# Bridging Languages through Images with Deep Partial Canonical Correlation Analysis: Supplementary Material

**Guy Rotman**[1], **Ivan Vulić**[2] and **Roi Reichart**[1]
[1] Faculty of Industrial Engineering and Management, Technion, IIT
[2] Language Technology Lab, University of Cambridge
grotman@campus.technion.ac.il
iv250@cam.ac.uk roiri@technion.ac.il

## 1 DPCCA Variant A

As stated in §4 of the paper, Algorithm 1 provides the pseudo-code of the optimization algorithm for DPCCA Variant A (illustrated by Figure 1). As discussed in the paper, DPCCA Variant A may be seen as a special case of Variant B: $\boldsymbol{Z}$ is kept fixed with Variant A.

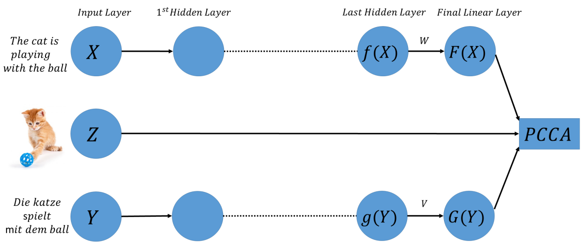


Figure 1: DPCCA (Variant A). $\boldsymbol{X}$ and $\boldsymbol{Y}$ (English and German image descriptions) are fed through two identical deep feed-forward neural networks followed by a final linear layer, while $\boldsymbol{Z}$ (an image) is kept fixed. Finally, the final nodes of the networks $\boldsymbol{F}(\boldsymbol{X})$ and $\boldsymbol{G}(\boldsymbol{Y})$ are maximally correlated conditioned on $\boldsymbol{Z}$.

## 2 Baselines

In this section we elaborate on the baselines we compared our models against. For all baselines, we employ the same hyperparameter tuning protocol as for our model (see §3 of this supplementary material), and report test set results with the configuration that performed best on the validation set. Finally, we use the same network architectures across all DCCA variants (see below).

**CCA** Canonical Correlation Analysis (Hotelling, 1936). The objective of this model is to find a linear transformation of two aligned sets of variables, $(\boldsymbol{x_i}, \boldsymbol{y_i})_{1=1}^{N}$, such that the correlation between the transformed variables is maximized. We use the implementation of this algorithm in the scikit-learn package http://www.scikit-learn.org.

**DCCA** The Deep CCA model (Andrew et al., 2013) is an extension of CCA to a deep feed-forward neural network. We use the implementation of the algorithm as suggested in (Wang et al., 2015a), which uses stochastic optimization with large minibatches. Code is available at http://ttic.uchicago.edu/~wwang5/dccae.html.

**DCCA_NOI** This is the same Deep CCA model as in the original DCCA, but the optimization is based on Nonlinear Orthogonal Iterations (NOI) (Wang et al., 2015b), another form of stochastic optimization. The covariance estimates are estimated using a moving average technique and hence the algorithm is not restricted to run over large minibatches. We implemented this algorithm and submit it as part of our code.

**DCCA_SDL** Deep CCA with a Stochastic Decorrelation Loss (SDL) (Chang et al., 2017). Instead of enforcing the whitening constraints explicitly, an L1 objective term is placed over the off-diagonal elements of the covariance estimates. The code for this model is not publicly available, we implemented it as part of our research.

**DCCAE** The Deep CCA Autoencoder model (Wang et al., 2015a) extends the DCCA model by adding an autoencoder component which aims to reconstruct the model's inputs. The DCCAE objective provides a trade-off between maximizing the correlation between the two sets of variables on the one hand, and finding informative features for the reconstruction of these variables on the other hand. We use the implementation from the authors of the original paper. Code is available at http://ttic.uchicago.edu/~wwang5/dccae.html.

