δ increases with γ, the factor quantifying the amount of difference $\|\mathbf{t}_2\|^2 - \|\mathbf{t}_1\|^2$. This suggests that a query object sampled from $\mathcal{S}$ is more likely to be closer to object $\mathbf{t}_1$ than to $\mathbf{t}_2$, if $\|\mathbf{t}_1\|^2 < \|\mathbf{t}_2\|^2$; i.e., $\mathbf{t}_1$ is closer to the origin than $\mathbf{t}_2$ is. Because this holds for any pair of objects $\mathbf{t}_1$ and $\mathbf{t}_2$ in the dataset, we can conclude that the objects closest to the origin in the dataset tend to be hubs.

Equation (4.2) also states the relationship between Δ and the component variance σ^2 of distribution $\mathcal{T}$, by which the following is implied: For a fixed query distribution $\mathcal{S}$, if we have two choices for distributions for $\mathbf{t}$, $\mathcal{T}_1 = \mathcal{N}(\mathbf{0},\sigma_1^2\mathbf{I})$ and $\mathcal{T}_2 = \mathcal{N}(\mathbf{0},\sigma_2^2\mathbf{I})$ with $\sigma_1^2 < \sigma_2^2$, it is preferable to choose $\mathcal{T}_1$, i.e., the distribution with a smaller variance, when attempting to reduce hubness. Indeed, assuming the independence of $\mathcal{S}$ and $\mathcal{T}$, we can show that the influence of Δ relative to the expected squared distance from $\mathbf{s}$ to $\mathbf{t}$ (which is also subject to whether $\mathbf{t} \sim \mathcal{T}_1$ or $\mathcal{T}_2$), is weaker for $\mathcal{T}_1$ than for $\mathcal{T}_2$.

Corollary 1. *Let* $\mathcal{T}_1 = \mathcal{N}(\mathbf{0}, \sigma_1^2\mathbf{I})$ *and* $\mathcal{T}_2 = \mathcal{N}(\mathbf{0}, \sigma_2^2\mathbf{I})$ *with* $\sigma_1^2 < \sigma_2^2$*, and let* $\mathbf{s}$*,* $\mathbf{t}_1$*,* $\mathbf{t}_2$*, and* Δ *be as defined in Proposition 1. Then,*

$$\frac{\Delta(\gamma, d, \sigma_1)}{E_{\mathcal{S},\mathcal{T}_1}[\|\mathbf{s}-\mathbf{t}\|^2]} < \frac{\Delta(\gamma, d, \sigma_2)}{E_{\mathcal{S},\mathcal{T}_2}[\|\mathbf{s}-\mathbf{t}\|^2]},$$

where we wrote Δ *explicitly as a function of* γ*,* d*, and* σ*.*

4.4 Hubness in Regression-Based Zero-Shot Learning

In this section, we analyze the emergence of hubs in the nearest neighbor step of ZSL. Through the analysis, it is shown that hubs are promoted by the use of ridge regression in the existing formulation of ZSL, i.e., mapping source objects (examples) into the target (label) space.

As a solution, we propose using ridge regression in a direction opposite to that in existing work. That is, we project target objects in the space of source objects, and carry out nearest neighbor search in the source space. Our argument for this approach consists of three steps.

1. We first show in Section 4.4.1 that, with ridge regression (and ordinary least squares as well), mapped observation data tend to lie closer to the origin than the target responses do. Because the existing work formulates ZSL as a regression problem that projects source objects into the target space, this means that the norm of the projected source objects tends to be smaller than that of target objects.

2. By combining the above result with the discussion in Section 4.3, we then argue that placing source objects closer to the origin is not ideal from the perspective of reducing hubness. On the contrary, placing target objects closer to the origin, as attained with the proposed approach, is more desirable (Section 4.4.2).

3. In Section 4.4.3, we present a simple additional argument against placing source objects closer to the origin; if the data is unimodal, such a configuration increases the possibility of another target object falling closer to the source object. This argument diverges from the discussion on hubness, but again justifies the proposed approach.

4.4.1 Shrinking the Projected Objects

We first prove that ridge regression tends to map observation data closer to the origin of the space. This tendency may be easily observed in ridge regression, for which the penalty term shrinks the estimated coefficients towards zero. However, the above tendency is also inherent in ordinary least squares.

Let $\|\cdot\|_{\mathrm{F}}$ and $\|\cdot\|_2$ respectively denote the Frobenius norm and the 2-norm of matrices.

Proposition 2. *Let $\mathbf{M} \in \mathbb{R}^{d\times c}$ be the solution for ridge regression with an observation matrix $\mathbf{A} \in \mathbb{R}^{c\times n}$ and a response matrix $\mathbf{B} \in \mathbb{R}^{d\times n}$; i.e.,*

$$\mathbf{M} = \arg\min_{\mathbf{M}} \left(\|\mathbf{MA} - \mathbf{B}\|_{\mathrm{F}}^2 + \alpha \|\mathbf{M}\|_{\mathrm{F}} \right). \tag{4.5}$$

where $\alpha \geq 0$ is a hyperparameter. Then, we have $\|\mathbf{MA}\|_2 \leq \|\mathbf{B}\|_2$.

Sketch. It is well known that $\mathbf{M} = \mathbf{B}\mathbf{A}^{\mathrm{T}} \left(\mathbf{A}\mathbf{A}^{\mathrm{T}} + \alpha\mathbf{I}\right)^{-1}$. Thus we have

$$\|\mathbf{MA}\|_2 = \|\mathbf{B}\mathbf{A}^{\mathrm{T}} \left(\mathbf{A}\mathbf{A}^{\mathrm{T}} + \alpha\mathbf{I}\right)^{-1}\mathbf{A}\|_2 \leq \|\mathbf{B}\|_2 \, \|\mathbf{A}^{\mathrm{T}} \left(\mathbf{A}\mathbf{A}^{\mathrm{T}} + \alpha\mathbf{I}\right)^{-1}\mathbf{A}\|_2. \tag{4.6}$$

Let λ be the largest singular value of $\mathbf{A}$. It can be shown that

$$\|\mathbf{A}^{\mathrm{T}} \left(\mathbf{A}\mathbf{A}^{\mathrm{T}} + \alpha\mathbf{I}\right)^{-1}\mathbf{A}\|_2 = \frac{\lambda^2}{\lambda^2 + \alpha} \leq 1. \tag{4.7}$$

Substituting this inequality in Eq. (4.6) establishes the proposition. □

Recall that if the data is centered, the matrix 2-norm can be interpreted as an indicator of the variance of data along its principal axis. Proposition 2 thus indicates that the variance along the principal axis of the mapped observations $\mathbf{MA}$ tends to be smaller than that of responses $\mathbf{B}$.

Furthermore, this tendency even persists in the ordinary least squares with no penalty term (i.e., $\alpha = 0$), since $\|\mathbf{MA}\|_2 \leq \|\mathbf{B}\|_2$ still holds in this case; note that $\mathbf{A}^{\mathrm{T}} \left(\mathbf{A}\mathbf{A}^{\mathrm{T}}\right)^{-1}\mathbf{A}$ is an orthogonal projection and its 2-norm is 1, but the inequality in Eq. (4.6) holds regardless. This tendency therefore cannot be completely eliminated by simply decreasing the ridge parameter α towards zero.

In existing work on ZSL, $\mathbf{A}$ represents the (training) source objects $\mathbf{X} = [\mathbf{x}_1 \cdots \mathbf{x}_n] \in \mathbb{R}^{c\times n}$, to be mapped into the space of target objects (by projection matrix $\mathbf{M}$); and $\mathbf{B}$ is

the matrix of labels for the training objects, i.e., $\mathbf{B} = \mathbf{Y} = [\mathbf{y}_1 \cdots \mathbf{y}_n] \in \mathbb{R}^{d \times n}$. Although Proposition 2 is thus only concerned with the training set, it suggests that the source objects at the time of testing, which are not in $\mathbf{X}$, are also likely to be mapped closer to the origin of the target space than many of the target objects in $\mathbf{Y}$.

4.4.2 Influence of Shrinking the Objects on Nearest Neighbor Search

We learned in Section 4.4.1 that ridge regression (and ordinary least squares) shrink the mapped observation data towards the origin of the space, relative to the response. Thus, in existing work on ZSL in which source objects X are projected to the space of target objects Y, the norm of the mapped source objects is likely to be smaller than that of the target objects.

The proposed approach, which was described in the beginning of Section 4.4, follows the opposite direction: target objects Y are projected to the space of source objects X. Thus, in this case, the norm of the mapped target objects is expected to be smaller than that of the source objects.

The question now is which of these configurations is preferable for the subsequent nearest neighbor step, and we provide an answer under the following assumptions: (i) The source space and the target space are of equal dimensions; (ii) the source and target objects are isotropically normally distributed and independent; and (iii) the projected data is also isotropically normally distributed, except that the variance has shrunk.

Let $\mathcal{D}_1 = \mathcal{N}(0, s_1^2\mathbf{I})$ and $\mathcal{D}_2 = \mathcal{N}(0, s_2^2\mathbf{I})$ be two multivariate normal distributions, with $s_1^2 < s_2^2$. We compare two configurations of source object $\mathbf{x}$ and target objects $\mathbf{y}$: (a) the one in which $\mathbf{x} \sim \mathcal{D}_1$ and $\mathbf{y} \sim \mathcal{D}_2$, and (b) the one in which $\mathbf{x}' \sim \mathcal{D}_2$ and $\mathbf{y}' \sim \mathcal{D}_1$ on the other hand; here, the primes in (b) were added to distinguish variables in two configurations.

These two configurations are intended to model situations in (a) existing work and (b) our proposal. In configuration (a), $\mathbf{x}$ is shorter in expectation than $\mathbf{y}$, and therefore this approximates the situation that arises from existing work. Configuration (b) represents the opposite situation, and corresponds to our proposal in which $\mathbf{y}$ is the projected vector and thus is shorter in expectation than $\mathbf{x}$.

Now, we aim to verify whether the two configurations differ in terms of the likeliness of hubs emerging, using Proposition 1. First, we scale the entire space of configuration (b) by (s_1/s_2), or equivalently, we consider transformation of the variables by $\mathbf{x}'' = (s_1/s_2)\mathbf{x}'$ and $\mathbf{y}'' = (s_1/s_2)\mathbf{y}'$. Note that because the two variables are scaled equally,

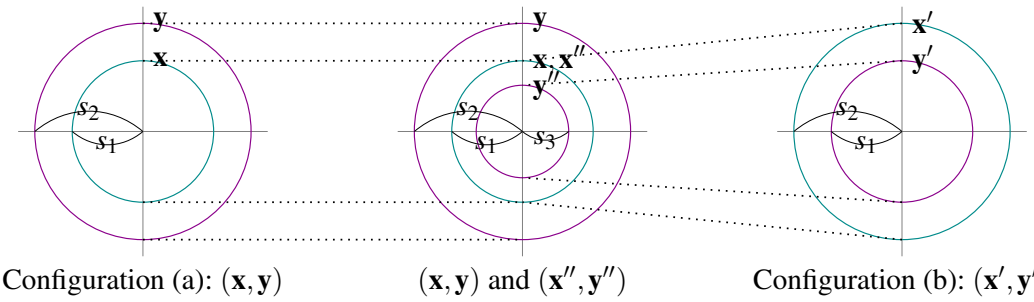


Figure 4.1: Schematic illustration for Section 4.4.2 in two-dimensional space. The left and the right panels depict configurations (a) and (b), respectively, with the center panel showing both configuration (a) and the scaled version of configuration (b) in the same space. A circle represents a distribution, with its radius indicating the standard deviation. The radius of the circles for $\mathbf{y}$ (on the left panel) and $\mathbf{x}'$ (right panel) is s_1, whereas that of the circles for $\mathbf{x}$ (left panel) and $\mathbf{y}'$ (right panel) is s_2, with $s_1 < s_2$. Circles $\mathbf{x}''$ and $\mathbf{y}''$ are the scaled versions of $\mathbf{x}'$ and $\mathbf{y}'$ such that the standard deviation (radius) of $\mathbf{x}''$ is equal to $\mathbf{x}$, which makes the standard deviation of $\mathbf{y}''$ equal to $s_3 = s_1^2/s_2$.

this change of variables preserves the nearest neighbor relations among the samples. See Fig. 4.1 for an illustration of the relationship among $\mathbf{x}$, $\mathbf{y}$, $\mathbf{x}'$, $\mathbf{y}'$, $\mathbf{x}''$, and $\mathbf{y}''$.

