Machine Translation Report

Hao Xue

September 2018

1 Mathematical Review

1.1 Orthogonal Procrustes Problem

Given two matrces A and B, the aim of Orthogonal Procrustes is to find an orthogonal linear transformation such that A is most closely mapped to B, which can be formularized as

$$\arg\min_{\Omega} \|\Omega A - B\|_F$$
$$\text{subject to}\, \Omega^T \Omega = 1$$

where $\|.\|_F$ denotes Frobenius norm. This problem can be solved by conducting singular value decomposition of BA^T, which can be shown by defining matrix inner product $\langle A, B\rangle = \text{tr}(A^T B)$. Let

$$BA^T = U\Sigma V^T$$

then one has

$$\begin{aligned}
\arg\min_{\Omega} \|\Omega A - B\|_F^2 & \\
&= \langle \omega A - B, \omega A - B\rangle \\
&= \arg\min_{\Omega} \langle A, A\rangle + \langle B, B\rangle - 2\langle \Omega A, B\rangle \\
&= \arg\max_{\Omega} \langle \Omega A, B\rangle \\
&= \arg\max_{\Omega} \langle \Omega, BA^T\rangle \\
&= \arg\max_{\Omega} \langle \Omega, U\Sigma V^T\rangle \\
&= \arg\max_{\Omega} \langle U^T \Omega V, \Sigma\rangle \\
&= U(\arg\max_{\Omega'} \langle \Omega' \Sigma\rangle)V^T \\
&= UV^T
\end{aligned}$$

1.2 Canonical Correlation Analysis (CCA)

The principle of canonical correlation analysis (CCA) is to find linear combinations of observations so that the correlation between consequential statistics is maximized. There are two prime purposes of canonical correlation analysis :

- Data reduction: explain covariation between two sets of variables using small number of linear combinations.
- Data interpretation: find features (i.e., canonical variates) that are important for explaining covariation between sets of variables.

Let $X \in \mathbb{R}^{n\times p}$ be empirical observations of n samples with p features and $Y \in \mathbb{R}^{n\times q}$ be observations of n samples with q features. They are assumed to be standardized to zero mean and unit variance. Then consider the following linear combinations:

$$\boldsymbol{u} = X\boldsymbol{w_x}, \quad \boldsymbol{v} = Y\boldsymbol{w_y}$$

where $\boldsymbol{w}_x \in \mathbb{R}^p$ and $\boldsymbol{w}_y \in \mathbb{R}^q$ are termed as weight vectors and $\boldsymbol{u} \in \mathbb{R}^n$ and $\boldsymbol{v} \in \mathbb{R}^n$ are referred as canonical variates.

$$\boldsymbol{w_x}, \boldsymbol{w_y} = \arg\max_{\boldsymbol{a},\boldsymbol{b}} \operatorname{corr}(X\boldsymbol{a}, Y\boldsymbol{b})$$

To ensure the uniqueness of $\boldsymbol{a}$ and $\boldsymbol{b}$, additional constraints that $\boldsymbol{a}^T\boldsymbol{a} = 1$ and $\boldsymbol{b}^T\boldsymbol{b} = 1$ are inserted. Recall that X and Y are standardized, hence the correlation between X and Y is $\Sigma_{xy} = X^TY$. Similarly, $\Sigma_{xx} = X^TX$ and $\Sigma_{yy} = Y^TY$. Maximizing the correlation of canonical variates then become equivalent to maximizing the correlation coefficient between $\boldsymbol{u}$ and $\boldsymbol{v}$, which is the cosine similarity between $\boldsymbol{u}$ and $\boldsymbol{v}$ as well. Let $\cos\theta$ denotes the correlation coefficient of canonical variates:

$$\cos\theta = \frac{\boldsymbol{u}^T\boldsymbol{v}}{\|\boldsymbol{u}\|\|\boldsymbol{v}\|} = \frac{\boldsymbol{a}^T\Sigma_{XY}\boldsymbol{b}}{\sqrt{\boldsymbol{a}^T\Sigma_{XX}\boldsymbol{a}}\sqrt{\boldsymbol{b}^T\Sigma_{YY}\boldsymbol{b}}}$$

