Unsupervised Cross-lingual Representation Learning

July 28, 2019
ACL 2019

Sebastian Ruder

Anders Søgaard

Ivan Vulić

Unsupervised Cross-lingual Representation Learning

Follow along with the tutorial:

- Slides: https://tinyurl.com/xlingual
- Useful repos and resources: listed at the end of the tutorial

Questions:

- Twitter: **#ACLUnsupXL** during the tutorial
- Ask us during the break, after the tutorial, or any time during the conference

Why Cross-Lingual NLP?

Speaking more languages means communicating with more people…
...and reaching more users and customers...

Why Cross-Lingual NLP?

...but there are **more profound** and **democratic** reasons to work in this area:

- decreasing **the digital divide**
- dealing with **inequality of information**
- mitigating **cross-cultural biases**
- deploying language technology for **underrepresented** languages, dialects, minorities; societal impact
- understanding cross-linguistic differences

"95% of all languages in use today will never gain traction online" (Andras Kornai)

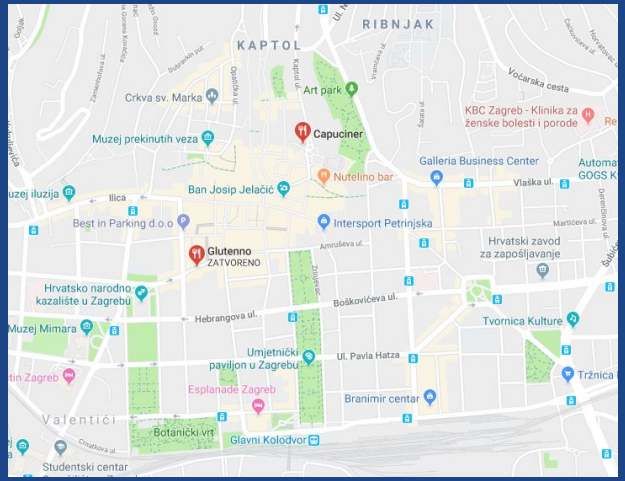
"The limits of my language *online* mean the limits of my world?"

Source: http://labs.theguardian.com/digital-language-divide/

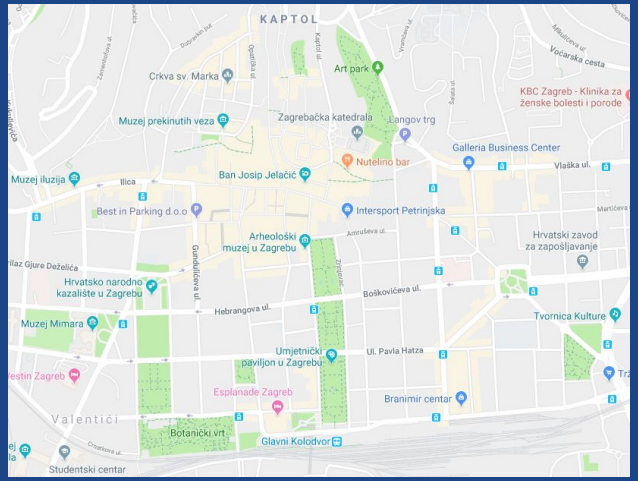
Why Cross-Lingual NLP?

Inequality of information and representation can also affect how we understand places, events, processes...

We're in Zagreb searching for...



...éttermek (HU)



...jatetxe (EU)

...restaurants (EN)

Motivation: Cross-Lingual Representations are Everywhere

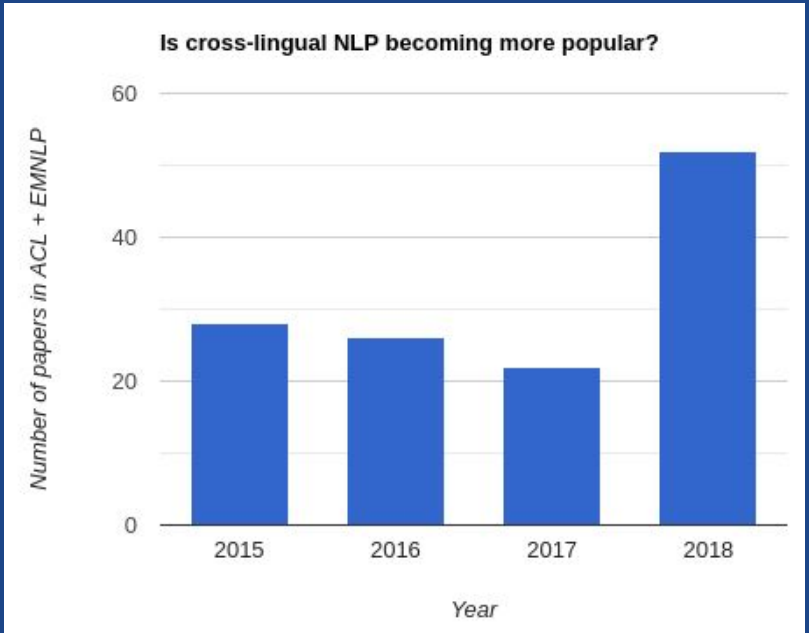
- Cross-lingual and multilingual semantic similarity
- Bilingual lexicon induction, multi-modal representations

- Cross-lingual SRL, POS tagging, NER
- Cross-lingual dependency parsing, sentiment analysis
- Cross-lingual natural language understanding for dialogue
- Cross-lingual lexical entailment
- Cross-lingual annotation and model transfer
- Cross-lingual *you-name-it-task*

- Statistical and neural MT
- Cross-lingual IR and QA

Motivation: Cross-Lingual Representations are Everywhere

Searching for “multilingual”, “cross-lingual” and “bilingual” in the ACL anthology (ACL+EMNLP)



- 10+ papers on unsupervised cross-lingual word embeddings at EMNLP 2018

- The trend continues:
  - 20+ papers on cross-lingual learning and applications at NAACL 2019.

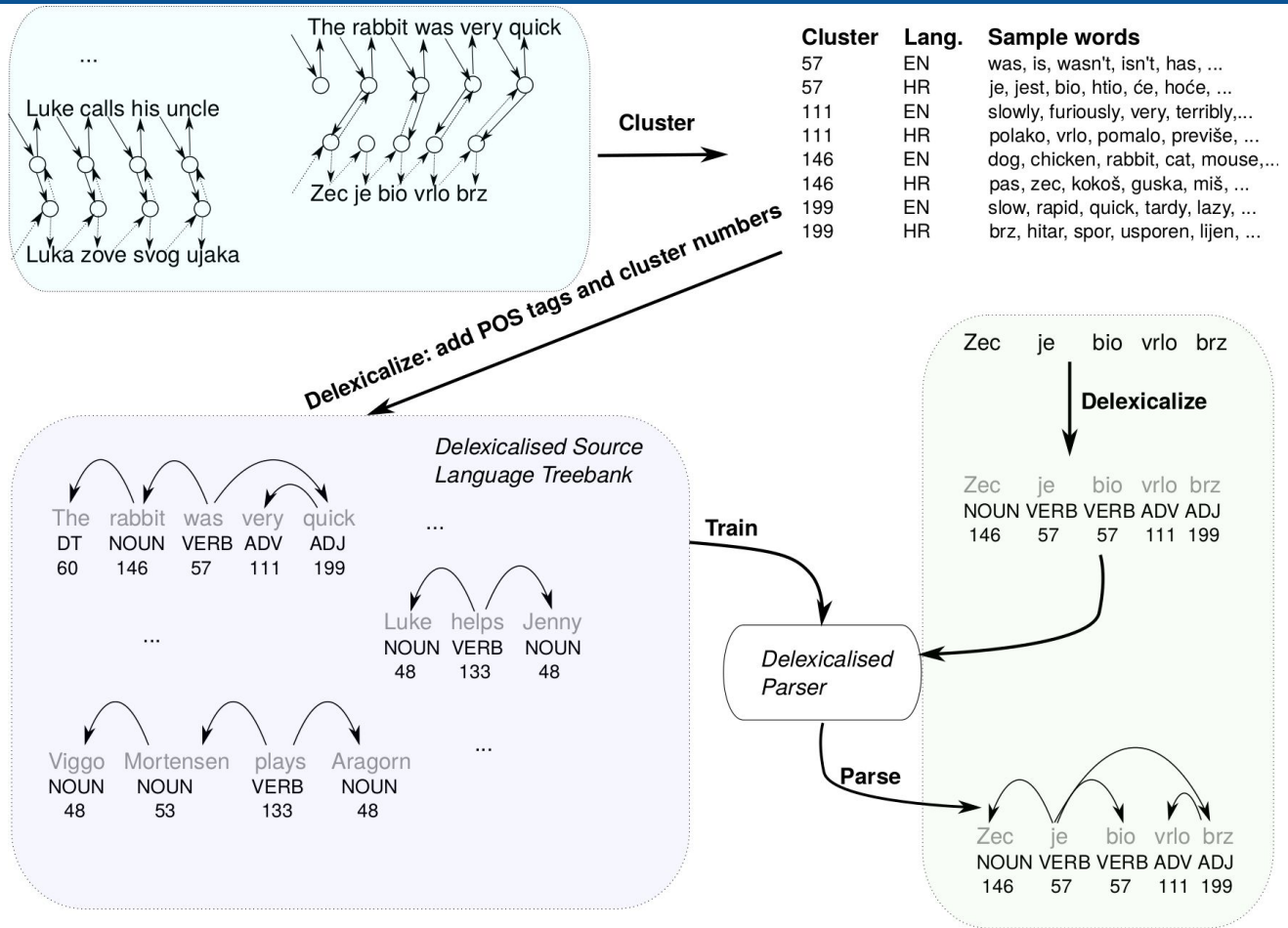
Motivation (Very High-Level)