Let $\{x_i'\}$ and $\{y_i'\}$ be the components of $\mathbf{x}'$ and $\mathbf{y}'$, respectively, and let $\{x_i''\}$ and $\{y_i''\}$ be those for $\mathbf{x}''$ and $\mathbf{y}''$. Then we have

$$\mathrm{Var}[x_i''] = \mathrm{Var}\left[\frac{s_1}{s_2}x_i'\right] = \left(\frac{s_1}{s_2}\right)^2 \mathrm{Var}[x_i'] = s_1^2,$$
$$\mathrm{Var}[y_i''] = \mathrm{Var}\left[\frac{s_1}{s_2}y_i'\right] = \left(\frac{s_1}{s_2}\right)^2 \mathrm{Var}[y_i'] = \frac{s_1^4}{s_2^2}.$$

Thus, $\mathbf{x}''$ follows $\mathcal{N}(0, s_1^2\mathbf{I})$, and $\mathbf{y}''$ follows $\mathcal{N}(0, (s_1^4/s_2^2)\mathbf{I})$. Since both $\mathbf{x}$ in configuration (a) and $\mathbf{x}''$ above follow the same distribution, it now becomes possible to compare the properties of $\mathbf{y}$ and $\mathbf{y}''$ in light of the discussion at the end of Section 4.3: In order to reduce hubness, the distribution with a smaller variance is preferred to the one with a larger variance, for a fixed distribution of source $\mathbf{x}$ (or equivalently, $\mathbf{x}''$).

It follows that $\mathbf{y}''$ is preferable to $\mathbf{y}$, because the former has a smaller variance. As mentioned above, the nearest neighbor relation between the scaled variables, $\mathbf{y}''$ against $\mathbf{x}''$ (or equivalently $\mathbf{x}$), is identical to $\mathbf{y}'$ against $\mathbf{x}'$ in configuration (b). Therefore, we

conclude that configuration (b) is preferable to configuration (a), in the sense that the former is more likely to suppress hubs.

Finally, recall that the preferred configuration (b) models the situation of our proposed approach, which is to map target objects in the space of source objects.

4.4.3 Additional Argument for Placing Target Objects Closer to the Origin

By assuming a unimodal data distribution of which the probability density function (pdf) $p(\mathbf{z})$ is decreasing in $\|\mathbf{z}\|$, we are able to present the following proposition which also advocates placing the source objects outside the target objects, and not the other way around.

Proposition 3 is concerned with the placement of a source object $\mathbf{x}$ at a fixed distance r from its target object $\mathbf{y}$, for which we have two alternatives $\mathbf{x}_1$ and $\mathbf{x}_2$, located at different distances from the origin of the space.

Proposition 3. *Consider a finite set Y of objects (i.e., points) in a Euclidean space, sampled i.i.d. from a distribution whose pdf $p(\mathbf{z})$ is a decreasing function of $\|\mathbf{z}\|$. Let $\mathbf{y} \in Y$ be an object in the set, and let $r > 0$. Further let $\mathbf{x}_1$ and $\mathbf{x}_2$ be two objects at a distance r apart from $\mathbf{y}$. If $\|\mathbf{x}_1\| < \|\mathbf{x}_2\|$, then the probability that $\mathbf{y}$ is the closest object in Y to $\mathbf{x}_2$ is greater than that of $\mathbf{x}_1$.*

Sketch. For $i = 1, 2$, if another object in Y appears within distance r of $\mathbf{x}_i$, then $\mathbf{y}$ is not the nearest neighbor of $\mathbf{x}_i$. Thus, we aim to prove that this probability for $\mathbf{x}_2$ is smaller than that for $\mathbf{x}_1$. Since objects in Y are sampled i.i.d, it suffices to prove

$$\int_{\mathbf{z} \in V_2} dp(\mathbf{z}) \leq \int_{\mathbf{z} \in V_1} dp(\mathbf{z}), \tag{4.8}$$

where V_i $(i = 1, 2)$ denote the balls centered at $\mathbf{x}_i$ with radius r. However, Eq. (4.8) obviously holds because the balls V_1 and V_2 have the same radii, $p(\mathbf{z})$ is a decreasing function of $\|\mathbf{z}\|$, and $\|\mathbf{x}_1\| \leq \|\mathbf{x}_2\|$. See Figure 4.2 for an illustration with a bivariate standard normal distribution in two-dimensional space.

□

In the context of existing work on ZSL, which uses ridge regression to map source objects in the space of target objects, $\mathbf{x}$ can be regarded as a mapped source object, and

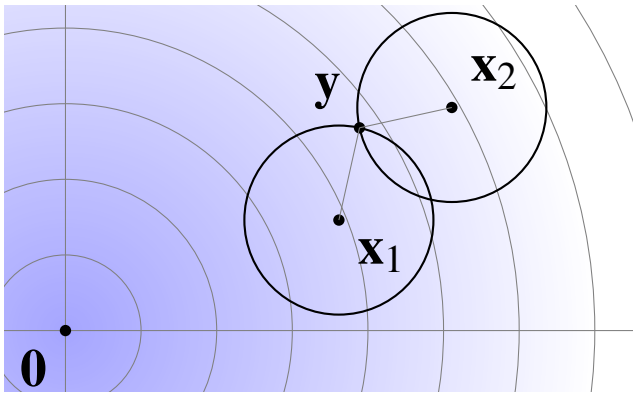


Figure 4.2: Illustration of the situation considered in Proposition 3. Here, it is assumed that $\|\mathbf{x}_1\| < \|\mathbf{x}_2\|$ and $\|\mathbf{y}-\mathbf{x}_1\| = \|\mathbf{y}-\mathbf{x}_2\|$. The intensity of the background shading represents the values of the pdf of a bivariate standard normal distribution, from which $\mathbf{y}$ and other objects (not depicted in the figure) in set Y are sampled. The probability mass inside the circle centered at $\mathbf{x}_1$ is greater than that centered at $\mathbf{x}_2$, as the intensity of the shading inside the two circles shows.

$\mathbf{y}$ as its target object. Proposition 3 implies that if we want to make a source object $\mathbf{x}$ the nearest neighbor of a target object $\mathbf{y}$, it should rather be placed farther than $\mathbf{y}$ from the origin, but this idea is not present in the objective function (Eq. (4.5)) for ridge regression; the first term of the objective allocates the same amount of penalty for $\mathbf{x}_1$ and $\mathbf{x}_2$, as they are equally distant from the target $\mathbf{y}$. On the contrary, the ridge regression actually *promotes* placement of the mapped source object $\mathbf{x}$ closer to the origin, as stated in Proposition 2.

4.4.4 Summary of the Proposed Approach

Drawing on the analysis presented in Sections 4.4.1–4.4.3, we propose performing regression that maps *target* objects in the space of *source* objects, and carry out nearest neighbor search in the source space. This opposes the approach followed in existing work on regression-based ZSL [Dinu and Baroni, 2014, 2015; Lazaridou et al., 2014; Mikolov et al., 2013a; Palatucci et al., 2009], which maps source objects into the space of target objects.

In the proposed approach, matrix $\mathbf{B}$ in Proposition 2 represents the source objects $\mathbf{X}$, and $\mathbf{A}$ represents the target objects $\mathbf{Y}$. Therefore, $\|\mathbf{MA}\|_2 \leq \|\mathbf{B}\|_2$ means $\|\mathbf{MY}\|_2 \leq$

$\|\mathbf{X}\|_2$, i.e., the mapped target objects tend to be placed closer than the corresponding source objects to the origin.

Admittedly, the above argument for our proposal relies on strong assumptions on data distributions (such as normality), which do not apply to real data. However, the effectiveness of our proposal is verified empirically in Section 4.6 by using real data.

4.5 Related Work

The first use of ridge regression in ZSL can be found in the work of Palatucci et al. [2009]. Ridge regression has since been one of the standard approaches to ZSL, especially for natural language processing tasks: phrase generation [Dinu and Baroni, 2014] and bilingual lexicon extraction [Dinu and Baroni, 2014, 2015; Mikolov et al., 2013a]. More recently, neural networks have been used for learning non-linear mapping [Frome et al., 2013; Socher et al., 2013]. All of the regression-based methods listed above, including those based on neural networks, map source objects into the target space.

ZSL can also be formulated as a problem of *canonical correlation analysis* (CCA). Hardoon et al. [2004] used CCA and kernelized CCA for image labeling. Lazaridou et al. [2014] compared ridge regression, CCA, singular value decomposition, and neural networks in image labeling. In our experiments (Section 4.6), we use CCA as one of the baseline methods for comparison.

Dinu and Baroni [2015] reported the hubness phenomenon in ZSL. They proposed two reweighting techniques to reduce hubness in ZSL, which are applicable to cosine similarity. Tomašev et al. [2013] proposed hubness-based instance weighting schemes for CCA. These schemes were applied to classification problems in which multiple instances (vectors) in the target space have the same class label. This setting is different from the one assumed in this chapter (see Section 4.2), i.e., we assume that a class label is represented by a single target vector.[2]

Structured output learning [Bakir et al., 2007] addresses a problem setting similar to ZSL, except that the target objects typically have complex structure, and thus the cost of embedding objects in a vector space is prohibitive. *Kernel dependency estimation* [Weston et al., 2002] is an approach that uses kernel PCA and regression to avoid this

[2] Perhaps because of this difference, the method in Tomašev et al. [2013] did not perform well in our experiment, and we do not report its result in Section 4.6.

issue. In this context, nearest neighbor search in the target space reduces to the *pre-image* problem [Mika et al., 1998] in the implicit space induced by kernels.

4.6 Experiments

We evaluated the proposed approach with both synthetic and real datasets. In particular, it was applied to two real ZSL tasks: bilingual lexicon extraction and image labeling.

The main objective of the following experiments is to verify whether our proposed approach is capable of suppressing hub formation and outperforming the existing approach, as claimed in Section 4.4.

4.6.1 Experimental Setups

Compared methods

The following methods were compared.

- $\text{Ridge}_{\text{X}\rightarrow\text{Y}}$: Linear ridge regression mapping source objects X into the space of target objects Y. This is how ridge regression was used in the existing work on ZSL [Dinu and Baroni, 2014, 2015; Lazaridou et al., 2014; Mikolov et al., 2013a; Palatucci et al., 2009].

- $\text{Ridge}_{\text{Y}\rightarrow\text{X}}$: Linear ridge regression mapping target objects Y into the source space. This is the proposed approach (Section 4.4.4).

- CCA: Canonical correlation analysis (CCA) for ZSL [Hardoon et al., 2004]. We used the code available from `http://www.davidroihardoon.com/Professional/Code.html`.

We calibrated the hyperparameters, i.e., the regularization parameter in ridge regression and the dimensionality of common feature space in CCA, by cross validation on the training set.

After ridge regression or CCA is applied, both X and Y (or their images) are located in the same space, wherein we find the closest target object for a given source object as measured by the Euclidean distance. In addition to the Euclidean distance, we also tested the *non-iterative contextual dissimilarity measure* (NICDM) [Jegou et al., 2007]

in combination with $\text{Ridge}_{\text{X}\to\text{Y}}$ and CCA. NICDM adjusts the Euclidean distance to make the neighborhood relations more symmetrical, and is known to effectively reduce hubness in non-ZSL context [Schnitzer et al., 2012].