**Algorithm 1** The non-linear orthogonal iterations (NOI) algorithm for DPCCA Variant A

**Input:** Data matrices $\boldsymbol{X} \in \mathbb{R}^{D_x \times N}$, $\boldsymbol{Y} \in \mathbb{R}^{D_y \times N}$, $\boldsymbol{Z} \in \mathbb{R}^{D_z \times N}$, time constant $\rho$, learning rate $\eta$.
**initialization:** Initialize weights $(\boldsymbol{W_F}, \boldsymbol{V_G})$.
Randomly choose a minibatch $(\boldsymbol{X_{b_0}}, \boldsymbol{Y_{b_0}}, \boldsymbol{Z_{b_0}})$.
Initialize covariances:
$\hat{\boldsymbol{\Sigma}}_{\boldsymbol{FF}} \leftarrow \frac{N-1}{|b_0|}\boldsymbol{F}(\boldsymbol{X_{b_0}})\boldsymbol{F}(\boldsymbol{X_{b_0}})^{\boldsymbol{T}}$
$\hat{\boldsymbol{\Sigma}}_{\boldsymbol{GG}} \leftarrow \frac{N-1}{|b_0|}\boldsymbol{G}(\boldsymbol{Y_{b_0}})\boldsymbol{G}(\boldsymbol{Y_{b_0}})^{\boldsymbol{T}}$
$\hat{\boldsymbol{\Sigma}}_{\boldsymbol{ZZ}} \leftarrow \frac{N-1}{|b_0|}\boldsymbol{Z_{b_0}}\boldsymbol{Z_{b_0}^T}$
$\hat{\boldsymbol{\Sigma}}_{\boldsymbol{FZ}} \leftarrow \frac{N-1}{|b_0|}\boldsymbol{F}(\boldsymbol{X_{b_0}})\boldsymbol{Z_{b_0}^T}$
$\hat{\boldsymbol{\Sigma}}_{\boldsymbol{GZ}} \leftarrow \frac{N-1}{|b_0|}\boldsymbol{G}(\boldsymbol{Y_{b_0}})\boldsymbol{Z_{b_0}^T}$

**for** $t = 1, 2, \ldots, n$ **do**
Randomly choose a minibatch $(\boldsymbol{X_{b_t}}, \boldsymbol{Y_{b_t}}, \boldsymbol{Z_{b_t}})$.
Update covariances:
$\hat{\boldsymbol{\Sigma}}_{\boldsymbol{FF}} \leftarrow \rho\hat{\boldsymbol{\Sigma}}_{\boldsymbol{FF}} + (1-\rho)\frac{N-1}{|b_t|}\boldsymbol{F}(\boldsymbol{X_{b_t}})\boldsymbol{F}(\boldsymbol{X_{b_t}})^{\boldsymbol{T}}$
$\hat{\boldsymbol{\Sigma}}_{\boldsymbol{GG}} \leftarrow \rho\hat{\boldsymbol{\Sigma}}_{\boldsymbol{GG}} + (1-\rho)\frac{N-1}{|b_t|}\boldsymbol{G}(\boldsymbol{Y_{b_t}})\boldsymbol{G}(\boldsymbol{Y_{b_t}})^{\boldsymbol{T}}$
$\hat{\boldsymbol{\Sigma}}_{\boldsymbol{ZZ}} \leftarrow \rho\hat{\boldsymbol{\Sigma}}_{\boldsymbol{ZZ}} + (1-\rho)\frac{N-1}{|b_t|}\boldsymbol{Z_{b_t}}\boldsymbol{Z_{b_t}^T}$
$\hat{\boldsymbol{\Sigma}}_{\boldsymbol{FZ}} \leftarrow \rho\hat{\boldsymbol{\Sigma}}_{\boldsymbol{FZ}} + (1-\rho)\frac{N-1}{|b_t|}\boldsymbol{F}(\boldsymbol{X_{b_t}})\boldsymbol{Z_{b_t}^T}$
$\hat{\boldsymbol{\Sigma}}_{\boldsymbol{GZ}} \leftarrow \rho\hat{\boldsymbol{\Sigma}}_{\boldsymbol{GZ}} + (1-\rho)\frac{N-1}{|b_t|}\boldsymbol{G}(\boldsymbol{Y_{b_t}})\boldsymbol{Z_{b_t}^T}$