We want to understand and model the meaning of…

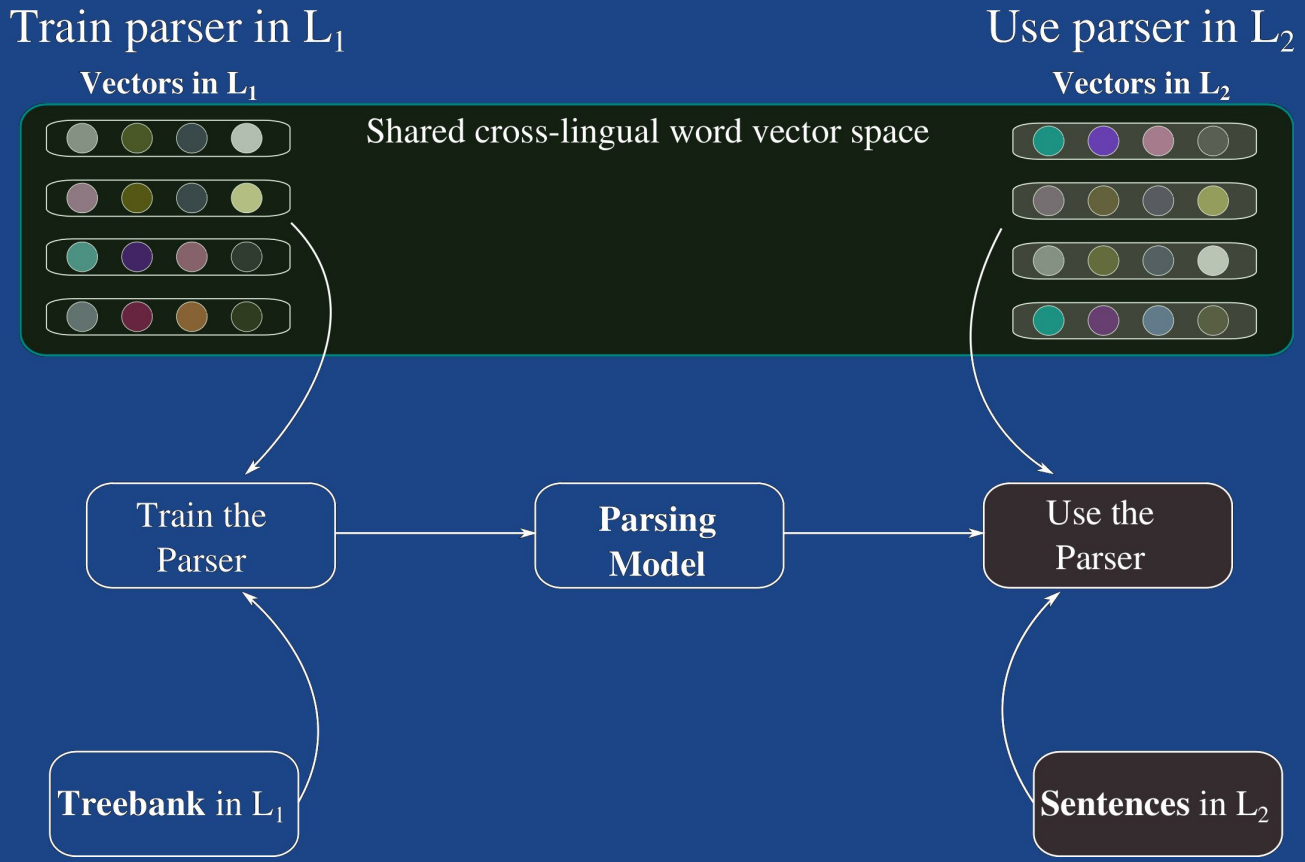
Source: dreamstime.com

…without manual/human input and without perfect MT

The World Existed B.E. (Before Embeddings)

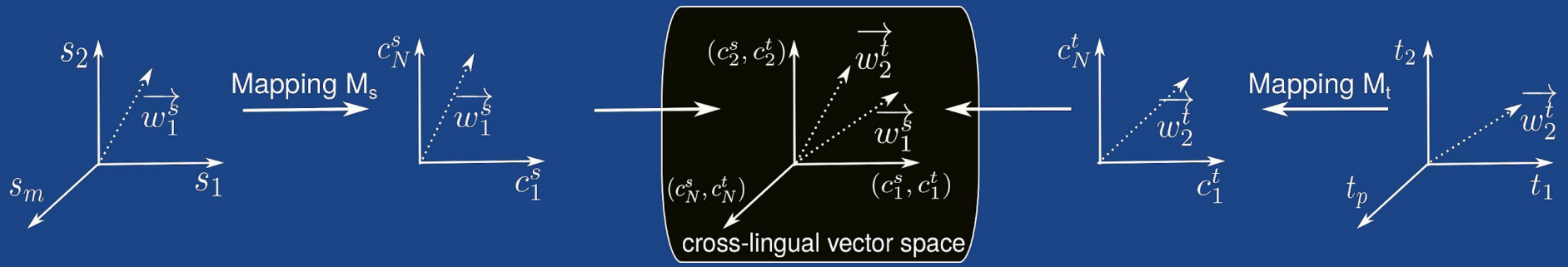


The World Existed B.E.



B.E. Example 1: Cross-lingual (parser) transfer

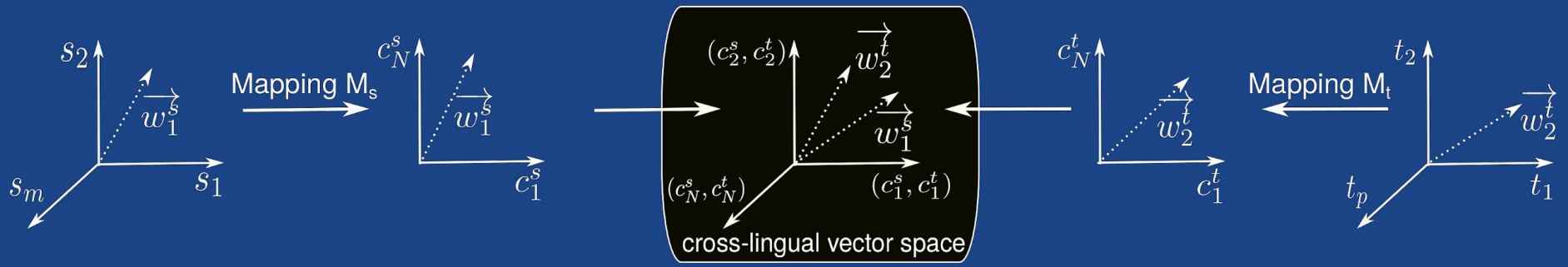
The World Existed B.E.