All data were centered before application of regression and CCA, as usual with these methods.

Evaluation criteria

The compared methods were evaluated in two respects: (i) the correctness of their prediction, and (ii) the degree of hubness in nearest neighbor search.

Measures of Prediction Correctness. In all our experiments, ZSL was formulated as a ranking task; given a source object, all the target objects were ranked by their likelihood for the source object. As the main evaluation criterion, we used the mean average precision (MAP) [Manning et al., 2008], which is one of the standard performance metrics for ranking methods. Note that the synthetic and the image labeling experiments are the single-label problems for which MAP is equal to the mean reciprocal rank [Manning et al., 2008]. We also report the top-k accuracy[3] (Acc_k) for $k = 1$ and 10, which is the percentage of source objects for which the correct target objects are present in their k nearest neighbors.

Measure of Hubness. To measure the degree of hubness, we used the *skewness* of the (empirical) N_k distribution, following the approach in the literature [Radovanović et al., 2010a; Schnitzer et al., 2012; Suzuki et al., 2013; Tomašev et al., 2013]. The N_k distribution is the distribution of the number $N_k(i)$ of times each target object i is found in the top k of the ranking for source objects, and its skewness is defined as follows:

$$(N_k \text{ skewness}) = \frac{\sum_{i=1}^{\ell} \left(N_k(i) - \text{E}\left[N_k\right]\right)^3 / \ell}{\text{Var}\left[N_k\right]^{\frac{3}{2}}}$$

where ℓ is the total number of test objects in Y, $N_k(i)$ is the number of times the ith target object is in the top-k closest target objects of the source objects. A large N_k skewness value indicates the existence of target objects that frequently appear in the k-nearest neighbor lists of source objects; i.e., the emergence of hubs.

[3] In image labeling (only), we report the top-1 accuracy (Acc_1) *macro-averaged* over classes, to allow direct comparison with published results. Note also that Acc_k with a larger k would not be an informative metric for the image labeling task, which only has 10 test labels.

4.6.2 Task Descriptions and Datasets

We tested our method on the following ZSL tasks.

Synthetic task

To simulate a ZSL task, we need to generate object pairs across two spaces in a way that the configuration of objects is to some extent preserved across the spaces, but is not exactly identical. To this end, we first generated 3000-dimensional (column) vectors $\mathbf{z}_i \in \mathbb{R}^{3000}$ for $i = 1, \ldots, 10000$, whose coordinates were generated from an i.i.d. univariate standard normal distribution. Vectors $\mathbf{z}_i$ were treated as *latent* variables, in the sense that they were not directly observable, but only their images $\mathbf{x}_i$ and $\mathbf{y}_i$ in two different features spaces were. These images were obtained via different random projections, i.e., $\mathbf{x}_i = \mathbf{R}_X \mathbf{z}_i$ and $\mathbf{y}_i = \mathbf{R}_Y \mathbf{z}_i$, where $\mathbf{R}_X, \mathbf{R}_Y \in \mathbb{R}^{300 \times 3000}$ are random matrices whose elements were sampled from the uniform distribution over $[-1, 1]$. Because random projections preserve the length and the angle of vectors in the original space with high probability [Bingham and Mannila, 2001; Dasgupta, 2000], the configuration of the projected objects is expected to be similar (but different) across the two spaces.

Finally, we randomly divided object pairs $\{(\mathbf{x}_i, \mathbf{y}_i)\}_{i=1}^{10000}$ into the training set (8000 pairs) and the test set (remaining 2000 pairs).

Bilingual lexicon extraction

Our first real ZSL task is bilingual lexicon extraction [Dinu and Baroni, 2014, 2015; Mikolov et al., 2013b], formulated as a ranking task: Given a word in the source language, the goal is to rank its gold translation (the one listed in an existing bilingual lexicon as the translation of the source word) higher than other non-translation candidate words.

In this experiment, we evaluated the performance in two settings. One is the tasks of finding the English translations of words in the following source languages: Czech (cs), German (de), French (fr), Russian (ru), Japanese (ja), and Hindi (hi). The other setting is the finding the translations in six languages of English words.

Following related work [Dinu and Baroni, 2014, 2015; Mikolov et al., 2013a], we trained a CBOW model [Mikolov et al., 2013a] on the pre-processed Wikipedia corpus distributed by the Polyglot project[4] (see [Al-Rfou et al., 2013] for corpus statistics),

[4] https://sites.google.com/site/rmyeid/projects/polyglot

using the word2vec[5] tool. The window size parameter of word2vec was set to 10, with the dimensionality of feature vectors set to 500.

To learn the projection function and measure the accuracy in the test set, we used the bilingual dictionaries[6] of Ács et al. [2013] as the gold translation pairs. These gold pairs were randomly split into the training set (80% of the whole pairs) and the test set (20%). We repeated experiments on four different random splits, for which we report the average performance.

Image labeling

The second real task is image labeling, i.e., the task of finding a suitable word label for a given image. Thus, source objects X are the images and target objects Y are the word labels.

We used the Animal with Attributes (AwA) dataset[7], which consists of 30,475 images of 50 animal classes. For image representation, we used the DeCAF features [Donahue et al., 2013], which are the 4096-dimensional vectors constructed with convolutional neural networks (CNNs). DeCAF is also available from the AwA website. To save computational cost, we used random projection to reduce the dimensionality of DeCAF features to 500.

As with the bilingual lexicon extraction experiment, label features (word representations) were constructed with word2vec, but this time they were trained on the English version of Wikipedia (as of March 4, 2015) to cover all AwA labels. Except for the corpus, we used the same word2vec parameters as with bilingual lexicon extraction.

We respected the standard zero-shot setup on AwA provided with the dataset; i.e., the training set contained 40 labels, and test set contained the other 10 labels.

4.6.3 Experimental Results

[5] https://code.google.com/p/word2vec/

[6] http://hlt.sztaki.hu/resources/dict/bylangpair/wiktionary_2013july/

[7] http://attributes.kyb.tuebingen.mpg.de/

Table 4.1: Synthetic data results: MAP is the mean average precision. Acc_k is the accuracy of the k-nearest neighbor list. N_k is the skewness of the N_k distribution. A high N_k skewness indicates the emergence of hubs (lower is better).

method	MAP	Acc_1	Acc_{10}	N_1	N_{10}
$\text{Ridge}_{\text{X}\to\text{Y}}$	21.5	13.8	36.3	24.19	12.75
$\text{Ridge}_{\text{X}\to\text{Y}}$ + NICDM	58.2	47.6	78.4	13.71	7.94
$\text{Ridge}_{\text{Y}\to\text{X}}$ (proposed)	**91.7**	**87.6**	**98.3**	**0.46**	**1.18**
CCA	78.9	71.6	91.7	12.0	7.56
CCA + NICDM	87.6	82.3	96.5	0.96	2.58

Table 4.2: Mean-average precision on bilingual lexicon extraction.

(a) Source language to English (target language is English).

method	cs	de	fr	ru	ja	hi
$\text{Ridge}_{\text{X}\to\text{Y}}$	1.7	1.0	0.7	0.5	0.9	5.3
$\text{Ridge}_{\text{X}\to\text{Y}}$ + NICDM	11.3	7.1	5.9	3.8	10.2	21.4
$\text{Ridge}_{\text{Y}\to\text{X}}$ (proposed)	**40.8**	**30.3**	**46.5**	**31.1**	**42.0**	**40.6**
CCA	24.0	18.1	33.7	21.2	27.3	11.8
CCA + NICDM	30.1	23.4	39.7	26.7	35.3	19.3

(b) English to target language (English is source language).

method	cs	de	fr	ru	ja	hi
$\text{Ridge}_{\text{X}\to\text{Y}}$	2.1	1.4	0.9	0.8	0.9	6.1
$\text{Ridge}_{\text{X}\to\text{Y}}$ + NICDM	5.8	4.1	3.8	2.0	6.8	14.3
$\text{Ridge}_{\text{Y}\to\text{X}}$ (proposed)	**42.5**	**32.9**	**44.1**	**33.5**	**46.4**	**45.3**
CCA	18.2	13.4	32.9	19.6	28.6	7.3
CCA + NICDM	28.7	21.5	39.2	27.1	37.7	15.2

Table 4.3: Accuracy of the k-nearest neighbor list on bilingual lexicon extraction.

(a) Source language to English (target language is English).

	cs		de		fr		ru		ja		hi	
method	Acc_1	Acc_{10}	Acc_1	Acc_{10}	Acc_1	Acc_{10}	Acc_1	Acc_{10}	Acc_1	Acc_{10}	Acc_1	Acc_{10}
$\mathrm{Ridge}_{\mathrm{X}\to\mathrm{Y}}$	0.7	2.8	0.4	1.6	0.3	1.2	0.2	0.8	0.2	1.3	2.9	8.2
$\mathrm{Ridge}_{\mathrm{X}\to\mathrm{Y}}$ + NICDM	7.2	17.9	4.3	11.4	3.5	9.8	2.1	6.3	6.1	16.8	14.4	32.6
$\mathrm{Ridge}_{\mathrm{Y}\to\mathrm{X}}$ (proposed)	**31.5**	**54.5**	**21.6**	**43.0**	**36.6**	**58.6**	**21.9**	**43.6**	**31.9**	**56.3**	**31.1**	**55.4**
CCA	17.9	32.7	12.9	25.2	27.0	41.7	15.2	28.8	20.2	37.3	7.4	18.9
CCA + NICDM	21.9	42.3	16.1	33.9	31.1	50.1	18.7	37.0	25.9	48.8	12.4	30.7

(b) English to target language (source language is English).

	cs		de		fr		ru		ja		hi	
method	Acc_1	Acc_{10}	Acc_1	Acc_{10}	Acc_1	Acc_{10}	Acc_1	Acc_{10}	Acc_1	Acc_{10}	Acc_1	Acc_{10}
$\mathrm{Ridge}_{\mathrm{X}\to\mathrm{Y}}$	0.9	4.3	0.6	2.6	0.4	1.7	0.3	1.5	0.4	1.3	3.7	9.6
$\mathrm{Ridge}_{\mathrm{X}\to\mathrm{Y}}$ + NICDM	3.2	10.6	2.0	7.5	2.1	6.6	0.9	4.0	4.1	11.2	9.2	22.6
$\mathrm{Ridge}_{\mathrm{Y}\to\mathrm{X}}$ (proposed)	**31.1**	**59.8**	**22.7**	**47.6**	**33.9**	**57.8**	**22.7**	**47.6**	**34.4**	**61.7**	**33.2**	**62.4**
CCA	12.6	28.0	9.1	20.5	26.1	42.4	13.8	28.1	21.6	37.8	4.4	11.6
CCA + NICDM	19.8	43.6	14.5	32.5	30.1	51.7	18.3	39.5	27.3	51.8	9.3	24.2

Table 4.4: Skewness of N_k distribution on bilingual lexicon extraction. Smaller values are desirable.