Update conditional variables:
$\boldsymbol{F}|\boldsymbol{Z} \leftarrow \boldsymbol{F}(\boldsymbol{X_{b_t}}) - \hat{\boldsymbol{\Sigma}}_{\boldsymbol{FZ}}\hat{\boldsymbol{\Sigma}}_{\boldsymbol{ZZ}}^{-1}\boldsymbol{Z_{b_t}}$
$\boldsymbol{G}|\boldsymbol{Z} \leftarrow \boldsymbol{G}(\boldsymbol{Y_{b_t}}) - \hat{\boldsymbol{\Sigma}}_{\boldsymbol{GZ}}\hat{\boldsymbol{\Sigma}}_{\boldsymbol{ZZ}}^{-1}\boldsymbol{Z_{b_t}}$
$\hat{\boldsymbol{\Sigma}}_{\boldsymbol{FF}|\boldsymbol{Z}} \leftarrow \hat{\boldsymbol{\Sigma}}_{\boldsymbol{FF}} - \hat{\boldsymbol{\Sigma}}_{\boldsymbol{FZ}}\hat{\boldsymbol{\Sigma}}_{\boldsymbol{ZZ}}^{-1}\hat{\boldsymbol{\Sigma}}_{\boldsymbol{FZ}}^{\boldsymbol{T}}$
$\hat{\boldsymbol{\Sigma}}_{\boldsymbol{GG}|\boldsymbol{Z}} \leftarrow \hat{\boldsymbol{\Sigma}}_{\boldsymbol{GG}} - \hat{\boldsymbol{\Sigma}}_{\boldsymbol{GZ}}\hat{\boldsymbol{\Sigma}}_{\boldsymbol{ZZ}}^{-1}\hat{\boldsymbol{\Sigma}}_{\boldsymbol{GZ}}^{\boldsymbol{T}}$

Fix $\widetilde{\boldsymbol{G}|\boldsymbol{Z}} = \hat{\boldsymbol{\Sigma}}_{\boldsymbol{GG}|\boldsymbol{Z}}^{-\frac{1}{2}}\boldsymbol{G}|\boldsymbol{Z}$, and compute $\nabla\boldsymbol{W_F}$ with respect to:
$\min_{\boldsymbol{W_F}} \frac{1}{|b_t|}\|\boldsymbol{F}|\boldsymbol{Z} - \widetilde{\boldsymbol{G}|\boldsymbol{Z}}\|_F^2$

Update parameters:
$\boldsymbol{W_F} \leftarrow \boldsymbol{W_F} - \eta\nabla\boldsymbol{W_F}$

Fix $\widetilde{\boldsymbol{F}|\boldsymbol{Z}} = \hat{\boldsymbol{\Sigma}}_{\boldsymbol{FF}|\boldsymbol{Z}}^{-\frac{1}{2}}\boldsymbol{F}|\boldsymbol{Z}$, and compute $\nabla\boldsymbol{V_G}$ with respect to:
$\min_{\boldsymbol{V_G}} \frac{1}{|b_t|}\|\boldsymbol{G}|\boldsymbol{Z} - \widetilde{\boldsymbol{F}|\boldsymbol{Z}}\|_F^2$

Update parameters:
$\boldsymbol{V_G} \leftarrow \boldsymbol{V_G} - \eta\nabla\boldsymbol{V_G}$
**end for**
**Output:** $(\boldsymbol{W_F}, \boldsymbol{V_G})$

**GCCA** The Generalized CCA model (Horst, 1961; Rastogi et al., 2015) extends the CCA model by maximizing the sum of canonical correlations of more than one pair of variable sets. Like CCA, the transformations are also linear. In contrast to PCCA, GCCA treats all variable sets in its input as equal and aims to maximize the correlation between any pair of variable sets. We use the implementation of Funaki and Nakayama (2015) in order to learn a shared space of two linguistic representations and their corresponding images. Code is available at `http://github.com/rupy/GCCA`.

**NCCA** Nonparametric CCA (Michaeli et al., 2016). A non-parametric variant of CCA. The model obtain optimal nonparametric projections from the SVD of a kernel defined via the pointwise mutual information between two sets of variables. We use the authors' implementation. Code is available at `http://webee.technion.ac.il/people/tomermic/`.

**PPCCA** The Probabilistic Partial CCA model is a probabilistic variant of PCCA. Mukuta and Harada (2014) solve the objective with a Bayesian estimation technique. We use the authors' implementation. Code is available at `http://github.com/mil-tokyo/bayes-pcca`.

**BCN** The main modeling assumption of the Bridge Correlational Network (BCN) (Rajendran et al., 2016) is that there exists a pivot view which shares parallel data with the other, non-pivot views, that in turn do not share parallel data with each other. In (Rajendran et al., 2016) the pivot view consists of English sentences, while the non-pivot views are sentences in other languages. We use the authors' implementation in our experiments. Unlike in the original paper we experimented with either English sentences or with images as the pivot view. The BCN code is available at `http://sarathchandar.in/bridge-cornet`.