B.E. Example 2a: Traditional “count-based” cross-lingual vector vector spaces…

[Gaussier et al., ACL 2004; Laroche and Langlais, COLING 2010]

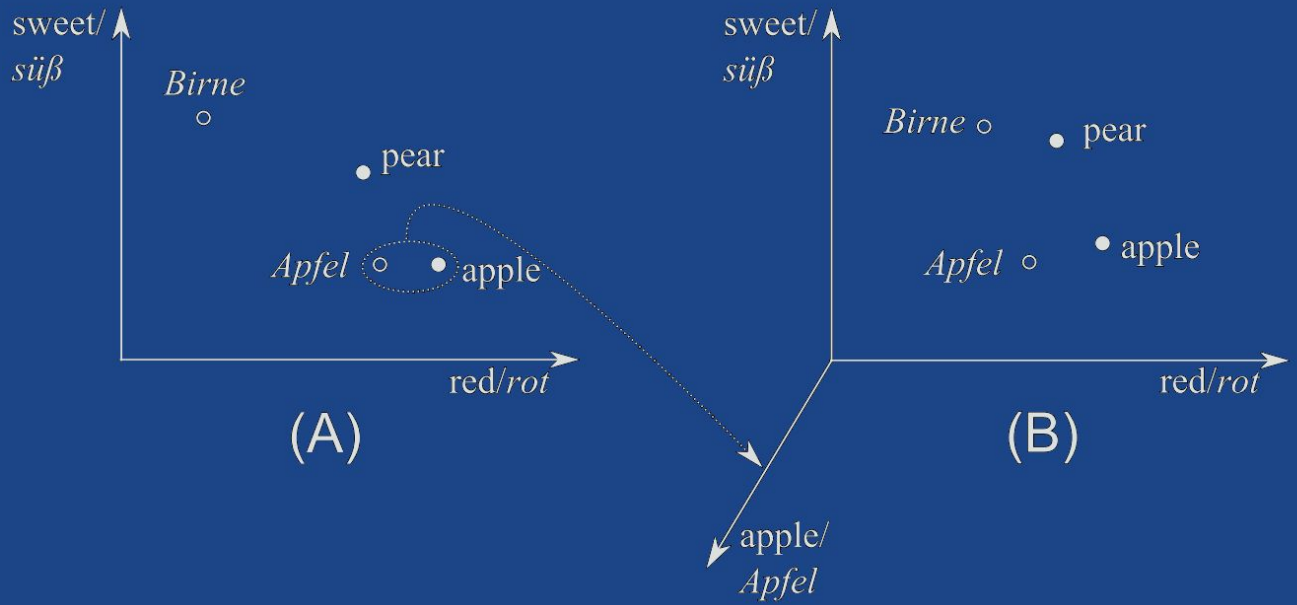
The World Existed B.E.



B.E. Example 2a: Traditional “count-based” cross-lingual vector vector spaces…

[Gaussier et al., ACL 2004; Laroche and Langlais, COLING 2010]

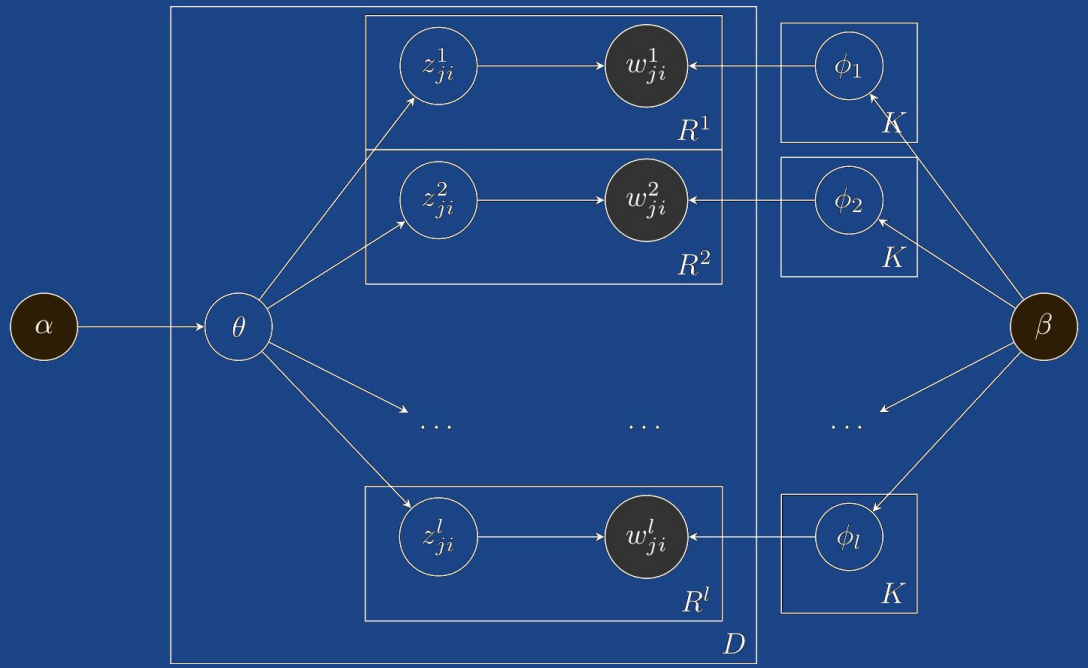
The World Existed B.E.

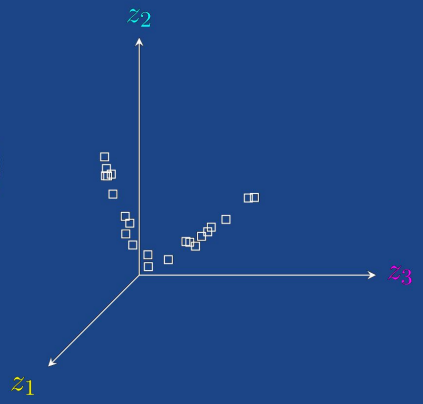


B.E. Example 2b: …and bootstrapping from limited bilingual signal (a sort of *self-learning*)

[Peirsman and Padó, NAACL-10; Vulić and Moens, EMNLP-13]

The World Existed B.E.





	soccer	fútbol
z_1	0.54	0.55
z_2	0.02	0.03
z_3	0.22	0.23

B.E. Example 3: Cross-lingual latent topic spaces

[Mimno et al., EMNLP-09; Vulić et al., ACL-11]

So, Why (Unsupervised) Cross-Lingual Embeddings Exactly?

Cross-lingual word embeddings (CLWE-s)

- Simple: quick and efficient to train
- (Still) state-of-the-art in cross-lingual NLP; omnipresent
- Lightweight and inexpensive
- Multilingual modeling of meaning and support for cross-lingual NLP

Unsupervised CLWE-s:

- Wide portability without bilingual resources?
- Deploying language technology for virtually any language?
- Increasing the ability of cross-lingual transfer?
- An interesting scientific problem still at its infancy:
  Potential for transforming cross-lingual and cross-domain NLP

- **What is the current state-of-the-art in unsupervised cross-lingual representation learning?**

Unsupervised Cross-Lingual Representations

	en-es	es-en	en-fr	fr-en	en-de	de-en	en-ru	ru-en	en-zh	zh-en	en-eo	eo-en
Methods with cross-lingual supervision and fastText embeddings												
Procrustes - NN	77.4	77.3	74.9	76.1	68.4	67.7	47.0	58.2	40.6	30.2	22.1	20.4
Procrustes - ISF	81.1	82.6	81.1	81.3	71.1	71.5	49.5	63.8	35.7	**37.5**	29.0	27.9
Procrustes - CSLS	81.4	82.9	81.1	**82.4**	73.5	**72.4**	**51.7**	**63.7**	**42.7**	36.7	**29.3**	25.3
Methods without cross-lingual supervision and fastText embeddings												
Adv - NN	69.8	71.3	70.4	61.9	63.1	59.6	29.1	41.5	18.5	22.3	13.5	12.1
Adv - CSLS	75.7	79.7	77.8	71.2	70.1	66.4	37.2	48.1	23.4	28.3	18.6	16.6
Adv - Refine - NN	79.1	78.1	78.1	78.2	71.3	69.6	37.3	54.3	30.9	21.9	20.7	20.6
Adv - Refine - CSLS	**81.7**	**83.3**	**82.3**	82.1	**74.0**	72.2	44.0	59.1	32.5	31.4	28.2	**25.6**

best supervised

best unsupervised

Conneau et al. (2018)

- Unsupervised approaches claim performance similar or superior to the best supervised approaches

Unsupervised Cross-Lingual Representations

[Conneau et al.; ICLR 2018]: *"Without using any character information, our model even outperforms existing supervised methods on cross-lingual tasks for some language pairs"*

[Artetxe et al.; ACL 2018]: *"Our method succeeds in all tested scenarios and obtains the best published results in standard datasets, even surpassing previous supervised systems"*

[Hoshen and Wolf; EMNLP 2018]: *"...our method achieves better performance than recent state-of-the-art deep adversarial approaches and is competitive with the supervised baseline"*

[Xu et al.; EMNLP 2018]: *"Our evaluation (...) shows stronger or competitive performance of the proposed method compared to other state-of-the-art supervised and unsupervised methods..."*

[Chen and Cardie; EMNLP 2018]: *"In addition, our model even beats supervised approaches trained with cross-lingual resources."*

So, Why (Unsupervised) Cross-Lingual Embeddings Exactly?