(a) Source language to English (English is target language).

	cs		de		fr		ru		ja		hi	
method	N_1	N_{10}	N_1	N_{10}	N_1	N_{10}	N_1	N_{10}	N_1	N_{10}	N_1	N_{10}
$\text{Ridge}_{\text{X}\to\text{Y}}$	50.29	23.84	43.00	24.37	67.79	35.83	95.05	35.36	62.12	22.78	23.75	10.84
$\text{Ridge}_{\text{X}\to\text{Y}}$ + NICDM	41.56	20.38	39.32	20.82	57.18	25.97	89.08	30.70	57.57	21.62	20.33	9.21
$\text{Ridge}_{\text{Y}\to\text{X}}$ (proposed)	**11.91**	**10.74**	**12.49**	**11.94**	**2.56**	**2.77**	**4.28**	**4.18**	**5.15**	**6.76**	**10.45**	**6.14**
CCA	28.00	18.67	36.66	18.98	30.18	15.95	51.92	21.60	37.73	18.27	22.31	8.95
CCA + NICDM	25.00	17.13	32.94	17.65	25.20	14.65	42.61	20.72	34.66	13.16	22.00	8.46

(b) English to target language (English is source language).

	cs		de		fr		ru		ja		hi	
method	N_1	N_{10}	N_1	N_{10}	N_1	N_{10}	N_1	N_{10}	N_1	N_{10}	N_1	N_{10}
$\text{Ridge}_{\text{X}\to\text{Y}}$	68.40	29.93	45.62	23.89	71.41	36.76	65.62	32.52	65.14	24.99	11.95	8.20
$\text{Ridge}_{\text{X}\to\text{Y}}$ + NICDM	58.59	21.57	26.49	15.83	48.31	23.00	31.47	21.24	52.49	22.16	8.60	5.60
$\text{Ridge}_{\text{Y}\to\text{X}}$ (proposed)	**2.80**	**1.74**	**2.70**	**1.53**	**2.78**	**1.88**	**3.36**	**2.09**	**3.75**	**3.57**	**4.03**	**3.31**
CCA	9.52	6.91	7.36	7.40	7.55	8.28	5.80	6.10	9.96	13.18	5.42	5.10
CCA + NICDM	6.47	3.65	3.53	4.01	3.87	5.47	3.13	3.69	7.22	9.05	3.82	3.88

Table 4.5: Image labeling results. MAP, Acc_k, and N_k skewness.

method	MAP	Acc_1	N_1
$\mathrm{Ridge}_{\mathrm{X}\to\mathrm{Y}}$	46.0	22.6	2.61
$\mathrm{Ridge}_{\mathrm{X}\to\mathrm{Y}}$ + NICDM	54.2	34.5	2.17
$\mathrm{Ridge}_{\mathrm{Y}\to\mathrm{X}}$ (proposed)	**62.5**	**41.3**	**0.08**
CCA	26.1	9.2	2.00
CCA + NICDM	26.9	9.3	2.42

Tables 4.1 and 4.5 show the synthetic and image labeling results, respectively. The results of bilingual lexicon extraction are shown in Tables 4.2–4.4. The trends are fairly clear: The proposed approach, $\mathrm{Ridge}_{\mathrm{Y}\to\mathrm{X}}$, outperformed other methods in both MAP and Acc_k, over all tasks. $\mathrm{Ridge}_{\mathrm{X}\to\mathrm{Y}}$ and CCA combined with NICDM performed better than those with Euclidean distances, although they still lagged behind the proposed method $\mathrm{Ridge}_{\mathrm{Y}\to\mathrm{X}}$ even with NICDM.

The N_k skewness achieved by $\mathrm{Ridge}_{\mathrm{Y}\to\mathrm{X}}$ was lower (i.e., better) than that of compared methods, meaning that it effectively suppressed the emergence of hub labels. In contrast, $\mathrm{Ridge}_{\mathrm{X}\to\mathrm{Y}}$ produced a high skewness which was in line with its poor prediction accuracy. These results support the expectation we expressed in the discussion in Section 4.4.

The results presented in the tables show that the degree of hubness (N_k) for all tested methods inversely correlates with the correctness of the output rankings, which strongly suggests that hubness is one major factor affecting the prediction accuracy.

From Tables 4.2–4.4, in both retrieval direction; English as X and six languages as Y, and English as X and six languages as Y, the trends are almost identical. The proposed approach, $\mathrm{Ridge}_{\mathrm{Y}\to\mathrm{X}}$ outperformed other methods. It also had the lowest N_k skewness. This observation suggests that not the quality of feature space but the proposed approach leads to the improvement.

For the AwA image dataset, Akata et al. [2014, the fourth row (CNN) and second column (φ^w) of Table 2] reported a 39.7% Acc_1 score, using image representations trained with CNNs, and 100-dimensional word representations trained with word2vec. For comparison, our proposed approach, $\mathrm{Ridge}_{\mathrm{Y}\to\mathrm{X}}$, was evaluated in a similar setting: We used the DeCAF features (which were also trained with CNNs) without random projection as the image representation, and 100-dimensional word2vec word vectors. In this setup, $\mathrm{Ridge}_{\mathrm{Y}\to\mathrm{X}}$ achieved a 40.0% Acc_1 score. Although the experimental se-

tups are not exactly identical and thus the results are not directly comparable, this suggests that even linear ridge regression can potentially perform as well as more recent methods, such as Akata et al.'s, simply by exchanging the observation and response variables.

4.6.4 Discussion

Shrinking property

Proposition 2 states that in ridge regression, the 2-norm of mapped input data matrix (**MA**) is smaller than that of the response **B**. The result also holds with ordinary least squares, i.e., when regularization parameter $\lambda = 0$.

On the basis of this proposition, we claimed that ridge regression has a tendency to place mapped inputs **MA** closer than the response **B**, to the origin. However, it does not say how much it actually shrinks the mapped input data relative to the response, nor how much parameter λ influences the shrinkage.

Table 4.6a shows the average ratio of the norm of the mapped input to that of its response: i.e., averaged over all training pairs of the mapped input **Ma** and its response **b**. The table shows that the norm of the mapped inputs is, in most cases, less than half that of the responses. Moreover, the trend is the same regardless of the values of λ.

Preposition 2 is concerned only with the training set, but we claimed that the same tendency is also likely for the test set; that is, the input data in the test set should also be mapped closer to the origin than their corresponding responses.

The empirical results shown in Table 4.6b support this claim. This table shows the average ratio of the norm of mapped inputs to that of the corresponding response, which is the same as Table 4.6a, except that the ratios are computed for the test set. By comparing the two tables, we see that the amount of shrinkage occurs at a nearly identical level across the training and test sets.

Table 4.6: The average ratio of $\|\mathbf{Ma}\|$ to $\|\mathbf{b}\|$.

(a) The training set.

Method for computing $\mathbf{M}$	Synthetic data	Bilingual lexicon extraction						Image labeling
		cs	de	fr	ru	ja	hi	
$\text{Ridge}_{\text{X}\to\text{Y}}(\lambda = 1)$	0.37	0.44	0.40	0.42	0.38	0.48	0.61	0.49
$\text{Ridge}_{\text{Y}\to\text{X}}(\lambda = 1)$	0.37	0.49	0.43	0.42	0.39	0.48	0.70	0.32
$\text{Ridge}_{\text{X}\to\text{Y}}(\lambda = 100)$	0.37	0.43	0.39	0.42	0.37	0.46	0.51	0.49
$\text{Ridge}_{\text{Y}\to\text{X}}(\lambda = 100)$	0.37	0.48	0.42	0.42	0.39	0.47	0.61	0.32

(b) The test set.

Method for computing $\mathbf{M}$	Synthetic data	Bilingual lexicon extraction						Image labeling
		cs	de	fr	ru	ja	hi	
$\text{Ridge}_{\text{X}\to\text{Y}}(\lambda = 1)$	0.37	0.45	0.40	0.42	0.38	0.48	0.68	0.47
$\text{Ridge}_{\text{Y}\to\text{X}}(\lambda = 1)$	0.37	0.50	0.43	0.42	0.39	0.49	0.79	0.19
$\text{Ridge}_{\text{X}\to\text{Y}}(\lambda = 100)$	0.37	0.44	0.39	0.42	0.37	0.47	0.53	0.46
$\text{Ridge}_{\text{Y}\to\text{X}}(\lambda = 100)$	0.37	0.49	0.43	0.42	0.39	0.48	0.65	0.19

The actual radius in Proposition 3

In Proposition 3, points $\mathbf{x}$ (to be precise, $\mathbf{x}_1$ and $\mathbf{x}_2$) and $\mathbf{y}$ signify a pair of a mapped input ($\mathbf{x}_i$) and its response ($\mathbf{y}$), so it appears that, as a result of regression, these points must be quite close. i.e., the distance r between $\mathbf{x}$ and $\mathbf{y}$ must be small. If this is the case, the volume of a ball with radius r should be small, and there should not be much difference in the probability mass in the balls around $\mathbf{x}_1$ and $\mathbf{x}_2$ discussed in the propositions.

In reality, r is not at all small—so the difference in probability mass can be large between the respective balls.

Table 4.7 shows the ratios of the average distance between corresponding pairs in the dataset to the distance between all possible pairs of $\mathbf{x}$ and $\mathbf{y}$, majority of which are non-corresponding pairs.

In this table, all the ratios are in the range of 0.7 to 1.00, which implies that the distance between corresponding pairs is not substantially smaller than the distance between randomly chosen pairs. Therefore, r, which signifies the distance between the corresponding pairs in Proposition 3, can in fact be quite large.

Table 4.7: The ratio of the average distance between corresponding pairs to the average over all pairs.

(a) The training set.

Method for computing **M**	Synthetic data	Bilingual lexicon extraction						Image labeling
		cs	de	fr	ru	ja	hi	
$\text{Ridge}_{\text{X}\to\text{Y}}(\lambda = 1)$	0.87	0.80	0.84	0.83	0.86	0.79	0.68	0.97
$\text{Ridge}_{\text{Y}\to\text{X}}(\lambda = 1)$	0.87	0.84	0.86	0.85	0.90	0.80	0.68	1.00
$\text{Ridge}_{\text{X}\to\text{Y}}(\lambda = 100)$	0.87	0.80	0.84	0.83	0.86	0.80	0.72	0.97
$\text{Ridge}_{\text{Y}\to\text{X}}(\lambda = 100)$	0.87	0.85	0.87	0.85	0.90	0.81	0.71	1.00

(b) The test set.

Method for computing **M**	Synthetic data	Bilingual lexicon extraction						Image labeling
		cs	de	fr	ru	ja	hi	
$\text{Ridge}_{\text{X}\to\text{Y}}(\lambda = 1)$	0.91	0.84	0.87	0.85	0.87	0.84	0.85	0.99
$\text{Ridge}_{\text{Y}\to\text{X}}(\lambda = 1)$	0.91	0.88	0.90	0.86	0.91	0.85	0.86	0.99
$\text{Ridge}_{\text{X}\to\text{Y}}(\lambda = 100)$	0.91	0.85	0.88	0.85	0.87	0.84	0.85	0.99
$\text{Ridge}_{\text{Y}\to\text{X}}(\lambda = 100)$	0.91	0.88	0.90	0.86	0.91	0.85	0.86	0.99

4.7 Summary

This chapter has presented our formulation of ZSL as a regression problem of finding a mapping from the target space to the source space, which opposes the way in which regression has been applied to ZSL to date. Assuming a simple model in which data follows a multivariate normal distribution, we provided an explanation as to why the proposed direction is preferable, in terms of the emergence of hubs in the subsequent nearest neighbor search step. The experimental results showed that the proposed approach outperforms the existing regression-based and CCA-based approaches to ZSL.

Future research topics include: (i) extending the analysis of Section 4.4 to cover multi-modal data distributions, or other similarity/distance measures such as cosine; (ii) investigating the influence of mapping directions in other regression-based ZSL methods, including neural networks; and (iii) investigating the emergence of hubs in CCA.

Chapter 5

Reducing Hubness for k-Nearest Neighbor Classification

5.1 Introduction

k-nearest neighbor (k-NN) classifiers predict the class label of an unknown object by its nearest neighbors. Given an unlabeled (test) object $\mathbf{x}$ and a set of labeled (training) objects $\mathcal{D} = \{(\mathbf{x}_i, y_i)\}_{i=1}^n$ where $\mathbf{x}, \mathbf{x}_i \in \mathcal{X}$ are the feature vectors and $y_i \in \mathcal{Y}$ is the class label of $\mathbf{x}_i$, the classifier first computes the distance between $\mathbf{x}$ and each labeled object $\mathbf{x}_i$. Then it predicts the class label $\hat{y}$ of $\mathbf{x}$ by the majority among its k nearest labeled objects. When $k = 1$, the decision rule of the k-NN (1-NN) classifier is simply:

$$\hat{y} = \underset{y_i:(\mathbf{x}_i, y_i)\in\mathcal{D}}{\arg\min}\ f(\mathbf{x}, \mathbf{x}_i), \tag{5.1}$$

where function $f : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ is some distance/dissimilarity function.