**IMG_PIVOT** The Image Pivoting model Gella et al. (2017). The model aims to learn representations of images and their descriptions in multiple languages. The loss function encourages images and their descriptions (modeled by LSTMs), as well as descriptions of the same image in different languages, to be closer to each other compared to images and sentences that do not describe them and to sentences in different language that do not describe the same image. The authors were very kind to provide us with the visual and textual embeddings outputted by their strongest model variants: PARALLEL-ASYM and PARALLEL-SYM (they ran their models on our test set after it was trained in the setup described in their paper).

**PCCA and DPCCA** PCCA was implemented with the scikit-learn http://www.scikit-learn.org package. DPCCA is implemented in pytorch.[1] Our code is submitted with the paper.

## 3 Hyperparameter Tuning

The hyperparameters of the different models are tuned with a grid search over the following values: {2,3,4,5} for number of layers, {*tanh, sigmoid, ReLU*} as the activation functions (we use the same activation function in all the layers of the same network), {64,128,256} for minibatch size, {0.001,0.0001} for learning rate, and {128,256} for $L$ (the size of the output vectors). The dimensions of all mid-layers are set to the input size. We use the Adam optimizer (Kingma and Ba, 2015), with the number of epochs set to 300. As mentioned in the main paper, the size of the input layer is 500 for the sentence task and 300 for the word task. Note, that not all models have all the aforementioned hyer-parameters (e.g. non-deep models like CCA and PCCA do not have activation functions and intermediate layers).

We used the word2vec skip-gram implementation in the Gensim package (http://radimrehurek.com/gensim/index.html). We trained skip-gram on the January 2017 Wikipedia dump (http://dumps.wikimedia.org/), and on the Wacky corpus (http://wacky.sslmit.unibo.it/doku.php?id=corpora). For visual features we use the publicly available FC_7 features (Simonyan and Zisserman, 2015) (http://www.statmt.org/wmt16/multimodal-task.html). Finally, the initial vectors we use in the multilingual word similarity task are available at http://github.com/nmrksic/attract-repel.

For word similarity, following a standard practice (Levy et al., 2015; Vulić et al., 2017) we tune all models on one half of the SimLex data and evaluate on the other half, and vice versa. The reported score is the average of the two halves.

## 4 Word Similarity Results

As promised in §7 of the main paper, we complement Table 2 of the main paper by presenting similar tables for the EN-IT (Table 1) and EN-RU (Table 2) setups. Like in the EN-DE setup, DPCCA gets more substantial improvements on adjectives and verbs compared to nouns. In both tables this is true for both languages, with the exception of English adjectives for the EN-RU setup, where the GCCA baseline outperforms both DPCCA variants (although DPCCA Variant B is the second best model in that setup too).

Like in the EN-DE setup of table 2 of the main paper, the DPCCA representations of the language trained jointly with English (DE, IT or RU) gain more than the English representations (in terms of improvement over the best baseline on the POS class). We keep the explanation of this phenomenon to future work.

## 5 Qualitative Results

**Cross-lingual Image Description Retrieval: Examples** Figure 2 presents representative results from the cross-lingual image description retrieval task. Results are presented for DPCCA Variant B and DCCA_NOI - the strongest models on this task. We split the examples to two parts: retrieval from English to German, and retrieval from German to English. In each part we show one example where both models are correct, two examples where our model is correct and the other model is incorrect, and a final example where our model is incorrect and the other model is correct.

These examples suggest that even when our model retrieves an incorrect description, this description is close to the correct description, and shares objects, or actions with it. For example, the correct description for the bottom example is "A dog jumps over an obstacle outside", while our model retrieved the description "A dog is jumping through a fiery obstacle". Indeed both description share objects: dog and obstacle, as well as an action: jump.

DCCA_NOI, in contrast, does not show this favorable behavior. For example, in the second (from top) EN-DE example it gets the main object incorrectly (footballers vs foxterrier), while in the third (again from top) DE-EN example it gets the main scene incorrectly (skateboarding rump vs. lake).