Then one can further construct the optimization problem below:

$$\begin{aligned} &\arg\max_{\boldsymbol{a},\boldsymbol{b}} \cos\theta = \boldsymbol{a}^T\Sigma_{XY}\boldsymbol{b} \\ &\text{subject to}\, \boldsymbol{a}^T\boldsymbol{a} = 1, \quad \boldsymbol{b}^T\boldsymbol{b} = 1 \end{aligned}$$

We can simplify the restriction by defining $\omega \in \mathbb{R}^{p\times q}$, $\boldsymbol{c} \in \mathbb{R}^n$ and $\boldsymbol{d} \in \mathbb{R}^q$ as

$$\begin{aligned} \Omega &= \Sigma_{XX}^{-\frac{1}{2}}\Sigma_{XY}\Sigma_{YY}^{-\frac{1}{2}} \\ \boldsymbol{c} &= \Sigma_{XX}^{\frac{1}{2}}\boldsymbol{a} \\ \boldsymbol{d} &= \Sigma_{XX}^{\frac{1}{2}}\boldsymbol{b} \end{aligned}$$

Now we can reform the optimization problem as:

$$\begin{aligned} &\arg\max_{\boldsymbol{c},\boldsymbol{d}} \cos\theta = \boldsymbol{c}^T\Omega\boldsymbol{d} \\ &\text{subject to}\, \boldsymbol{c}^T\boldsymbol{c} = 1, \quad \boldsymbol{d}^T\boldsymbol{d} = 1 \end{aligned}$$

with Lagrangian:

$$\mathcal{L}(\boldsymbol{c}, \boldsymbol{d}, \lambda_1, \lambda_2) = \boldsymbol{c}^T\Omega\boldsymbol{d} - \frac{\lambda_1}{2}(\boldsymbol{c}^T\boldsymbol{c} - 1) - \frac{\lambda_2}{2}(\boldsymbol{d}^T\boldsymbol{d} - 1)$$

By taking the gradient of the Lagrangian with respect to $\boldsymbol{c}$ and $\boldsymbol{d}$ respectively, one obtains

$$\begin{aligned} \Omega\boldsymbol{d} &= \lambda_1\boldsymbol{c} \\ \Omega^T\boldsymbol{c} &= \lambda_2\boldsymbol{d} \end{aligned}$$

which leads to

$$\boldsymbol{c}^T\Omega\boldsymbol{d} = \lambda_1 \tag{1}$$

$$\boldsymbol{d}^T\Omega^T\boldsymbol{c} = \lambda_2 \tag{2}$$

since $\boldsymbol{c}^T\Omega\boldsymbol{b} = \boldsymbol{d}^T\Omega^T\boldsymbol{c}$, this implies that $\boldsymbol{c}$ is the left unit singular vector and $\boldsymbol{d}$ is the right unit singular vector of Ω with singular value λ.

1.3 Kernel Canonical Correlation Analysis (KCCA)

It is likely that there exists non-linear relation between X and Y, to capture this non-linear relation, kernel method is applied[1], that is transform the original observations $\boldsymbol{x}_i$ and $\boldsymbol{y}_i$, correspondingly, from original spaces to Hilbert spaces by feature maps $\Phi(\boldsymbol{x}_i) : \mathbb{R}^p \mapsto \mathbb{H}_x$ and $\Psi(\boldsymbol{y}_i) : \mathbb{R}^q \mapsto \mathbb{H}_y$. The similarity of objects is then measured by a symmetric positive semi-definite matrix, i.e. Gram matrices, K_x and K_y, where the element at the ith row and jth column of K_x is $(K_x)_{ij} = \Phi(\boldsymbol{x}_i)^T \Phi(\boldsymbol{x}_j)$ and similarly, $(K_y)_{ij} = \Psi(\boldsymbol{y}_i)^T \Psi(\boldsymbol{y}_j)$. The weight vector after kernelization can be written as