- Supervision is **readily available**
  - Small word translation dictionaries
  - Linguistic resources, e.g. ASJP database (Wichmann et al., 2018) with 40-item word lists in all the world's languages
  - Weak supervision (shared vocabulary)

- Unsupervised representations are **practically justified only** if they outperform their supervised counterparts

How well do unsupervised cross-lingual representations actually perform?

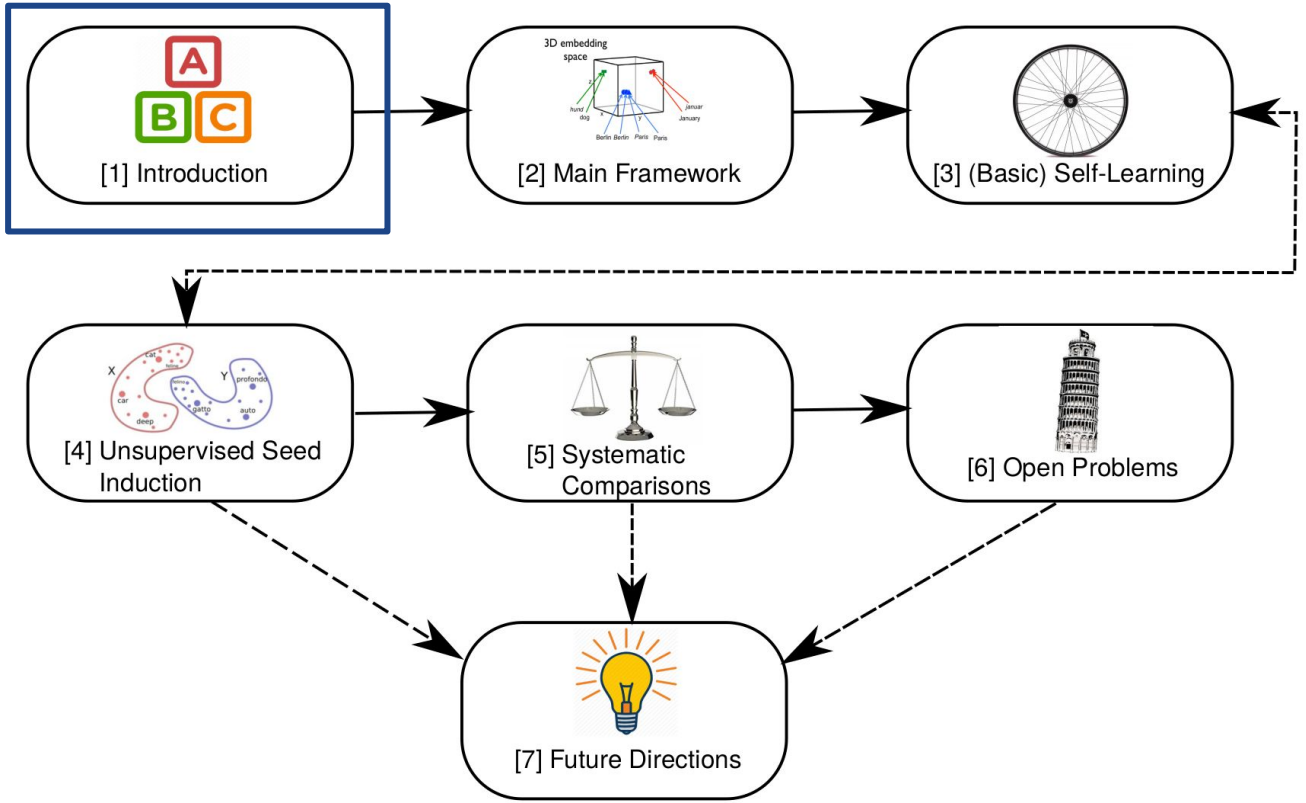
Tutorial Goals

- Provide a systematic overview and typology of unsupervised cross-lingual representation learning models
- Discuss the similarities between current unsupervised approaches
- Analyze the modeling and empirical similarities and differences between fully unsupervised and weakly supervised methods
- Critically examine current limitations, with the focus on (in)stability, robustness, and applicability to distant language pairs and low-data regimes
- Stress the importance of (unsupervised) cross-lingual representations in cross-lingual downstream tasks and applications
- Detect a large number of challenges and open questions for future research

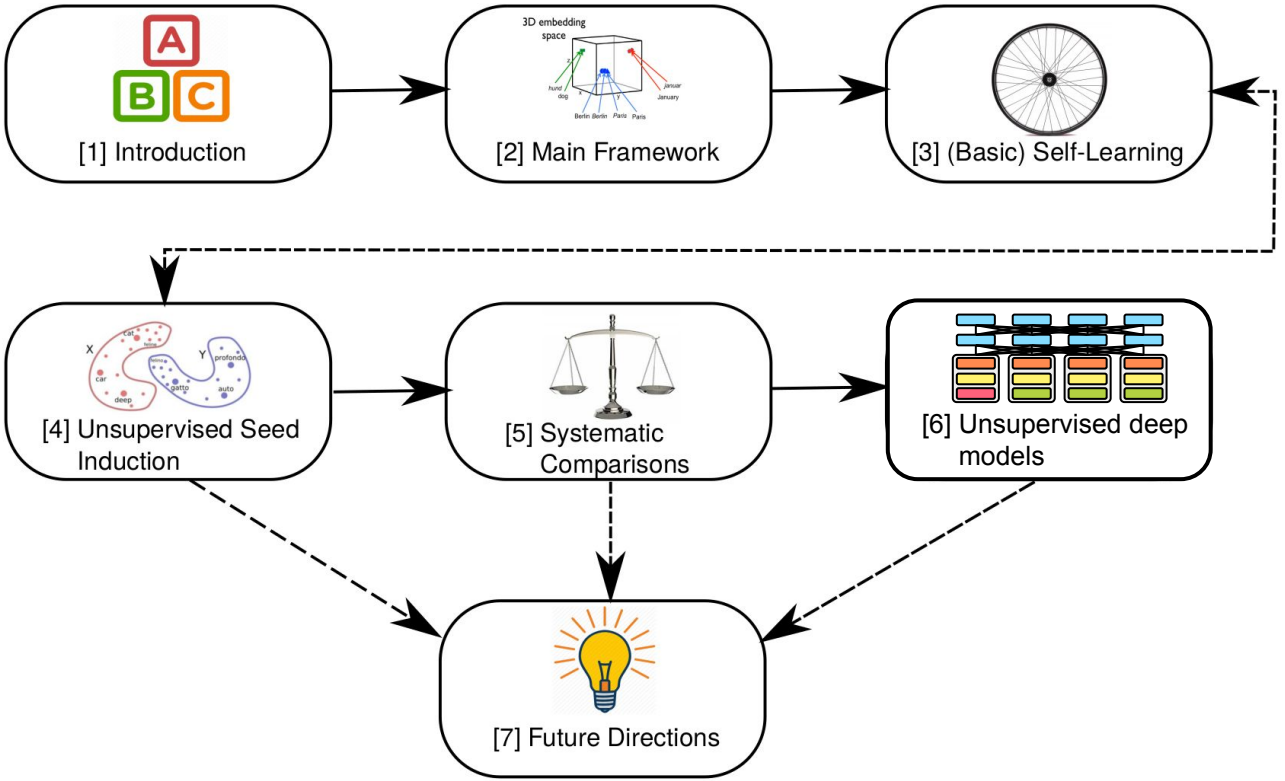
Agenda

What this is not:
Comprehensive. While we'll try to tell a compelling and coherent story from multiple angles, it is impossible to cover all related papers in one tutorial.