Obviously, the choice of function f affects the accuracy of k-NN classification. Therefore, many researchers [Davis et al., 2007; Weinberger and Saul, 2009; Xing et al., 2002; Ying and Li, 2012] have tackled *metric learning*, which is the task of learning a suitable distance function from data.

For Euclidean object space $\mathcal{X} = \mathbb{R}^d$, metric learning is usually formulated as the task of finding Mahalanobis distance. In this formulation, the squared distance between two objects $\mathbf{x}, \mathbf{z} \in \mathbb{R}^d$ is defined by

$$f(\mathbf{x}, \mathbf{z}) = (\mathbf{x} - \mathbf{z})^{\mathrm{T}} \mathbf{M} (\mathbf{x} - \mathbf{z}), \tag{5.2}$$

with some positive (semi)definite matrix $\mathbf{M}$. By defining matrix $\mathbf{L}$ by $\mathbf{M} = \mathbf{L}^{\mathrm{T}}\mathbf{L}$, we can write the squared Mahalanobis distance in Eq. (5.2) as

$$f(\mathbf{x}, \mathbf{z}) = \|\mathbf{L}\mathbf{x} - \mathbf{L}\mathbf{z}\|^2. \tag{5.3}$$

This equation shows that learning Mahalanobis distance is equivalent to learning a suitable linear transformation $\mathbf{L}$.

In the context of k-NN classification, distance needs to be measured only between unlabeled (test) objects and labeled (training) objects, as can be seen from Eq. (5.1); when distance $f(\mathbf{x}, \mathbf{z})$ is computed, the first object $\mathbf{x}$ is always an unlabeled object, and the second object $\mathbf{z}$ is always a labeled object $\mathbf{x}_i$. Moreover, function f need not be metric and can be any measure of dissimilarity; for instance, f being noncommutative is perfectly acceptable.

In this chapter, we learn one such dissimilarity function. The idea is to compute a transformation of labeled objects to new points while test objects are kept at their original points. Thus, our objective is to find a suitable matrix $\mathbf{W}$ that defines a dissimilarity function

$$f(\mathbf{x}, \mathbf{z}) = \|\mathbf{x} - \mathbf{W}\mathbf{z}\|^2, \tag{5.4}$$

where $\mathbf{x}$ is a test object, and $\mathbf{z}$ is a labeled object.

Because the coordinates of test data are fixed, our formulation might appear less flexible than Mahalanobis distance learning (Eq. (5.3)). However, as shown in a subsequent section, it gives a better k-NN classification accuracy than Mahalanobis distance learning methods on many datasets that feature high-dimensional space. Moreover, optimizing $\mathbf{W}$ in Eq. (5.4) is much easier and substantially (often more than two orders of magnitude) faster.

The effectiveness of the proposed approach has a theoretical foundation in terms of reduction of *hubness* in data [Radovanović et al., 2010a]. Recent studies have shown that the presence of hubs, which are a few objects that appear in the k-NNs of many objects, is an obstacle that can harm the performance of many vector space methods [Radovanović et al., 2010a; Schnitzer et al., 2012; Suzuki et al., 2013]. We show that metric learning is no exception, and transformation of labeled objects suppresses the emergence of hubs, as justified by our results in Chapter 4, which used regression to reduce hubness in zero-shot problems. In Chapter 4, the problem was cast as a task of cross-domain matching, whereas in this chapter, we are concerned with improving the accuracy of k-NN classification in a single space.

Another notable feature of the proposed method is that it does away with optimization over "negative" object pairs, i.e., objects belonging to difference classes. In other words, our method optimizes only over "positive" object pairs, i.e., object of the same classes that should be made closer after transformation, and does not have any constraints or terms in the objective function that attempt to keep negative object pairs apart from each other. Such constraints are indispensable in Mahalanobis metric learning to prevent trivial solutions $\mathbf{M} = \mathbf{O}$ or $\mathbf{L} = \mathbf{O}$ in Eqs. (5.2) or (5.3), and metric learning typically optimizes over a large number of negative object pairs. Moreover, incorporating negative pairs results in a non-convex optimization problem with respect to matrix $\mathbf{L}$. Existing metric learning methods [Davis et al., 2007; Jain et al., 2012; Weinberger and Saul, 2009; Xing et al., 2002; Ying and Li, 2012] hence resorts to optimizing $\mathbf{M} = \mathbf{L}^{\mathrm{T}}\mathbf{L}$ using computationally intensive methods such as semi-definite programming. By contrast, since we only transforms labeled objects, we need not worry about $\mathbf{W} = \mathbf{O}$ being the solution (see Eq. (5.4)), thus eliminating the need of negative pairs. This makes the solution easily obtained with ridge regression, which contributes to reduced computation time.

5.2 Related Work

We briefly review some of the metric learning methods, mostly those used in the experiments in Section 5.5. For comprehensive survey of the field, see [Bellet et al., 2014; Kulis, 2013].

A majority of the metric learning methods adopt Mahalanobis distance (Eq. (5.3)) as the distance function, and minimize the training loss under various constraints. As mentioned earlier, these methods do not make distinction between test (unlabeled) objects and training (labeled) objects, in the sense that their coordinates are transformed by the same matrix, $\mathbf{L}$ in Equation (5.3). Our approach differs from these methods in that it projects only the labeled objects to new coordinates.

There are various strategies for learning Mahalanobis distance. Xing et al. [2002] formulated metric learning as a convex optimization problem, and demonstrated its effectiveness in clustering tasks. The *large-margin nearest neighbor* (LMNN) method [Weinberger and Saul, 2009] is probably the most popular of all metric learning methods. Its objective is to minimize distances between objects with the same label, and to penalize objects with different labels when they are closer than a certain distance. Hence objects from different classes are separated by a large margin. To make the

problem convex, in Xing et al.'s method and LMNN, optimization is done over not $\mathbf{L}$ but $\mathbf{M} = \mathbf{L}^{\mathrm{T}}\mathbf{L}$, with semidefinite programming. Ying and Li [2012] presented an eigenvalue optimization framework for learning Mahalanobis distance. Davis et al. [2007] proposed *information-theoretic metric learning* (ITML). ITML minimizes the LogDet divergence subject to linear constraints. It thus requires no eigenvalue computation or semi-definite programming.

Although it has been shown that these methods work well in many applications, learning Mahalanobis distance typically incurs high computational cost. Indeed, as we show in an experiment (Section 5.5), these methods spend substantial time in optimizing $\mathbf{M}$, when applied to large datasets.

5.3 Proposed Method

In this section, we present our approach for improving the k-NN classification accuracy.

In nearly all metric learning methods, the objective function to be optimized involves a term that encourages objects of the same class to be placed closer. In the same vein, our method also optimizes the transformation matrix $\mathbf{W}$ in Eq. (5.4) by minimizing the distance between objects of the same class. However, in our formulation, the learned transformation $\mathbf{W}$ is only applied to labeled objects.

Our training procedure consists of two steps. We first make training object pairs for which the distance should be minimized. To this end, we follow Weinberger and Saul [2009]: for each labeled object $\mathbf{x}_i \in \mathbb{R}^d$ in the training set, we define its "target" objects $\mathcal{T}_i$ to be the k objects in the training set that belong to the same class as $\mathbf{x}_i$ and are closest to $\mathbf{x}_i$ as measured by the original Euclidean distance (i.e., the one before training). We then find a matrix $\mathbf{W} \in \mathbb{R}^{d \times d}$ that moves objects in $\mathcal{T}_i$ towards $\mathbf{x}_i$, by solving the following optimization problem:

$$\min_{\mathbf{W}} \sum_{i=1}^{n} \sum_{\mathbf{z} \in \mathcal{T}_i} \|\mathbf{x}_i - \mathbf{W}\mathbf{z}\|^2 + \lambda \|\mathbf{W}\|_{\mathrm{F}}^2, \tag{5.5}$$

where $\lambda \geq 0$ is a hyperparameter for regularization and $\|\cdot\|_{\mathrm{F}}$ represents the Frobenius norm. Equation (5.5) is a familiar objective function of ridge regression, and we have the closed-form solution:

$$\mathbf{W} = \mathbf{X}\mathbf{J}\mathbf{X}^{\mathrm{T}}(\mathbf{X}\mathbf{X}^{\mathrm{T}} + \lambda \mathbf{I})^{-1}, \tag{5.6}$$

where $\mathbf{X} = [\mathbf{x}_1, \ldots, \mathbf{x}_n] \in \mathbb{R}^{d \times n}$, and $\mathbf{J} \in \{0,1\}^{n \times n}$ is an indicator matrix such that $[\mathbf{J}]_{i,j} = 1$ if $\mathbf{x}_j \in \mathcal{T}_i$ and 0 otherwise.

In the test phase, we first compute the image $\mathbf{x}'_i = \mathbf{W}\mathbf{x}_i$ of every labeled object $\mathbf{x}_i$ by the learned matrix $\mathbf{W}$. We then carry out k-NN classification by regarding $\mathcal{D}' = \{(\mathbf{x}'_i, y_i)\}$ as the labeled objects in place of the original one, $\mathcal{D} = \{(\mathbf{x}_i, y_i)\}$. In the case of 1-NN classification, for example, this amounts to using the dissimilarity function f given by Eq. (5.4) in the decision rule of Eq. (5.1), i.e.,

$$\hat{y} = \operatorname*{arg\,min}_{y_i:(\mathbf{x}'_i, y_i) \in \mathcal{D}'} \|\mathbf{x} - \mathbf{x}'_i\|^2 = \operatorname*{arg\,min}_{y_i:(\mathbf{x}_i, y_i) \in \mathcal{D}} \|\mathbf{x} - \mathbf{W}\mathbf{x}_i\|^2. \tag{5.7}$$

5.4 Proposed Method Reduces Hubness

In this section, we argue that the proposed method is by design less susceptible to producing hubs [Radovanović et al., 2010a] in the transformed labeled objects. This property is desirable, as hubs have been recognized as one of the major factors that harm the performance of nearest neighbor methods.

5.4.1 Hubness and the Proposed Method

Ridge regression reduces the variance of mapped feature values (observables) relative to that of target (response) variables; see Proposition 2 in Section 4.4.1. Thus, in our model of Eq. (5.5), the variance of the components of the mapped objects $\mathbf{W}\mathbf{z}$ tends to be smaller than that of $\mathbf{x}$. From the discussion on hubness in Section 4.3, reducing the variance of data objects (which correspond to the image $\mathbf{W}\mathbf{z}$ of the labeled objects $\mathbf{z}$ in the proposed method) relative to the query (test object $\mathbf{x}$) can reduce the spatial centrality. By combining these arguments, we expect that the proposed approach should alleviate the emergence of hubs, and, consequently, improve the accuracy of k-NN classification.

Note that we could think of a different regression problem in which test object $\mathbf{x}$, not labeled object $\mathbf{z}$, is mapped to new coordinates:

$$\min_{\mathbf{W}} \sum_{i=1}^{n} \sum_{\mathbf{z} \in \mathcal{T}_i} \|\mathbf{W}\mathbf{x}_i - \mathbf{z}\|^2 + \lambda \|\mathbf{W}\|_{\mathrm{F}}^2. \tag{5.8}$$

This would result in function f as follows:

$$f(\mathbf{x}, \mathbf{z}) = \|\mathbf{W}\mathbf{x} - \mathbf{z}\|^2. \tag{5.9}$$

However, this dissimilarity function is useless as it actually *promotes* hubness. The variance of the transformed test objects shrinks as a result of regression. Thus, in this model, the variance of the labeled objects is made larger than the transformed test objects, but this is not a desirable situation according to Proposition 1. We also verify this in one of the experiments in Section 5.5.