We observed these patterns quite frequently. In many cases when DPCCA makes a mistake it still generally understands the main scene, actions and objects, while for DCCA_NOI that was case only in a small fraction of its mistakes. The above exam-

[1] http://www.pytorch.org/.

| Model | English-Italian | | | | | |
|---|---|---|---|---|---|---|
| | EN-Adj | EN-Verbs | EN-Nouns | IT-Adj | IT-Verbs | IT-Nouns |
| DPCCA (Variant A) | 0.672 | **0.347** | 0.369 | **0.348** | 0.395 | **0.434** |
| DPCCA (Variant B) | **0.679** | 0.339 | 0.371 | 0.339 | **0.415** | 0.430 |
| DCCA_NOI (Wang et al., 2015b) | 0.677 | 0.346 | 0.373 | 0.326 | 0.378 | 0.429 |
| DCCA (Wang et al., 2015a) | 0.639 | 0.253 | 0.365 | 0.264 | 0.333 | 0.413 |
| PCCA (Rao, 1969) | 0.638 | 0.326 | 0.317 | 0.323 | 0.347 | 0.402 |
| CCA (Hotelling, 1936) | 0.623 | 0.314 | 0.312 | 0.336 | 0.355 | 0.402 |
| GCCA (Funaki and Nakayama, 2015) | 0.670 | 0.326 | **0.386** | 0.297 | 0.356 | 0.382 |
| INIT_EMB | 0.582 | 0.160 | 0.306 | 0.251 | 0.356 | 0.382 |

Table 1: Results on EN and IT SimLex-999 (POS-based evaluation). All scores are Spearman's rank correlations. INIT_EMB refers to initial pre-trained monolingual word embeddings.

| Model | English-Russian | | | | | |
|---|---|---|---|---|---|---|
| | EN-Adj | EN-Verbs | EN-Nouns | RU-Adj | RU-Verbs | RU-Nouns |
| DPCCA (Variant A) | 0.658 | 0.322 | 0.370 | **0.446** | **0.375** | **0.434** |
| DPCCA (Variant B) | 0.672 | **0.328** | 0.378 | 0.438 | 0.355 | 0.431 |
| DCCA_NOI (Wang et al., 2015b) | 0.663 | 0.325 | 0.374 | 0.431 | 0.370 | 0.425 |
| DCCA (Wang et al., 2015a) | 0.666 | 0.282 | 0.375 | 0.394 | 0.374 | 0.433 |
| PCCA (Rao, 1969) | 0.634 | 0.284 | 0.331 | 0.371 | 0.333 | 0.409 |
| CCA (Hotelling, 1936) | 0.633 | 0.284 | 0.330 | 0.362 | 0.348 | 0.407 |
| GCCA (Funaki and Nakayama, 2015) | **0.685** | 0.305 | **0.384** | 0.409 | 0.328 | **0.434** |
| INIT_EMB | 0.582 | 0.160 | 0.306 | 0.370 | 0.350 | 0.420 |

Table 2: Results on EN and RU SimLex-999 (POS-based evaluation). All scores are Spearman's rank correlations. INIT_EMB refers to initial pre-trained monolingual word embeddings.

ples also reflect on the difficulty of our sentence-level tasks - the dataset often contains images and descriptions that are very much alike.

As discussed in §7 of the main paper, in our quantitative results we therefore report the retrieval results using two metrics: *R@1* that captures the fraction of correct descriptions retrieved, and *BLUE+1* that captures the number of shared n-grams between the query and the retrieved description.

**Nearest Neighbors Examples** In Figure 3 we explore the monolingual spaces formulated from DPCCA Variant B. We project the vocabulary of the Multilingual SimLex-999 dataset by the learned parameters, and seek for neighbor words in each language. According to the figure, in fact, the generated neighborhoods contain mostly words that are similar, rather than related. Examples for such similarities are: *month, week, year, day* and *weekend* which are all time units; or *song*, *music*, *ballad*, *melody*, and *hymn*, which are all works of music (or a part of it). There are also some examples for words that are more related than similar such as *month* and *calendar*; or *essay* and *literature*. Another interesting example comes from the closest German words for the word *dollar*, which are *meter*, *centimeter* and *second*. Although, not highly similar, our algorithm projected them close to each other, after having the understanding that they are all unit measures of money, length and time.

Finally, our quantitative and qualitative results indeed suggest that our algorithm did a significant progress towards the ability to distinguish between *similarity* to *relatedness*.