(If we haven't mentioned your paper, don't be too mad at us…)



Agenda


[1] Introduction
3D embedding space
[2] Main Framework
[3] (Basic) Self-Learning
[4] Unsupervised Seed Induction
[5] Systematic Comparisons
[6] Unsupervised deep models
[7] Future Directions

Motivation: Crossing the Lexical Chasm

“The(ir) model, however, is only applicable to English, as large enough training sets do not exist for other languages…”

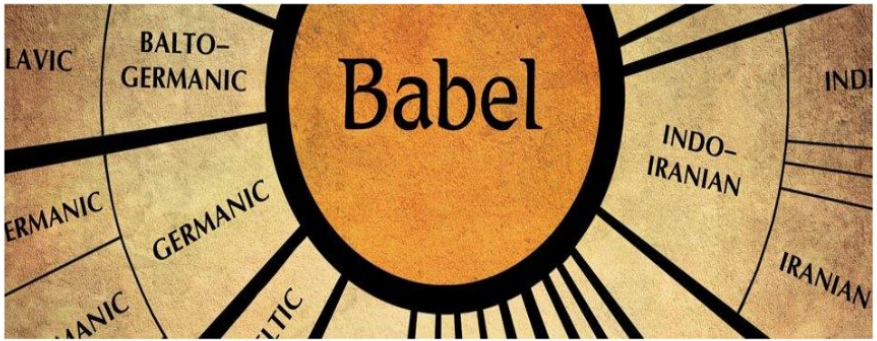
Old paradigm:

- **Language-specific** NLP models
- **Language-specific** feature computation and preprocessing

New paradigm:

- **Representation learning**: inputs are semantic vectors (embeddings)

Multilingual / cross-lingual representation learning



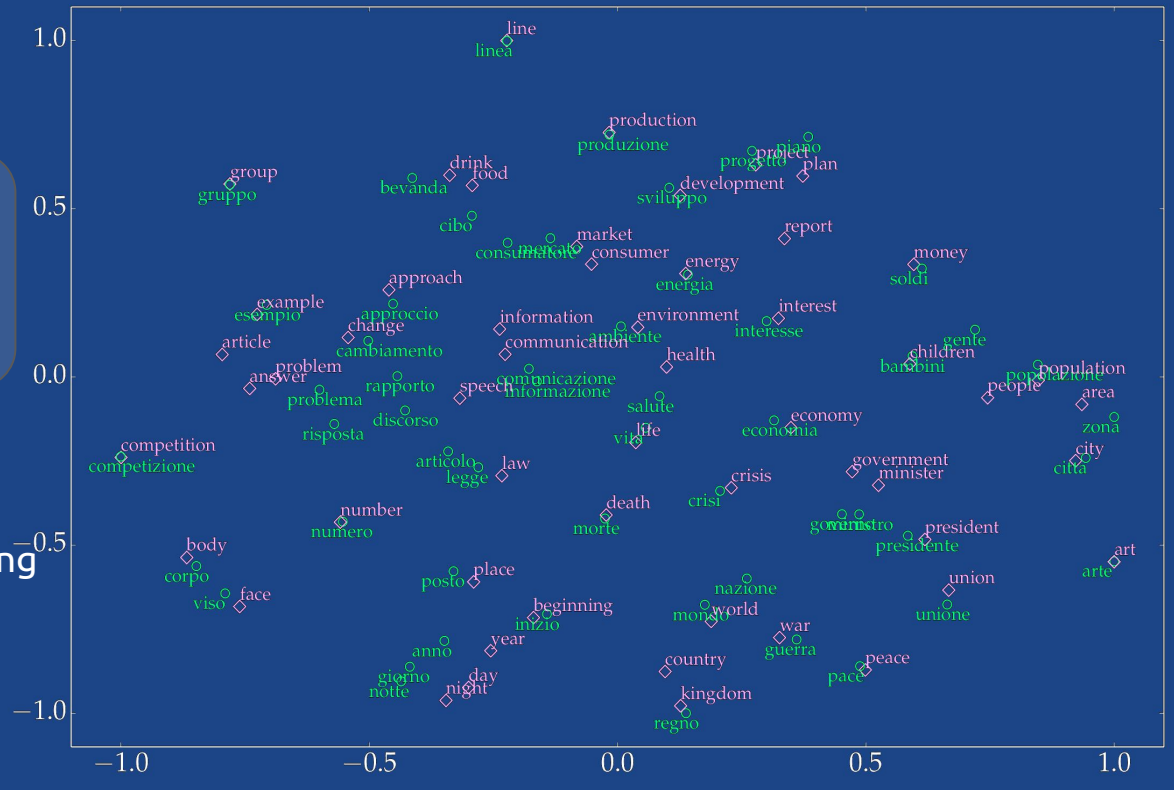
Motivation: Crossing the Lexical Chasm

Multilingual / Cross-lingual representation of meaning

- Word-level
  - Cross-lingual word embeddings
  - Words with similar meanings across languages have similar vectors

- Sentence-/paragraph-level
  - Most recent developments
  - Multilingual unsupervised pretraining [Conneau and Lample, arXiv-19]

- (Unsupervised) NMT?



Recently: Cross-Lingual Language Modeling Pretraining

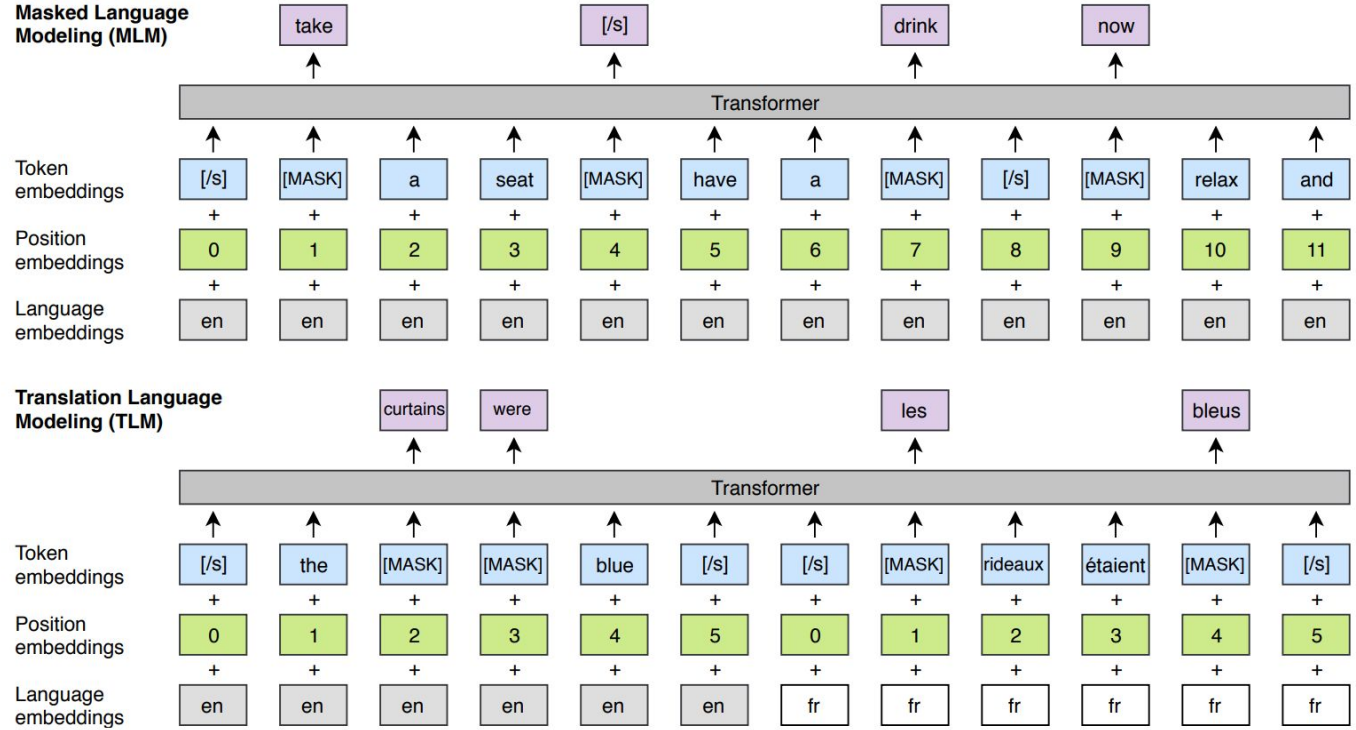


Image from: [Lample and Conneau, arXiv-19]

Cross-lingual word embeddings obtained via the lookup table of a cross-lingual LM

[Artetxe et al., ACL-19]: use UNMT and CLWE induction in an *alternate-and-iterate* fashion

Very recent sub-area of research…

(see also [Pires et al., ACL-19])

Cross-Lingual Word Embeddings

Representation of a word $w_1^S \in V^S$:

$$vec(w_1^S) = [f_1^1, f_2^1, \ldots, f_{dim}^1]$$

Exactly the same representation for $w_2^T \in V^T$:

$$vec(w_2^T) = [f_1^2, f_2^2, \ldots, f_{dim}^2]$$

Language-independent word representations in the same shared semantic (or *embedding*) space!