5.5 Experiments

We evaluate the proposed approach on various classification tasks. The objective of these experiments is to investigate whether the proposed approach can reduce the emergence of hubs, and improve the performance of k-NN classification. The performance is measured against several popular metric learning methods.

5.5.1 Experimental Setups

Dataset description

Three types of datasets were used for our evaluation: UCI, document, and image datasets.

From the UCI machine learning datasets,[1] we chose balance-scale, glass, ionosphere, iris, and wine, as they are frequently used for evaluation in metric learning literature [Davis et al., 2007; Jain et al., 2012; Weinberger and Saul, 2009; Ying and Li, 2012]. However, they are mostly toy problems, and their small feature dimensions, the numbers of labels and objects do not necessarily reflect real-world problems. We therefore used document and image datasets also for our evaluation.

For document and image classification, support vector machines are known to provide state-of-the-art accuracy. Notice, however, that our goal is not to design a state-of-the-art classifier. Rather, the main objective of this experiments is to evaluate the performances of the proposed method in comparison with metric learning methods, and to show its usefulness for k-NN classification.

For document classification tasks, we used four publicly available document datasets: RCV1-v2 (RCV), 20 newsgroups (News), Reuters21578 (Reuters), and TDT2 (TDT).[2]

[1] http://archive.ics.uci.edu/ml/

[2] Datasets were downloaded from http://www.cad.zju.edu.cn/home/dengcai/Data/TextData.html

Table 5.1: Dataset statistics. In document and image datasets, "original dim." indicates the number of raw dimensions before applying PCA.

(a) UCI datasets.

dataset	ionoshpere	balance-scale	iris	wine	glass
#objects	351	625	150	178	214
#classes	2	3	3	3	6
dimension	34	4	4	13	9

(b) Document datasets.

dataset	RCV	News	Reuters	TDT
#objects	9625	18846	8213	10021
#classes	4	20	41	56
dimension	300	300	300	300
original dim.	29992	26214	18933	36771

(c) Image datasets.

dataset	AwA	CUB	SUN	aPY
#objects	30475	11788	14340	15339
#classes	50	200	717	32
dimension	300	300	300	300
original dim.	4096	4096	4096	4096

In Reuters21578 and TDT2, we removed minority classes that hold less than 10 objects in the dataset. After this removal, Reuters21578 and TDT2 had 56 and 41 classes, respectively.

For image classification, we used the following image datasets: aPascal & aYahoo (aPY), Animals with Attributes (AwA), Caltech-UCSD Birds-200-2011 (CUB), and SUN Attribute.[3]

The computational cost of metric learning methods is heavily dependent on the dimension of the feature space. In our preliminary experiment, training of the metric

[3]We used the publicly available features from https://zimingzhang.files.wordpress.com/2014/10/cnn-features1.key

learning methods (LMNN, ITML, and DML-eig; see below) did not complete in a reasonable time on document and image datasets. We therefore had to use principal component analysis to reduce the dimensionality of features to 300 for these datasets.

The dataset statistics are summarized in Table 5.1.

All data (set of feature vectors) were centered before training. For the wine dataset, we further converted the features to z-scores, following the remark on the UCI website that a k-NN classifier achieved a high accuracy with this standardization.

Each dataset was randomly split into training (70%) and test (30%) sets. Experiments were repeated on four different random splits, for which we report the average performance.

Compared methods

We trained distance/dissimilarity functions using the following methods, and carried out k-NN classification on the above datasets.

- original metric: Euclidean distance in the original object space, without any training. This is the baseline.
- LMNN: Large margin nearest neighbor classification [Weinberger and Saul, 2009]. This method was often used in distance metric learning experiments as a baseline.
- ITML: Information theoretic metric learning [Davis et al., 2007].
- DML-eig: Distance metric learning with eigenvalue optimization [Ying and Li, 2012].
- proposed method: This is the proposed approach that optimizes Eq. (5.5), and then predicts the label with Eq. (5.7).

LMNN, ITML, and DML-eig learn a Mahalanobis distance. For these methods, we used the publicly available MATLAB implementations provided by the respective authors[4]. We implemented the proposed method also in MATLAB[5], for fair evaluation of running time.

[4]LMNN: https://bitbucket.org/mlcircus/lmnn/downloads,
ITML: http://www.cs.utexas.edu/~pjain/itml/,
DML-eig: http://www.albany.edu/~yy298919/software.html

[5]This code will be made available at our homepage.

To discuss the behavior of hubness, we also evaluated the following method:

- map-test: Mapping test objects. This is the method mentioned in Section 5.4.1 that optimizes Eq. (5.8). Then, the resulting dissimilarity function of Eq. (5.9) is used for k-NN classification.

Notice again that this method was tested only to verify the claim made in Section 5.4.1, i.e., although both the proposed method and map-test are based on ridge regression, the proposed method is expected to perform well by reducing hubness, whereas map-test is expected to do the contrary.

For LMNN, the proposed method, and map-test, the number of target objects for each training object was set to 1; i.e., for each object $\mathbf{x}_i$ in the training set, we made a training pair $(\mathbf{x}_i, \mathbf{z})$ whose distance should be minimized, where $\mathbf{z}$ is the object nearest to $\mathbf{x}_i$ among those with the same class label as $\mathbf{x}_i$ in the training set, with the distance measured by the original Euclidean metric. For the parameters of ITML on UCI datasets, we used the default values in the authors' implementation, and for document and image datasets, we followed Jain et al. [2012]. For DML-eig, we used the default setting in the authors' code to obtain pairwise constraints. We calibrated the parameter k of k-NN classification to be used at the test time and all other parameters (γ in ITML, μ in DML-eig, and λ in the proposed method and test-map) by cross validation on the training set.

Evaluation criteria

The methods were evaluated in three respects: (i) the accuracy of k-NN classification using the distance/dissimilarity measure learned by each method, (ii) training time, and (iii) the degree of hubness in the data with respect to the learned distance/dissimilarity.

Following the literature [Hara et al., 2015; Radovanović et al., 2010a; Schnitzer et al., 2012; Suzuki et al., 2013], we used the skewness of N_{10} distribution as the measure of hubness in the data. The N_{10} distribution is the empirical distribution of the number $N_{10}(i)$ of times each labeled object i is found in the 10-nearest neighbors of test objects, and its skewness is defined as follows:

$$(N_{10}\text{ skewness}) = \frac{\sum_{i=1}^{n} \left(N_{10}(i) - \mathrm{E}\left[N_{10}\right]\right)^3 / n}{\mathrm{Var}\left[N_{10}\right]^{\frac{3}{2}}}$$

where n is the total number of labeled objects, and $\mathrm{E}[N_{10}]$ and $\mathrm{Var}[N_{10}]$ are respectively the empirical mean and variance of $N_{10}(i)$ over n labeled objects. A large N_{10} skewness

value indicates the existence of labeled objects that frequently appear in the 10-nearest neighbor lists of test objects, i.e., hubs.

5.5.2 Experimental Results and Discussion

Skewness

Table 5.2 shows the skewness of N_{10} distribution. For all datasets, we observe that the proposed approach reduced N_{10} skewness considerably compared with the original Euclidean distance, meaning that it effectively suppressed the emergence of hub objects. N_{10} skewness was reduced by metric learning methods (LMNN, ITML, and DML-eig) on many datasets, most notably by DML-eig. Also, as expected from the discussion of Section 5.4.1, mapping test objects (map-test in the tables) increased N_{10} skewness except for the iris dataset.

Accuracy

Tables 5.3 shows the classification accuracy. In most datasets, both the metric learning methods and the proposed method outperformed the original distance metric. The proposed method is comparable with, or slightly better than, the metric learning methods. Although map-test optimized the minimizing distance between objects in same class (our proposed method also optimized such distance), the method obtained poor results even compared with the original Euclidean metric except for the iris datasets.

Note that, in UCI datasets, we observed that the proposed method did not work well, and even map-test were competitive with others. This is an expected result, because the UCI datasets did not have much hubness even with the original metric (see Table 5.2a). Hubs tend to be emerge in high dimensional space [Hara et al., 2015; Radovanović et al., 2010a; Schnitzer et al., 2012], but all the UCI datasets have a small dimensionality (see Table 5.1a). Consequently, hub reduction/promotion methods did not affect the result significantly.

Training time

To investigate the computational cost, we measured the elapsed real time needed to train the proposed method and the metric learning methods.

Table 5.4 shows the average training time in document and image datasets. We observe that the proposed approach has a clear advantage in terms of training cost. It was faster than any metric learning methods compared. Indeed, on all datasets except RCV, it was more than two orders of magnitude faster than the fastest metric learning methods. This can be explained by the fact that the metric learning methods take burden of optimizing over Mahalanobis metric. To enforce the constraint that the matrix $\mathbf{M}$ in Eq. (5.2) should remain positive semi-definite, these methods pay high computational cost, e.g., to check the non-negativity of eigenvalues, at every training iteration. In contrast, the proposed approach has a closed-form solution, and hence it depends on computing matrix inverse, but this computation needs to be done only once.

5.6 Summary

In this chapter, we have proposed a simple regression-based technique to improve k-NN classification accuracy.

The results of our work can be summarized as follows:

- To improve the accuracy of k-NN classification, we proposed learning a transformation of labeled objects only, without altering the coordinates of test (unlabeled) objects (Section 5.3). This approach optimizes the transformation such that the distance between objects that belong to the same class is minimized.

  At first sight, not moving test objects might seem loss of flexibility and expressiveness of the learned transformation. However, this approach is justified from the perspective of reducing hubness in the labeled objects. Because our method is inherently designed to suppress hubness, it need not consider pairs of objects from different classes during training. The number of such object pairs can be huge and their use also renders the optimization problem non-convex, which is thus a major obstacle to the scalability of metric learning methods.

- In our experiments (Section 5.5), we demonstrated empirically that after the labeled objects were transformed with the proposed approach, k-NN classification accuracy was improved. Specifically, our approach reduced the emergence of hubs, and improved the classification accuracy accordingly. The proposed method showed better k-NN classification accuracy than the metric learning methods on most document and image datasets, and comparable on the rest. We also evaluated the proposed method on the UCI datasets which are frequently

used as the benchmark for metric learning. Because of the low dimensionality of the datasets, the hubness effect was not evident. As a result, the effectiveness of the proposed method was not observed on these datasets.

- The experiments showed that our approach was substantially faster than the compared metric learning methods. For large document and image datasets, the speed-up was more than two orders of magnitude over the fastest metric learning methods, although the classification accuracy was better or comparable.

We have focused on multi-class classification problems in this chapter, but hubness is known to be harmful in other situations, such as clustering and semi-supervised classification in high-dimensional space [Radovanović et al., 2010a]. We plan to extend our approach to deal with such situations. We will also extend our method to learn nonlinear metrics.

Another direction of future work is to investigate the effect of our approach on kernel machines. Metric learning has been shown to be an effective preprocessing for kernel machines [Dhillon et al., 2010; Weinberger and Tesauro, 2007; Xu et al., 2013], and we will pursue a similar line using our approach.

Table 5.2: Skewness of N_{10} distribution: A high skewness indicates the emergence of hubs (smaller is better). The bold figure indicates the best performer.