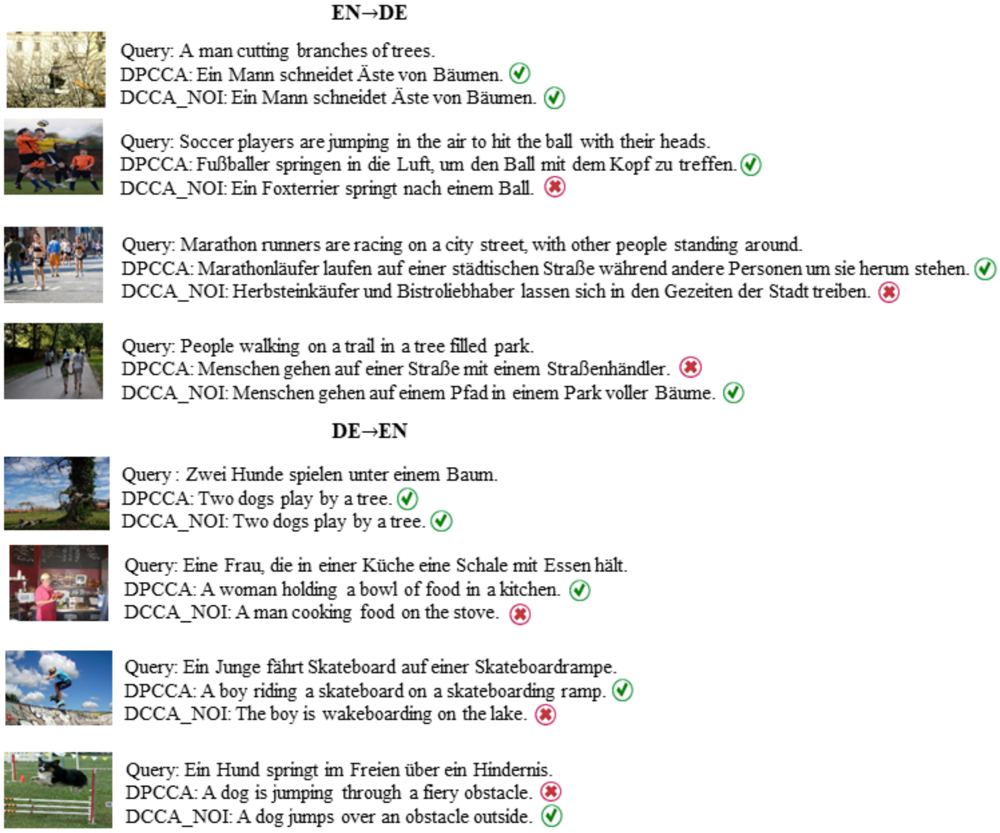


Figure 2: Example results from the cross-lingual image description retrieval task. Every example consists of a query in the target language (Query), the relevant image (left to the text), and the retrieved descriptions of DPCCA Variant B and of DCCA_NOI. A green "v" indicates a correct answer, while a red "x" indicates an incorrect answer.

essay / aufsatz / saggio / эссе

| EN | DE | IT | RU |
|---|---|---|---|
| book | buch | libro | статья |
| article | brief | biografia | роман |
| paper | artikel | argomento | автор |
| biography | bibel | rivista | текст |
| literature | wörterbuch | lettore | книга |

month / monat / mese / месяц

| EN | DE | IT | RU |
|---|---|---|---|
| week | woche | settimana | неделя |
| year | tag | giorno | год |
| day | jahrzehnt | anno | день |
| calendar | jahr | finesettimana | десятилетие |
| weekend | abend | agosto | минута |

song / lied / canzone / песня

| EN | DE | IT | RU |
|---|---|---|---|
| music | ballade | melodia | мелодия |
| ballad | melodie | inno | музыка |
| melody | hymne | ballata | хор |
| hymn | film | musica | гитара |
| guitar | musik | cantante | певец |

dollar / dollar / dollaro / доллар

| EN | DE | IT | RU |
|---|---|---|---|
| cent | cent | economico | цент |
| money | meter | centesimo | метр |
| fee | zentimeter | salario | деньги |
| tax | sekunde | borsa | задолжать |
| salary | mark | reddito | гонорар |

Figure 3: Top 5 nearest neighbors in the Multilingual SimLex-999 dataset according to DPCCA Variant B. Each table is related to one of the four English words (*essay*, *month*, *song* and *dollar*) and their translations to German (DE), Italian (IT) and Russian (RU). For each word in each of the table titles we report the five nearest neighbors in its own language according to cosine similarity between word vectors.