Why Cross-Lingual Word Representations?

Capturing meaning across languages: a standard task of **bilingual lexicon induction (BLI)**

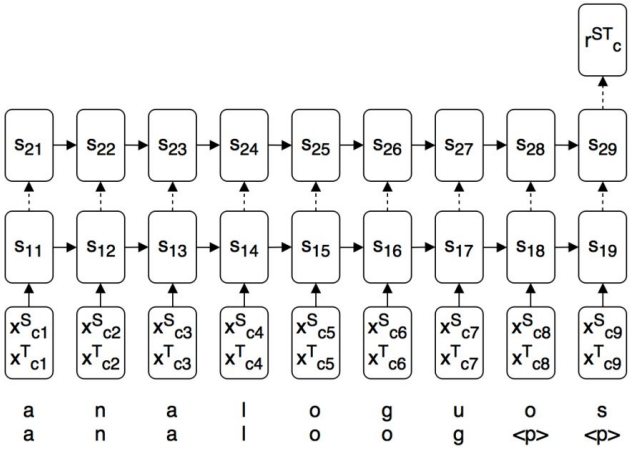
en_morning			en_carpet		
Slavic+EN	**Germanic**	**Romance+EN**	**Slavic+EN**	**Germanic**	**Romance+EN**
en_daybreak	de_vormittag	pt_madrugada	en_rug	de_teppichboden	en_rug
en_morn	nl_krieken	it_mattina	bg_килим	nl_tapijten	it_moquette
bg_разсъмване	en_dawn	en_dawn	ru_ковролин	en_rug	it_tappeti
hr_svitanje	nl_zonsopkomst	pt_madrugadas	bg_килими	de_teppich	pt_tapete
hr_zore	sv_morgonen	es_madrugada	pl_dywany	en_carpeting	es_moqueta
bg_изгрев	de_tagesanbruch	it_nascente	bg_мокет	de_teppiche	it_tappetino
en_dawn	en_sunrise	en_morn	pl_dywanów	sv_mattor	en_carpeting
ru_утро	nl_opgang	es_aurora	hr_tepih	sv_matta	pt_carpete
bg_аврора	de_sonnenaufgang	fr_matin	pl_wykładziny	en_carpets	pt_tapetes
hr_jutro	nl_dageraad	fr_aurora	ru_ковер	nl_tapijt	fr_moquette
ru_рассвет	de_anbruch	es_amaneceres	ru_коврик	nl_kleedje	en_carpets
hr_zora	sv_morgon	en_sunrises	hr_ćilim	nl_vloerbedekking	es_alfombra
hr_zoru	en_daybreak	es_mañanero	en_carpeting	de_brücke	es_alfombras
pl_poranek	de_morgengrauen	fr_matinée	pl_dywan	de_matta	fr_tapis
en_sunrise	nl_zonsopgang	it_mattinata	ru_ковров	nl_matta	pt_tapeçaria
bg_зазоряване	nl_goedemorgen	pt_amanhecer	en_carpets	en_mat	it_zerbino

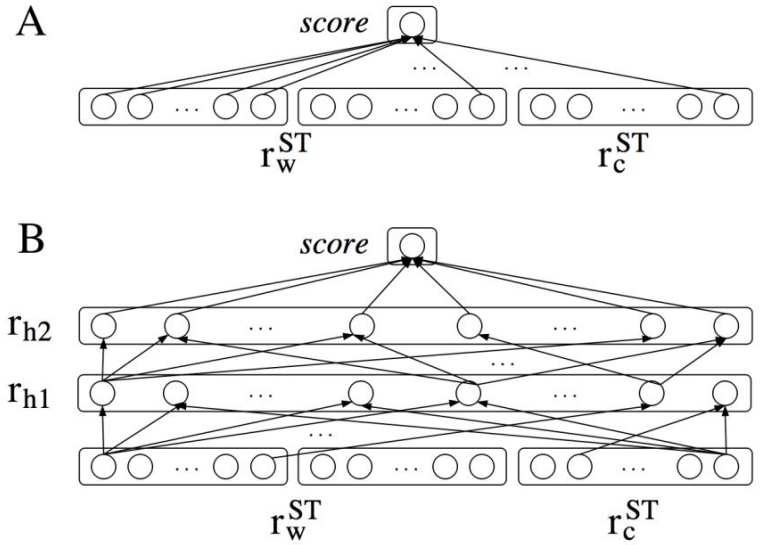
Retrieving nearest neighbours from a shared cross-lingual embedding space (P@1, MRR, MAP)

Similarity-Based vs. Classification-Based BLI

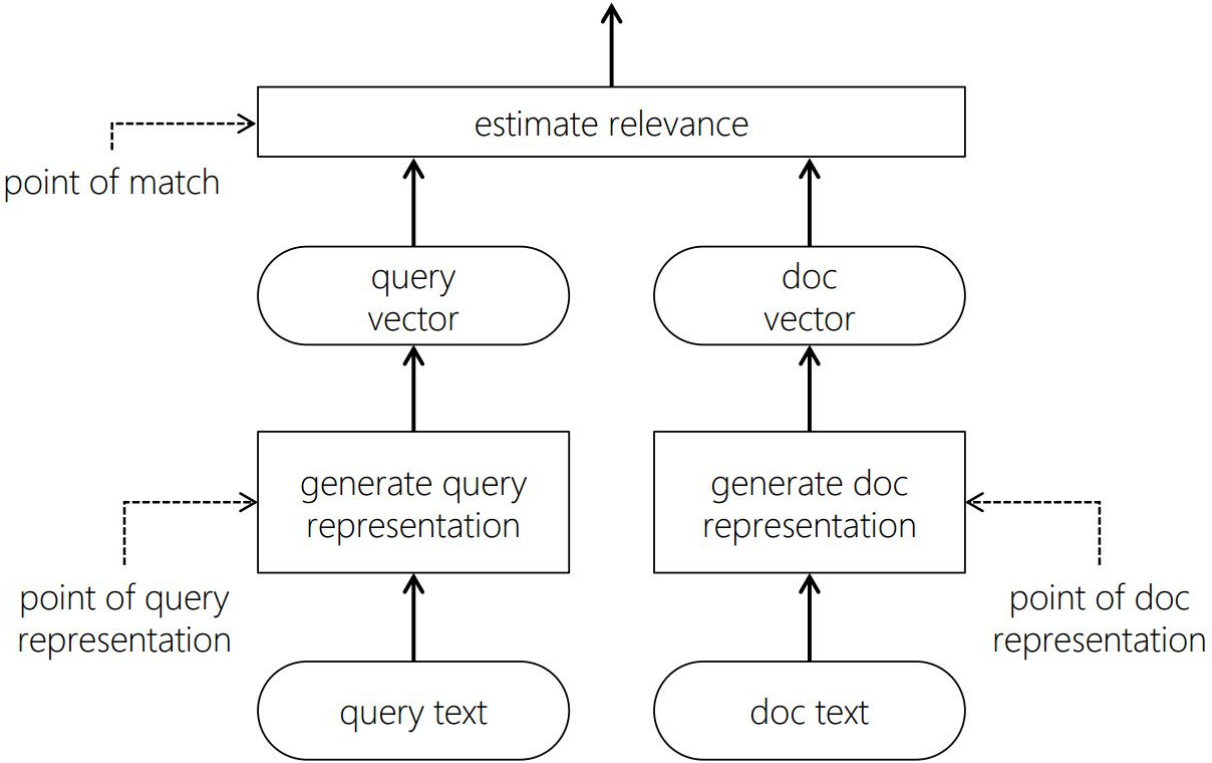
Two modes of how to use CLWE-s: **similarity-based** versus **feature-based**.
Classification-based BLI (combining heterogeneous features)
[Irvine and Callison-Burch, NAACL-13; Heyman et al., EACL-17]