(a) UCI datasets.

method	ionosphere	balance-scale	iris	wine	glass
original metric	1.65	0.93	0.40	0.71	0.77
LMNN	1.05	0.63	0.39	0.61	0.74
ITML	0.96	0.79	**0.10**	0.43	0.70
DML-eig	**0.78**	0.66	0.41	**0.38**	**0.59**
proposed method	1.04	**0.56**	0.32	0.55	**0.59**
map-test	1.67	1.13	0.32	0.89	1.18

(b) Document datasets.

method	RCV	News	Reuters	TDT
original metric	13.35	21.93	7.61	4.89
LMNN	3.86	14.74	7.63	4.01
ITML	4.27	19.65	7.30	2.39
DML-eig	1.71	**1.45**	**3.05**	**1.34**
proposed method	**1.14**	2.88	4.53	1.44
map-test	21.57	33.36	17.49	6.71

(c) Image datasets.

method	AwA	CUB	SUN	aPY
original metric	2.49	2.38	2.52	2.80
LMNN	3.10	2.96	2.80	3.94
ITML	2.42	2.27	2.37	2.69
DML-eig	1.90	1.77	2.39	2.17
proposed method	**1.24**	**0.97**	**1.02**	**1.23**
map-test	7.81	7.83	7.48	11.65

Table 5.3: Classification accuracy [%]: Bold figures indicate the best performers for each dataset.

(a) UCI datasets.

method	ionosphere	balance-scale	iris	wine	glass
original metric	86.8	89.5	97.2	98.1	68.1
LMNN	**90.3**	90.0	96.7	98.1	67.7
ITML	87.7	89.5	**97.8**	**99.1**	65.0
DML-eig	87.7	**91.2**	96.7	98.6	66.5
proposed method	89.6	89.5	97.2	98.6	**70.8**
map-test	79.7	89.4	97.2	96.3	62.3

(b) Document datasets.

method	RCV	News	Reuters	TDT
original metric	92.1	76.9	89.5	96.1
LMNN	**94.7**	79.9	91.5	96.6
ITML	93.2	77.0	90.8	96.5
DML-eig	94.5	73.3	85.9	95.7
proposed method	94.4	**81.6**	**91.6**	**96.7**
map-test	89.1	70.0	85.9	95.4

(c) Image datasets.

method	AwA	CUB	SUN	aPY
original metric	83.2	51.6	26.2	82.2
LMNN	83.0	**54.7**	24.4	81.8
ITML	83.1	51.3	26.0	82.4
DML-eig	82.0	53.5	22.4	81.6
proposed method	**84.1**	52.4	**28.3**	**83.4**
map-test	79.2	43.3	14.6	78.7

Table 5.4: Training time [sec]: Bold figures indicate the best performer for each dataset.

(a) Document datasets.

method	RCV	News	Reuters	TDT
LMNN	1713.0	1164.7	676.2	886.1
ITML	35.5	1512.5	124.1	169.0
DML-eig	762.2	6145.9	2710.4	2350.6
proposed	**6.0**	**7.0**	**4.6**	**16.1**

(b) Image datasets.

method	AwA	CUB	SUN	aPY
LMNN	1525.5	1098.2	15704.3	317.3
ITML	1536.3	577.6	1126.4	9211.2
MDL-eig	2048.0	2084.7	2006.1	1787.1
proposed	**9.5**	**1.5**	**4.1**	**6.4**

Chapter 6

Conclusion

6.1 Summary

In Chapter 3, we have investigated the effect of hubness phenomenon on bilingual lexicon extraction, and indeed observed the emergence of hubs, which indicated the words in target language were frequently extracted as the translation words for given source words. It has suggested that the emergence of hub words were harmful to bilingual lexicon extraction. In addition to this results, we have extended the existing hubness reduction methods, centering and mutual proximity, to bilingual lexicon extraction. Theses methods have reduced the emergence of hub words, and thus improving the extraction accuracy accordingly.

In Chapter 4, we have first presented the theoretical analysis which explains why hubs emerge in zero-shot problem. Guided by this analysis, we have proposed a simple and straightforward method for reducing hubs. The proposed approach has efficiently reduced the hubs, and achieved better results compared with baseline on synthetic and real datasets.

In Chapter 5, we have casted suggested that k-nearest neighbor classifications can be regarded as zero-shot problem. From this point of view, we have extended the method proposed in Chapter 4 to ordinary classification problem. In our experiments, the method has surely reduced the emergence of hub, and showed better k-NN classification accuracies than the metric learning methods on most document and image datasets, and comparable on the rest. In addition, the proposed method was substantially faster in terms of training speed.

6.2 Future Directions

Future research topics include: (i) extending the analysis of Section 4.4 to cover multi-modal data distributions, or other similarity/distance measures such as cosine; (ii) investigating the influence of mapping directions in other regression-based ZSL methods, including neural networks; and (iii) investigating the emergence of hubs in CCA.

We have focused on multi-class classification problems in Chapter 5, but hubness is known to be harmful in other situations, such as clustering and semi-supervised classification in high-dimensional space [Radovanović et al., 2010a]. We plan to extend our approach to deal with such situations. We will also extend our method to learn nonlinear metrics.

Another direction of future work is to investigate the effect of our approach on kernel machines. Metric learning has been shown to be an effective preprocessing for kernel machines [Dhillon et al., 2010; Weinberger and Tesauro, 2007; Xu et al., 2013], and we will pursue a similar line using our approach.

Bibliography

Judit Ács, Katalin Pajkossy, and András Kornai. Building basic vocabulary across 40 languages. In *Proceedings of the 6th Workshop on Building and Using Comparable Corpora*, pages 52–58, 2013. 47

Zeynep Akata, Honglak Lee, and Bernt Schiele. Zero-shot learning with structured embeddings. *arXiv preprint arXiv:1409.8403v1*, pages 1–10, 2014. URL http://arxiv.org/pdf/1409.8403.pdf. 31, 51, 52

Ahmet Aker, Monica Paramita, and Robert Gaizauskas. Extracting bilingual terminologies from comparable corpora. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (ACL '13)*, pages 402–411, 2013. 12, 27

Rami Al-Rfou, Bryan Perozzi, and Steven Skiena. Polyglot: Distributed word representations for multilingual NLP. In *Proceedings of the 17th Conference on Computational Natural Language Learning (CoNLL '13)*, pages 183–192, 2013. 46

Daniel Andrade, Takuya Matsuzaki, and Jun'ichi Tsujii. Effective use of dependency structure for bilingual lexicon creation. In *Proceedings of the 12th international conference on Computational linguistics and intelligent text processing*, pages 80–92, 2011. 11

Gökhan Bakir, Thomas Hofmann, Bernhard Schölkopf, Alexander J. Smola, Ben Taskar, and S.V.N Vishwanathan, editors. *Predicting Structured Data*. MIT Press, 2007. 43

Aurélien Bellet, Amaury Habrard, and Marc Sebban. A survey on metric learning for feature vectors and structured data. *arXiv preprint arXiv:1306.6709*, 2014. URL http://arxiv.org/abs/1306.6709. 59

Ella Bingham and Heikki Mannila. Random projection in dimensionality reduction: Applications to image and text data. In *Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining (KDD '01)*, pages 245–250, 2001. 46

Dhouha Bouamor, Nasredine Semmar, and Pierre Zweigenbaum. Building specialized bilingual lexicons using word sense disambiguation. In *Proceedings of the 6th International Joint Conference on Natural Language Processing (IJCNLP '13)*, pages 952–956, 2013. 11

Blur V Dasarathy, editor. *Nearest Neighbor (NN) Norms: NN Pattern Classification Techniques*. IEEE Computer Society Press, 1991. 1

Sanjoy Dasgupta. Experiments with random projection. In *Proceedings of the 16th conference on Uncertainty in artificial intelligence (UAI '00)*, pages 143–151, 2000. 46

Jason V. Davis, Brian Kulis, Prateek Jain, Suvrit Sra, and Inderjit S. Dhillon. Information-theoretic metric learning. In *Proceedings of the 24th international conference on Machine learning (ICML '07)*, pages 209–216, 2007. 57, 59, 60, 62, 64

Paramveer S Dhillon, Partha Pratim Talukdar, and Koby Crammer. Learning better data representation using inference-driven metric learning. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics (ACL '10)*, pages 377–381, 2010. 68, 74

Mona Diab and Steve Finch. A statistical word-level translation model for comparable corpora. In *Proceedings of the Conference on Content-Based Multimedia Information Access*, pages 1500–1508, 2000. 12

Georgiana Dinu and Marco Baroni. How to make words with vectors: Phrase generation in distributional semantics. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (ACL '14)*, pages 624–633, 2014. 31, 32, 42, 43, 44, 46

Georgiana Dinu and Marco Baroni. Improving zero-shot learning by mitigating the hubness problem. In *Workshop Track on International Conference of Learning Rep-*

resentaion, 2015. URL http://arxiv.org/abs/1412.6568. 12, 31, 32, 42, 43, 44, 46

Jeff Donahue, Yangqing Jia, Oriol Vinyals, Judy Hoffman, Ning Zhang, Eric Tzeng, and Trevor Darrell. DeCAF: A deep convolutional activation feature for generic visual recognition. *arXiv preprint arXiv:1310.1531*, 2013. URL https://arxiv.org/abs/1310.1531. 47

Richard O Duda, Peter E Hart, and David G Stork. *Pattern Classificaiton*. Wiley-Interscience, 2000. 1

Lennart Eriksson, Erick Johansson, N. Kettaneh-Wold, Johan Trygg, C. Wikström, and S. Wold. *Multi- and Megavariate Data Analysis: Basic Principles and Applications*. Umetrics Academy, 2006. 14

Ali Farhadi, Ian Endres, Derek Hoiem, and David Forsyth. Describing objects by their attributes. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR '09)*, pages 1778–1785, 2009. 31

Douglas Fisher and Hans-Joachim Lenz. *Learning From Data: Artificial Intelligence and Statistics V*. Lecture Notes in Statistics 112. Springer, 1996. 14

Andrea Frome, Greg S Corrado, Jonathon Shlens, Samy Bengio, Jeffrey Dean, Marc'Aurelio Ronzato, and Tomas Mikolov. Devise: A deep visual-semantic embedding model. In *Advances in Neural Information Processing Systems (NIPS '13)*, pages 2121–2129, 2013. 31, 32, 43

Pascale Fung. Compiling bilingual lexicon entries from a non-parallel English-Chinese corpus. In *Proceedings of the 3rd Workshop on Very Large Corpora*, pages 173–183, 1995. 11

Pascale Fung and Lo Yuen Yee. An IR approach for translating new words from nonparallel, comparable texts. In *Proceedings of the 36th Annual Meeting of the Association for Computational Linguistics and 17th International Conference on Computational Linguistics (ACL '98)*, pages 414–420, 1998. 9

Aria Haghighi, Percy Liang, Taylor Berg-Kirkpatrick, and Dan Klein. Learning bilingual lexicons from monolingual corpora. In *Proceedings of the 46th Annual Meeting of the Association of Computational Linguistics: Human Language Technologies (ACL '08)*, pages 771–779, 2008. 11

Kazuo Hara, Ikumi Suzuki, Masashi Shimbo, Kei Kobayashi, Kenji Fukumizu, and Miloš Radovanović. Localized centering: Reducing hubness in large-sample data. In *Proceeding of the 29th AAAI Conference on Artificial Intelligence (AAAI '15)*, 2015. 65, 66

David R Hardoon, Sandor Szedmak, and John Shawe-Taylor. Canonical correlation analysis: an overview with application to learning methods. *Neural computation*, 16:2639–2664, 2004. 43, 44

Amir Hazem and Emmanuel Morin. Word co-occurrence counts prediction for bilingual terminology extraction from comparable corpora. In *Proceedings of the 6th International Joint Conference on Natural Language Processing (IJCNLP '13)*, pages 1392–1400, 2013. 11

Ann Irvine and Chris Callison-Burch. Supervised bilingual lexicon induction with multiple monolingual signals. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics (NAACL '13)*, 2013. 9, 12, 27

Prateek Jain, Brian Kulis, Jason V. Davis, and Inderjit S. Dhillon. Metric and kernel learning using a linear transformation. *Journal of Machine Learning Research*, 13: 519–547, 2012. 59, 62, 65

Herve Jegou, Hedi Harzallah, and Cordelia Schmid. A contextual dissimilarity measure for accurate and efficient image search. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR '07)*, pages 1–8, 2007. 44