$$\boldsymbol{a}_\Phi = \sum_{i=1}^{n} \alpha_i \Phi(\boldsymbol{x}_i) = \Phi(X)^T \boldsymbol{\alpha} \quad \text{and} \quad \boldsymbol{b}_\Psi = \sum_{i=1}^{n} \beta_i \Psi(\boldsymbol{y}_i) = \Psi(Y)^T \boldsymbol{\beta}$$

where $\Phi(X) = [\Phi(\boldsymbol{x}_1), \Phi(\boldsymbol{x}_2), \ldots, \Phi(\boldsymbol{x}_n)]^T$, $\Psi(Y) = [\Psi(\boldsymbol{y}_1), \Psi(\boldsymbol{y}_2), \ldots, \Psi(\boldsymbol{y}_n)]^T$ and both $\boldsymbol{\alpha} and \boldsymbol{\beta} denote the corresponding line$
$\boldsymbol{\alpha}^T \Phi\Phi^T \Psi\Psi^T \boldsymbol{\beta}$, which results in optimization problem:

$$\begin{aligned} &\arg\max_{\boldsymbol{\alpha},\boldsymbol{\beta}} \boldsymbol{\alpha}^T K_x K_y \boldsymbol{\beta} \\ &\text{subject to}\, \boldsymbol{\alpha}^T K_x^2 \boldsymbol{\alpha} = 1,\, \boldsymbol{\beta}^T K_y^2 \boldsymbol{\beta} = 1 \end{aligned}$$

KCCA can be further regularized with regularization parameter γ as below

$$\begin{aligned} &\arg\max_{\boldsymbol{\alpha},\boldsymbol{\beta}} \frac{\boldsymbol{\alpha}^T K_x K_y \boldsymbol{\beta}}{\sqrt{\boldsymbol{\alpha}^T (K_x^2 + \gamma) \boldsymbol{\alpha} \boldsymbol{\beta}^T (K_y^2 + \gamma) \boldsymbol{\beta}}} \\ &\text{subject to}\, \boldsymbol{\alpha}^T (K_x^2 + \gamma) \boldsymbol{\alpha} = 1,\, \boldsymbol{\beta}^T (K_y^2 + \gamma) \boldsymbol{\beta} = 1 \end{aligned}$$

similarly, we can solve this by singular value decomposition described in previous subsection

2 Experiment

2.1 Source Data

List of word embedding files:

- zhvec: Chinese word embedding file provided by fastText(300-dim)[2]
- wordvec: Chinese word embedding (500-dim)
- termvec: termvecs trained by Luwan (500-dim)
- cuivec: cui vectors which we have from the begining (200-dim)
- wikivec: English word embedding file provided by fastText, which is trained with wikipedia and statmt news (300-dim)[3]
- crawlvec: English word embedding file provided by fastText, which is trained on Common Crawl (300-dim)

List of dictionaries:

- Xiang Ya Dictionary: Chinese-English medical dictionary
- Wikidata: Chinese-English dictionary
- gt_zh: google translation of every terms in of zhvec
- gt_wordvec: google translation of every terms of wordvec
- CUIdict: English-CUI dictionary obtained from UMLS

2.2 Matching Procedure

Terms of both wordvec and zhvec are translated into English according to corresponding google tranlsation and corrected by Xiang Ya Dictionary and Wikidata. Furthermore, since we only care about medical terms, only vectors with terms intersecting with words contained by CUIdict are kept. The flow chart illustrate how this matching is done.