Combining character-level and word-level information with a classifier



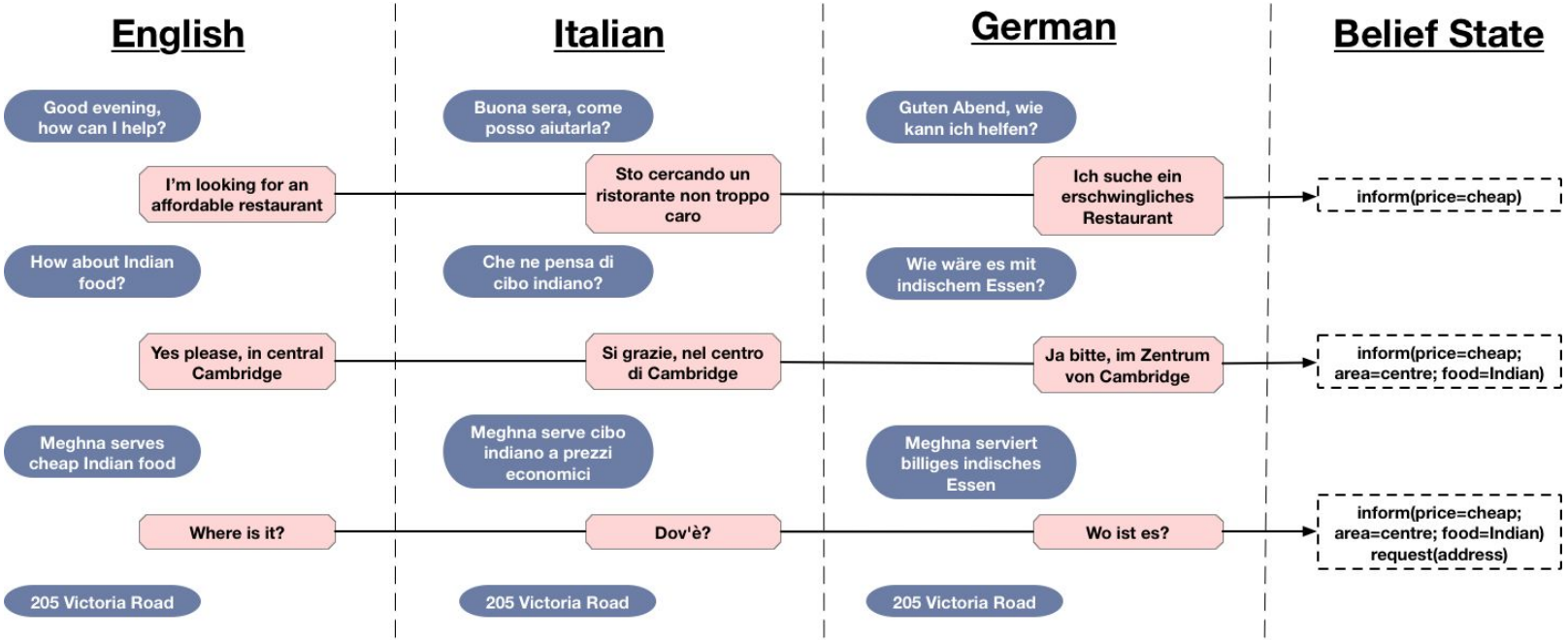


More Applications: Cross-Lingual IR and QA



[Vulić and Moens, SIGIR-15; Mitra and Craswell arXiv-17; Litschko et al., SIGIR-18, SIGIR-19]

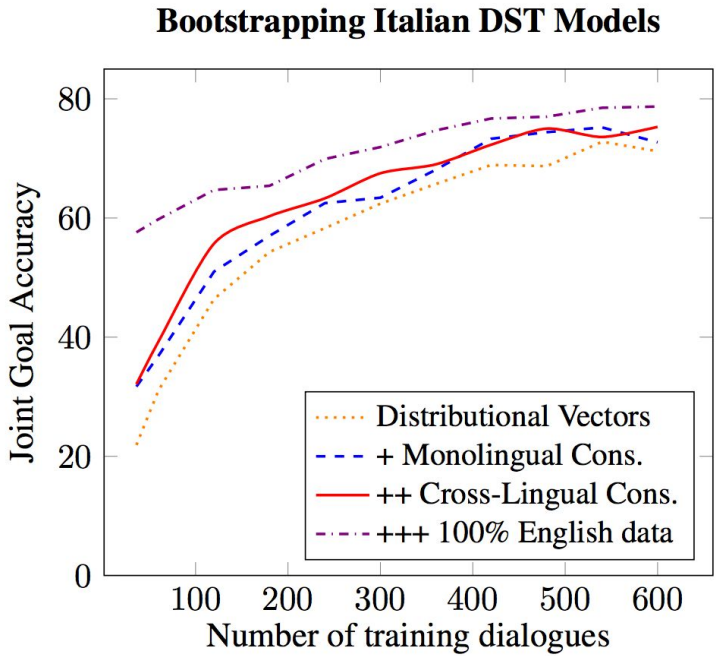
More Applications: Cross-Lingual NLU for Task-Oriented Dialog

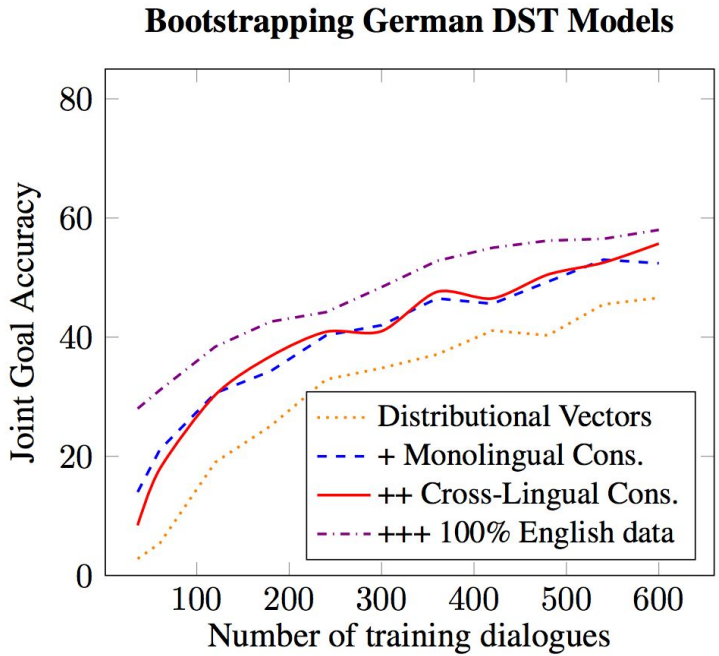


NLU architectures such as NBT [Mrkšić et al., ACL-17; Ramadan et al., ACL-18], GLAD [Zhong et al., ACL-18] or StateNet [Ren et al., EMNLP-18] rely on word embeddings…

- CLWE-s: **use training data from a resource-rich language?**

More Applications: Cross-Lingual NLU for Task-Oriented Dialog





- Some results from [Mrkšić et al., TACL-17]
- CLWE-s are used to leverage additional dialogue training data in a resource-rich source language

Why (Unsupervised) Cross-Lingual Word Representations?

- Recently: **unsupervised neural and statistical machine translation**

[Artetxe et al., ICLR-18, EMNLP-18, ACL-19; Lample et al., ICLR-18, EMNLP-18; Wu et al., NAACL-19;...]

Key component: initialization via unsupervised cross-lingual word embeddings

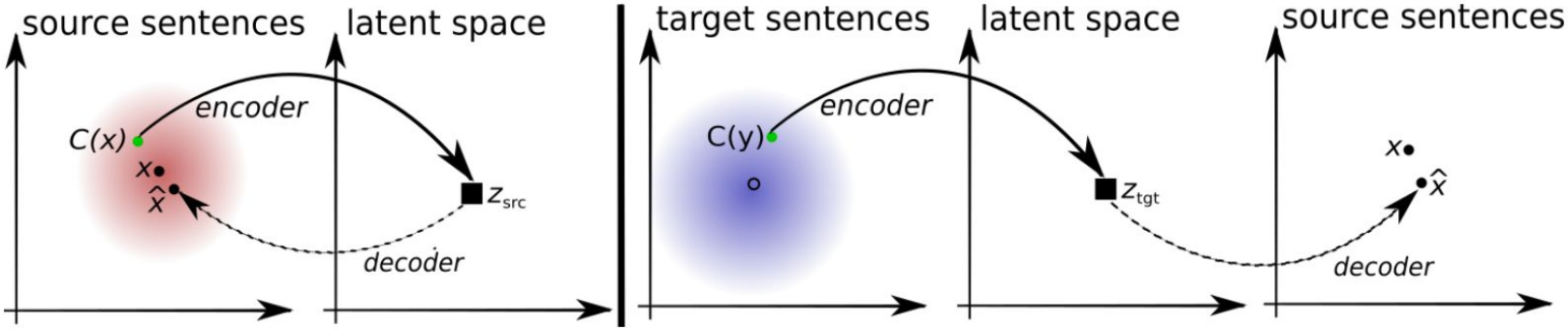


Image from [Lample et al., ICLR-18]

- L1-L1 and L2-L2 reconstruction with denoising autoencoders: C() is added noise
- L1-L2 and L2-L1 reconstruction via a latent space (initialised using CLWE-s)
- Adversarial component: classify between the encoding of source sentences and the encoding of target sentences, i.e., predict the language of the encoded sentence.