Philipp Koehn and Kevin Knight. Learning a translation lexicon from monolingual corpora. In *Proceedings of the ACL-02 workshop on Unsupervised lexical acquisition*, pages 9–16, 2002. 9, 11, 12

Brian Kulis. Metric learning: A survey. *Foundations and Trends® in Machine Learning*, 5(4):287–364, 2013. 59

Christoph H. Lampert, Hannes Nickisch, and Stefan Harmeling. Learning to detect unseen object classes by between-class attribute transfer. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR '09)*, pages 951–958, 2009. 31

Hugo Larochelle, Dumitru Erhan, and Yoshua Bengio. Zero-data learning of new tasks. In *Proceedings of the 23rd National Conference on Artificial Intelligence (AAAI '08)*, pages 646–651, 2008. 31

Florian Laws, Lukas Michelbacher, Beate Dorow, Christian Scheible, Ulrich Heid, and Hinrich Schütze. A linguistically grounded graph model for bilingual lexicon extraction. In *Proceedings of the 23rd International Conference on Computational Linguistics (COLING '10)*, pages 614–622, 2010. 27

Angeliki Lazaridou, Elia Bruni, and Marco Baroni. Is this a wampimuk? Cross-modal mapping between distributional semantics and the visual world. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (ACL '14)*, pages 1403–1414, 2014. 42, 43, 44

Xiaodong Liu, Kevin Duh, and Yuji Matsumoto. Topic models + Word alignment = A flexible framework for extracting bilingual dictionary from comparable corpus. In *Proceedings of the 17th Conference on Computational Natural Language Learning (CoNLL '13)*, pages 212–221, 2013. 12, 27

Christopher D Manning, Prabhakar Raghavan, and Hinrich Schütze. *Introduction to Information Retrieval*. Cambridge University Press, 2008. 45

Kanti V Mardia, John T Kent, and John M Bibby. *Multivariate Analysis*. Academic Press, 1979. 14

Sebastian Mika, Bernhard Schölkopf, Alex Smola, Klaus-robert Müller, Matthias Scholz, and Gunnar Rätsch. Kernel PCA and de-noising in feature spaces. In *Advances in Neural Information Processing Systems (NIPS '98)*, pages 536–542, 1998. 44

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781v3*, 2013a. URL http://arxiv.org/abs/1301.3781. 32, 42, 43, 44, 46

Tomas Mikolov, Quoc V. Le, and Ilya Sutskever. Exploiting similarities among languages for machine translation. *arXiv preprint arXiv:1309.4168*, 2013b. URL http://arxiv.org/abs/1309.4168. 12, 31, 32, 46

David Mimno, Hanna M. Wallach, Jason Naradowsky, David A. Smith, and Andrew McCallum. Polylingual topic models. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing (EMNLP '09)*, pages 880–889, 2009. 12, 27

Emmanuel Morin and Emmanuel Prochasson. Bilingual lexicon extraction from comparable corpora enhanced with parallel corpora. In *Proceedings of the 4th Workshop on Building and Using Comparable Corpora: Comparable Corpora and the Web*, pages 27–34, 2011. 11

Mohammad Norouzi, Tomas Mikolov, Samy Bengio, Yoram Singer, Jonathon Shlens, Andrea Frome, Greg S Corrado, and Jeffrey Dean. Zero-shot learning by convex combination of semantic embeddings. In *International Conference on Learning Representations (ICLR '14)*, 2014. URL https://arxiv.org/abs/1312.5650. 31, 32

Mark Palatucci, Dean Pomerleau, Geoffrey Hinton, and Tom M Mitchell. Zero-shot learning with semantic output codes. In *Advances in Neural Information Processing Systems (NIPS '09)*, pages 1410–1418, 2009. 31, 32, 42, 43, 44

Emmanuel Prochasson and Pascale Fung. Rare word translation extraction from aligned comparable documents. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics (ACL '11)*, pages 1327–1335, 2011. 12

Miloš Radovanović, Alexandros Nanopoulos, and Mirjana Ivanović. Hubs in space: Popular nearest neighbors in high-dimensional data. *Journal of Machine Learning Research*, 11:2487–2531, 2010a. 1, 5, 6, 7, 10, 13, 14, 23, 26, 32, 33, 34, 35, 36, 45, 58, 61, 65, 66, 68, 74

Miloš Radovanović, Alexandros Nanopoulos, and Mirjana Ivanović. On the existence of obstinate results in vector space models. In *Proceedings of the 33rd international ACM SIGIR conference on Research and development in information retrieval (SIGIR '10)*, pages 186–193, 2010b. 26

Reinhard Rapp. Automatic identification of word translations from unrelated English and German corpora. In *Proceedings of the 37th annual meeting of the Association for Computational Linguistics on Computational Linguistics (ACL '99)*, pages 519–526, 1999. 9, 11

Dominik Schnitzer, Arthur Flexer, Markus Schedl, and Gerhard Widmer. Local and global scaling reduce hubs in space. *Journal of Machine Learning Research*, 13: 2871–2902, 2012. 1, 5, 11, 17, 18, 34, 45, 58, 65, 66

Richard Socher, Milind Ganjoo, Christopher D. Manning, and Andrew Y. Ng. Zero-shot learning through cross-modal transfer. In *Advances in Neural Information Processing Systems (NIPS '13)*, pages 935–943, 2013. 31, 32, 43

Ikumi Suzuki, Kazuo Hara, Masashi Shimbo, Marco Saerens, and Kenji Fukumizu. Centering similarity measures to reduce hubs. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing (EMNLP '13)*, pages 613–623, 2013. 1, 5, 7, 8, 11, 14, 15, 22, 23, 26, 34, 45, 58, 65

Akihiro Tamura, Taro Watanabe, and Eiichiro Sumita. Bilingual lexicon extraction from comparable corpora using label propagation. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP '12)*, pages 24–36, 2012. 9, 10, 11, 21, 22, 25

Nenad Tomašev, Jan Rupnik, and Dunja Mladenić. The role of hubs in cross-lingual supervised document retrieval. In *Advances in Knowledge Discovery and Data Mining (PAKDD '13)*, pages 185–196, 2013. 43, 45

Alexei Vinokourov, John Shawe-Taylor, and Nello Cristianini. Inferring a semantic representation of text via cross-language correlation analysis. In *Advances in Neural Information Processing Systems (NIPS '02)*, pages 1473–1480, 2002. 31

Ivan Vulić and Marie-Francine Moens. Cross-lingual semantic similarity of words as the similarity of their semantic word responses. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL '13)*, pages 106–116, 2013a. 9, 12, 27

Ivan Vulić and Marie-Francine Moens. A study on bootstrapping bilingual vector spaces from non-parallel data (and nothing else). In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing (EMNLP'13)*, pages 1613–1624, 2013b. 21

Ivan Vulić, Wim De Smet, and Marie-Francine Moens. Identifying word translations from comparable corpora using latent topic models. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics (ACL '11)*, pages 479–484, 2011. 12

Kilian Q. Weinberger and Lawrence K. Saul. Distance metric learning for large margin nearest neighbor classification. *Journal of Machine Learning Research*, 10:207–244, 2009. 57, 59, 60, 62, 64

Kilian Q Weinberger and Gerald Tesauro. Metric learning for kernel regression. In *Proceeding of the 11th International Conference on Artificial Intelligence and Statistics (AISTATS '07)*, pages 608–615, 2007. 68, 74

Jason Weston, Olivier Chapelle, André Elisseeff, Bernhard Schölkopf, and Vladimir Vapnik. Kernel dependency estimation. In *Advances in Neural Information Processing Systems (NIPS '02)*, pages 873–880, 2002. 43

Xindong Wu, Vipin Kumar, J. Ross Quinlan, Joydeep Ghosh, Qiang Yang, Hiroshi Motoda, Geoffrey J. McLachlan, Angus Ng, Bing Liu, Philip S. Yu, Zhi-Hua Zhou, Michael Steinbach, David J. Hand, and Dan Steinberg. Top 10 algorithms in data mining. *Knowledge and Information Systems*, 14(1):1–37, 2008. 1

Eric P. Xing, Andrew Y. Ng, Michael I. Jordan, and Stuart Russell. Distance metric learning, with application to clustering with side-information. In *Advances in Neural Information Processing Systems (NIPS '02)*, pages 505–512, 2002. 57, 59, 60

Zhixiang Xu, Kilian Q. Weinberger, and Olivier Chapelle. Distance metric learning for kernel machines. *arXiv preprint arxiv:1208.3422*, 2013. URL http://arxiv.org/abs/1208.3422. 68, 74

Yiming Ying and Peng Li. Distance metric learning with eigenvalue optimization. *Journal of Machine Learning Research*, 13:1–26, 2012. 57, 59, 60, 62, 64

Kun Yu and Jun'ichi Tsujii. Extracting bilingual dictionary from comparable corpora with dependency heterogeneity. In *Proceedings of the 2009 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL '09)*, pages 121–124, 2009. 11

Xiaojin Zhu, Zoubin Ghahramani, and John Lafferty. Semi-supervised learning using gaussian fields and harmonic functions. In *Proceedings of the 20th International Conference on Machine Learning (ICML '03)*, pages 912–919, 2003. 11, 22

List of Publications

Journal Paper

1. 重藤 優太郎, 鈴木 郁美, 原 一夫, 新保 仁, 松本 裕治. ハブの抑制によるコンパラブルコーパスからの対訳抽出精度の改善. 人工知能学会論文誌, Vol. 31, No. 2, pages E-F43_1–12, 2016.

International Conference

1. Yutaro Shigeto, Masashi Shimbo, and Yuji Matsumoto. A fast easy regression technique for k-NN classification without using negative pairs. Accepted at The Pacific-Asia Conference on Knowledge Discovery and Data Mining (PAKDD), 2017.

2. Yutaro Shigeto, Ikumi Suzuki, Kazuo Hara, Masashi Shimbo, and Yuji Matsumoto. Ridge regression, hubness, and zero-shot learning. Machine Learning and Knowledge Discovery in Databases (ECML/PKDD), pages 135–151, 2015.

Other Publications

1. 重藤 優太郎, 鈴木 郁美, 原 一夫, 新保 仁, 松本 裕治. Zero-shot learning における線形回帰の影響. 情報処理学会研究報告第 222 回自然言語処理研究会, pages 1–8, 2015.

2. 重藤 優太郎, 新保 仁, 松本 裕治. ベクトルのスパース化を用いた k 近傍法におけるハブの軽減. 情報処理学会研究報告第 216 回自然言語処理研究会第 101 回音声言語情報処理研究会 合同研究会, pages 1–6, 2014.

3. 重藤 優太郎, 新保 仁, 松本 裕治. 対訳抽出におけるハブの影響. 言語処理学会第 20 回年次大会 発表論文集, pages 388–391, 2014.

4. Yutaro Shigeto, Ai Azuma, Sorami Hisamoto, Shuhei Kondo, Tomoya Kose, Keisuke Sakaguchi, Akifumi Yoshimoto, Frances Yung, and Yuji Matsumoto. Construction of English MWE dictionary and its application to POS tagging. Proceedings of the 9th Workshop on Multiword Expressions, pages 139–144, 2013.

5. 重藤 優太郎, 東 藍, 近藤 修平, 北裏 龍太, 坂口 慶祐, 光瀬 智哉, 久本 空海, 吉本 暁文, Frances Yung, 松本 裕治. 英語の複単語表現辞書の構築と品詞タグ付けへの応用. 情報処理学会研究報告第 209 回自然言語処理研究会, pages 1–8, 2012.

Awards

1. Runner-Up for the Best Student Paper Award. European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases (ECML/PKDD). Ridge Regression, Hubness, and Zero-Shot Learning, 2015.

2. 学生奨励賞. 情報処理学会 第 101 回音声言語情報処理研究会 第 216 回自然言語処理研究会 合同研究会. ベクトルのスパース化を用いた k 近傍法におけるハブの軽減, 2014.