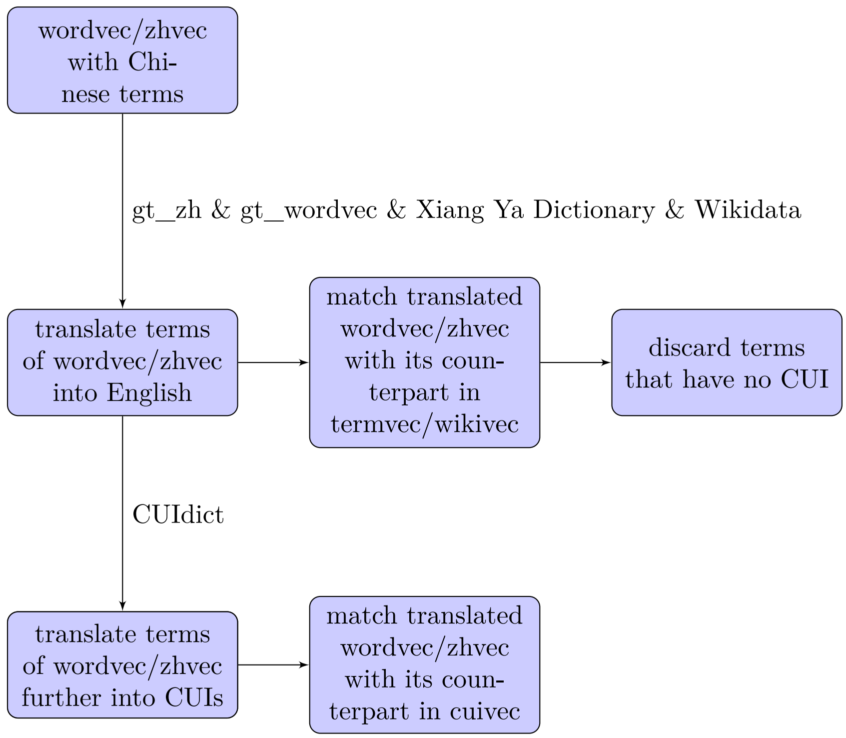


Table 2.3 below records the matching result, matched pairs stands for how many vectors pairs from two different language spaces are matched and intersections with CUIs shows the quantity of pairs whose term has its corresponding CUI.

Table 1: matching result

	matched pairs	intersections with CUIs
zhvec to termvec	13066	7059
zhvec to wikivec	13306	6993
zhvec to cuivec	2950	2950
wordvec to termvec	15164	10042
wordvec to wikivec	13322	8880
wordvec to cuivec	7577	7577

2.3 Results

Here we employ a similar structure to [4], setting training set size to be 5000 and testing to be 2000. When the dimensions of vectors from two language spaces are incompatible for Orthogonal Procruste, the vectors with larger dimension are simply truncated to accord with the ones with smaller dimension.

To reduce effect of overfitting, CCA is regularized with regularization parameter γ stepping from 10, 20 to 100 then from 100, 200 to 1000. Different number of canonical variates n are tried at the same time, varying from 50, 100 by each step equals 50, till the original dimension . For weighted CCA, each canonical component is additionally weighted by its principle component scores (i.e. singular value of Ω in (1)) divided by sum of all principle component scores. See Table below for value of regularization parameter γ and number of canonical component n and their corresponding training and testing set accuracy when the maximal test accuracy occurs. As comparison, Joulin et.al [5] uses cross-domain similarity local scaling criterion instead of leaner regression as loss function, which gives 0.4460 accuracy for En-Zh and 0.4560 for Zh-En (using data from fastText). Though the structure of training and testing sets are not stated explicitly in this paper, given that this paper is a related work to [4], the assignments of training and testing sets is assumed to be invariant.

	Set size		Procruste	
	train	test	train	test
zhvec to termvec	2000	5000	0.4984	0.2215
zhvec to wikivec	1993	5000	0.5996	0.4486
zhvec to cuivec	590	2360	0.5419	0.3441
wordvec to termvec	2000	5000	0.2758	0.1230
wordvec to wikivec	2000	5000	0.2302	0.1145
wordvec to cuivec	2000	5000	0.2890	0.3055

	Weighted CCA				Ordinary CCA			
	train	test	γ	n	train	test	γ	n
zhvec to termvec	0.6320	0.3250	100	250	0.5320	0.3150	500	100
zhvec to wikivec	**0.6390**	**0.4847**	50	200	0.5956	0.4867	200	1000
zhvec to cuivec	0.7059	0.4203	40	200	0.6462	0.4051	200	100
wordvec to termvec	0.2570	0.1940	200	100	0.3248	0.1970	70	90
wordvec to wikivec	0.2730	0.1895	70	100	0.2612	0.1825	800	100
wordvec to cuivec	**0.4196**	**0.4440**	80	100	0.3402	0.4170	600	90

As can be seen from the table that the highest accuracy that mapping zhvec to English space is achieved by weighted CCA with $\gamma = 50$ and $n = 200$. Since the size of data for zhvec to cuivec is insufficient to validate the significance of results, they may be ignored. Since γ and n are recorded only when the maximal testing set accuracy occurs, for wordvec to cuivec case, the accuracy for testing set are slightly higher than that in training set. However, it is bizarrd that in Orthogonal Procruste, we have the same problem.