## References


Galen Andrew, Raman Arora, Jeff Bilmes, and Karen Livescu. 2013. Deep canonical correlation analysis. In *Proceedings of ICML*. pages 1247–1255. http://proceedings.mlr.press/v28/andrew13.html.

Xiaobin Chang, Tao Xiang, and Timothy M. Hospedales. 2017. Deep multi-view learning with stochastic decorrelation loss. *CoRR* abs/1707.09669. http://arxiv.org/abs/1707.09669.

Ruka Funaki and Hideki Nakayama. 2015. Image-mediated learning for zero-shot cross-lingual document retrieval. In *Proceedings of EMNLP*. pages 585–590. http://www.aclweb.org/anthology/D15-1070.

Spandana Gella, Rico Sennrich, Frank Keller, and Mirella Lapata. 2017. Image pivoting for learning multilingual multimodal representations. In *Proceedings of EMNLP*. pages 2839–2845. http://www.aclweb.org/anthology/D17-1303.

Paul Horst. 1961. Generalized canonical correlations and their applications to experimental data. *Journal of Clinical Psychology* 17(4):331–347. https://doi.org/10.1002/1097-4679(196110)17:4¡331::AID-JCLP2270170402¿3.0.CO;2-D.

Harold Hotelling. 1936. Relations between two sets of variates. *Biometrika* 28(3/4):321–377. http://www.jstor.org/stable/2333955?seq=1.

Diederik Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. In *Proceedings of ICLR (Conference Track)*. https://arxiv.org/abs/1412.6980.

Omer Levy, Yoav Goldberg, and Ido Dagan. 2015. Improving distributional similarity with lessons learned from word embeddings. *Transactions of the ACL* 3:211–225. http://anthology.aclweb.org/Q/Q15/Q15-1016.pdf.

Tomer Michaeli, Weiran Wang, and Karen Livescu. 2016. Nonparametric canonical correlation analysis. In *Proceedings of ICML*. pages 1967–1976. http://proceedings.mlr.press/v48/michaeli16.html.

Yusuke Mukuta and Harada. 2014. Probabilistic partial canonical correlation analysis. In *Proceedings of ICML*. pages 1449–1457. http://proceedings.mlr.press/v32/mukuta14.html.

Janarthanan Rajendran, Mitesh M. Khapra, Sarath Chandar, and Balaraman Ravindran. 2016. Bridge correlational neural networks for multilingual multimodal representation learning. In *Proceedings of NAACL-HLT*. pages 171–181. http://www.aclweb.org/anthology/N16-1021.

B. Raja Rao. 1969. Partial canonical correlations. *Trabajos de estadistica y de investigación operativa* 20(2-3):211–219. http://link.springer.com/article/10.1007/BF03028532.

Pushpendre Rastogi, Benjamin Van Durme, and Raman Arora. 2015. Multiview LSA: Representation learning via generalized CCA. In *Proceedings of NAACL-HLT*. pages 556–566. http://www.aclweb.org/anthology/N15-1058.

Karen Simonyan and Andrew Zisserman. 2015. Very deep convolutional networks for large-scale image recognition https://arxiv.org/abs/1409.1556.

Ivan Vulić, Roy Schwartz, Ari Rappoport, Roi Reichart, and Anna Korhonen. 2017. Automatic selection of context configurations for improved class-specific word representations. In *Proceedings of CoNLL*. pages 112–122. http://aclweb.org/anthology/K17-1013.

Weiran Wang, Raman Arora, Karen Livescu, and Jeff Bilmes. 2015a. On deep multi-view representation learning. In *Proceedings of ICML*. pages 1083–1092. http://proceedings.mlr.press/v37/wangb15.html.

Weiran Wang, Raman Arora, Karen Livescu, and Nathan Srebro. 2015b. Stochastic optimization for deep CCA via nonlinear orthogonal iterations. In *Proceedings of Communication, Control, and Computing*. pages 688–695. https://doi.org/10.1109/ALLERTON.2015.7447071.