Here, mapping results from two scenarios, wordvec to cuivec and zhvec to wikivec, are chosen and demonstrated in Excel files. The former is shown in Figure 1, whose columns are arranged as following:

- A: CUI of Chinese term in column B
- B: Chinese term of wordvec
- C: English translation of column B given by google translate
- D: English term predicted by weighted CCA
- E: Chinese translation of Column D (this column is added manually just for convenience)
- F: T if prediction of this CUI by weighted CCA is correct, F otherwise
- G: CUI of the Chinese term in column B
- H: CUI of the English term in column D
- I-end: all English words that belong to CUI in column G

They are mainly clinical terminologies, and among false translation, those terms that share conceptional similarity were manually highlighted. Those highlighted words are synonyms, opposites or one specific instance of the other. For example, it is quite often that the predicted CUI refers to one particular brand of the drug while Chinese terms refers to the category of this drug or its principle chemical component.

As for the latter shown in Figure 2:

	A	B	C	D	E	F	G	H	I
28	C0003445	抗毒素	antitoxin	toxoid	类毒素	F	C0003445	C0040555	toxoids
29	C0003452	鹿角	antlers	color taste	颜色味道	F	C0003452	C0392700	colour taste
30	C0003467	焦虑反应	anxiety reaction	despair	绝望	F	C0003467	C0233488	feeling despair
31	C0003469	神经质	anxiety disorder	depersonalization	人格解体	F	C0003469	C0011551	depersonalisatic
32	C0003641	pancreatic	aprotinin	cauda pancreatis	胰腺炎	F	C0003641	C0227590	pancreatic tail
33	C0003704	蛛形纲	arachnida	class insecta	班级昆虫	F	C0003704	C0021585	insects
34	C0003811	心律失常	heart arrhythmia	increased heart rate	心率加快	F	C0003811	C0039231	high pulse rate
35	C0003819	砷剂	arsenical	vinblastine	长春碱	F	C0003819	C0042670	vlb
36	C0003956	升主动脉	aorta ascendens	outflow tract of right	右心室流出道	F	C0003956	C0225892	outflow tract of
37	C0004048	吸气	inspiration	tidal volume	潮量	F	C0004048	C0040210	respiratory airw
38	C0004134	失调	ataxia	hyperphagia	饮食过量	F	C0004134	C0020505	extreme overeati
39	C0004268	注意	attention	pertinent information	相关信息	F	C0004268	C1301772	pertinent inform
40	C0004271	形势	outlook	understanding	理解	F	C0004271	C0162340	comprehension
41	C0004409	生长素	auxin	luteotropin	催乳	F	C0004409	C0033371	mammary stimulat
42	C0004454	腋窝的	axillary	latissimus dorsi muscl	背阔肌	F	C0004454	C0224362	musculus latissi
43	C0004461	神经纤维	axon	nerve fiber	神经纤维	F	C0004461	C0027749	fiber nerve
44	C0004504	唑类	azoles	itraconazole	伊曲康唑	F	C0004504	C0064113	icz
45	C0004576	巴贝虫病	babesiasis	rabies	狂犬病	F	C0004576	C0034494	rabies virus inf

Figure 1: screen-shot of wordvec to cuivec tranlsation results

	A	B	C	D
181	凝固	clotting	dissolve	F
182	凝胶	agar	polymer	F
183	粪子	stools	stool	F
184	出版社	publishing	monograph	F
185	出生	parturition	daughter	F
186	分割	dissection	bifurcate	F
187	分叉	bifurcate	anastomoses	F
188	分子量	molecular	molecule	F
189	分担	sharing	share	F
190	分數	fraction	grade	F
191	分歧	bifurcation	conflict	F
192	切	cutting	slice	F
193	刑部	criminal	forensics	F

Figure 2: screen-shot of zhvec to wikivec tranlsation results

- A: Chinese term of zhvec
- B: English translation of column A given by google translate
- C: English term predicted by weighted CCA
- D: T if prediction of this CUI by weighted CCA is correct, F otherwise

Since both zhvec and wikivec were trained on general corpora, even only terms having CUI are considered, most of them looks like non-clinical terms. In this case, the selection is stricter compared with former case, in other words, only terms pairs that are synonyms were highlighted.

3 Future Work

Since tuning KCCA parameters is quite time consuming and the outcome is not satisfactory, the result is not given yet. Similar process of CCA can be applied to KCCA to see whether overfitting can be controlled. Word similarity check of wordvec revealed that for clinical terms, good quality of synonyms were returned which satisfies our needs but for non-clinical terms, those terms returned with high cosine similarity did not make good sense. It would also be helpful to check the size of corpora to ensure they are large enough and compatible with the size of corpora used to train zhvec and wikivec (16 billion tokens). Other techniques that turn word into vectors like GloVe may also be considered as alternative of word2vec.

References

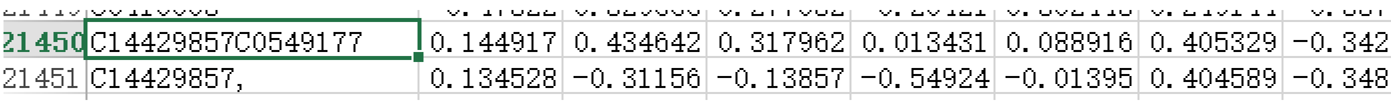
[1] Natalia Y. Bilenko and Jack L. Gallant. Pyrcca: Regularized kernel canonical correlation analysis in python and its applications to neuroimaging. *Frontiers in Neuroinformatics*, 10:49, 2016.

[2] Edouard Grave, Piotr Bojanowski, Prakhar Gupta, Armand Joulin, and Tomas Mikolov. Learning word vectors for 157 languages. In *Proceedings of the International Conference on Language Resources and Evaluation (LREC 2018)*, 2018.

[3] Tomas Mikolov, Edouard Grave, Piotr Bojanowski, Christian Puhrsch, and Armand Joulin. Advances in pre-training distributed word representations. In *Proceedings of the International Conference on Language Resources and Evaluation (LREC 2018)*, 2018.

[4] Tomas Mikolov, Quoc V. Le, and Ilya Sutskever. Exploiting similarities among languages for machine translation. *CoRR*, abs/1309.4168, 2013.

[5] Armand Joulin, Piotr Bojanowski, Tomas Mikolov, and Edouard Grave. Improving supervised bilingual mapping of word embeddings. *CoRR*, abs/1804.07745, 2018.



21450	C14429857C0549177	0.144917	0.434642	0.317962	0.013431	0.088916	0.405329	-0.342
21451	C14429857,	0.134528	-0.31156	-0.13857	-0.54924	-0.01395	0.404589	-0.348

Figure 3: erroneous data in CUIvec

Appendix

Table 2: Kernel Descriptions

Gaussian RBF kernel	$k(\boldsymbol{x}, \boldsymbol{x'}) = \exp(-\sigma\|\boldsymbol{x} - \boldsymbol{x'}\|^2)$
Polynomial kernel	$k(\boldsymbol{x}, \boldsymbol{x'}) = (scale\langle \boldsymbol{x}, \boldsymbol{x'}\rangle + offset)^{degree}$
Laplacian kernel	$k(\boldsymbol{x}, \boldsymbol{x'}) = \exp(-\sigma\|\boldsymbol{x} - \boldsymbol{x'}\|)$

Error Report

In CUIvec, there is one erroneous line as showing by Figure 3. This line was simply eradicated from the file, hopefully, this error did not influence the accuracy of other CUIs.