Unsupervised MT

- Recently: **unsupervised neural and statistical machine translation**

[Artetxe et al., ICLR-18, EMNLP-18, ACL-19; Lample et al., ICLR-18, EMNLP-18; Wu et al., NAACL-19;...]

Key component: initialization via unsupervised cross-lingual word embeddings

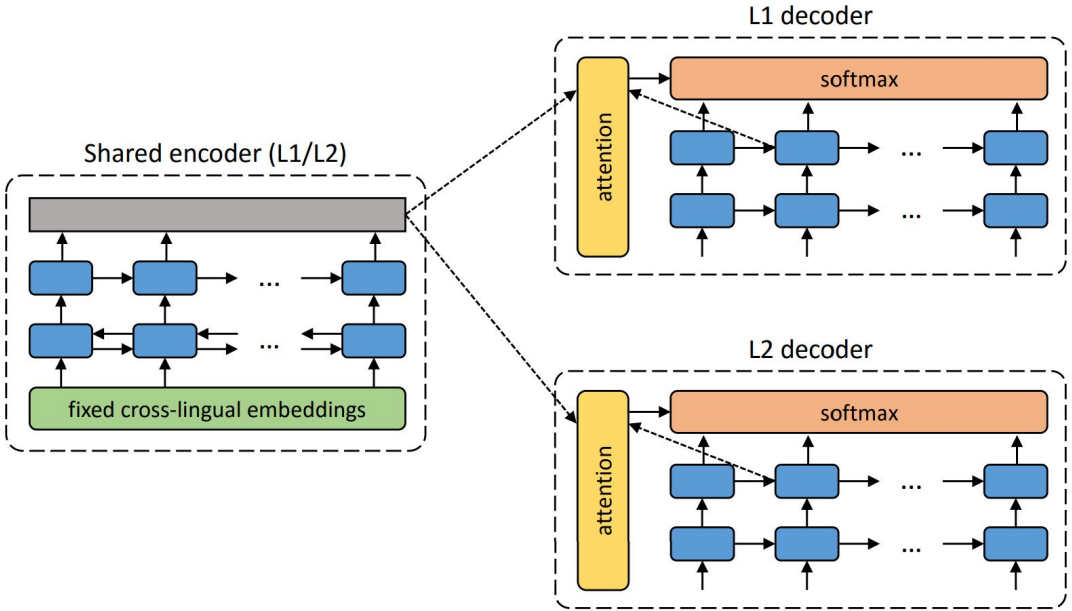


Image from [Artetxe et al., ICLR-18]

Unsupervised MT

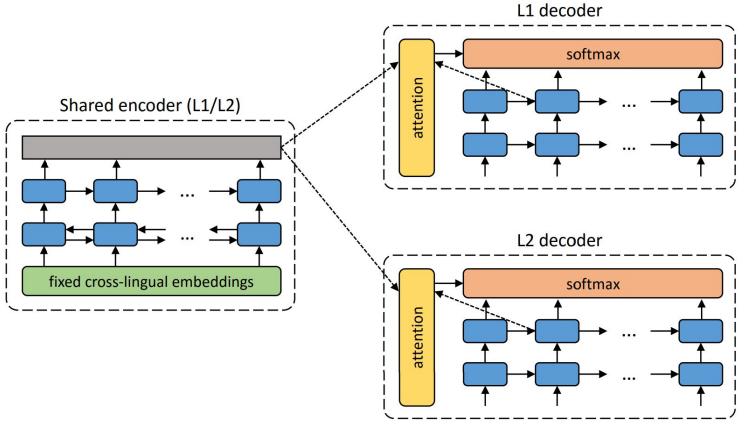


Image from [Artetxe et al., ICLR-18]

Training regime in a nutshell:

- **Denoising autoencoder 1:** noisy input in L1, try to reconstruct the input in the same language (E+L1)
- **Denoising autoencoder 2:** noisy input in L2, try to reconstruct the input in the same language (E+L2)

- **Back-translation:** input in L1, translate E+D2, translate E+D1, output in L1
- **Back-translation**: input in L2, translate E+D1, translate E+D2, output in L2

- Both translation directions handled together
- Shared encoder
- Two decoders for each language
- Embeddings are **fixed**

WMT now has the track on **unsupervised MT!**

Unsupervised MT: Further Improvements

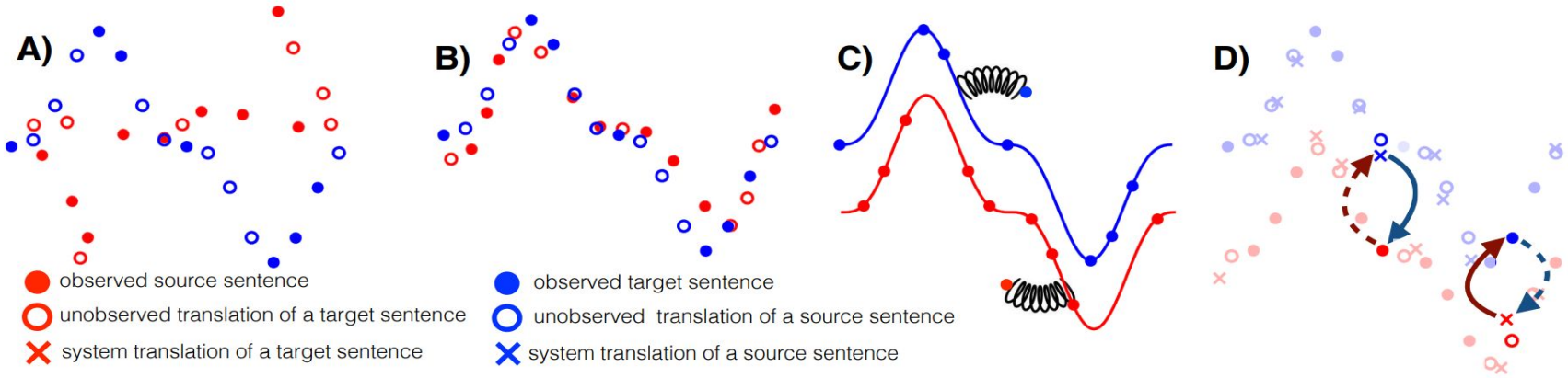


Image and algorithm from [Lample et al., EMNLP-18], a similar idea in [Artetxe et al., EMNLP-18]

Algorithm 1: Unsupervised MT

1. **Language models:** Learn language models P_s and P_t over source and target languages;
2. **Initial translation models:** Leveraging P_s and P_t, learn two initial translation models, one in each direction: $P^{(0)}_{s\to t}$ and $P^{(0)}_{t\to s}$;
3. **for** k=1 **to** N **do**
4. **Back-translation:** Generate source and target sentences using the current translation models, $P^{(k-1)}_{t\to s}$ and $P^{(k-1)}_{s\to t}$, factoring in language models, P_s and P_t;
5. Train new translation models $P^{(k)}_{s\to t}$ and $P^{(k)}_{t\to s}$ using the generated sentences and leveraging P_s and P_t;
6. **end**

NMT:

- Initialisation: CLWE-s directly; LM: denoising autoencoding; Back-translation

PBSMT:

- Phrase-tables generated from CLWE-based bilingual lexicons; n-gram based LM-s; Back-translation

Why (Unsupervised) Cross-Lingual Word Representations?

- Very recently (even going to the future…): “An effective approach to unsupervised MT” [Artetxe et al., ACL-19]

- A set of improvements to their unsupervised MT framework, e.g., using subword-level information, adapting MERT-style training to unsupervised settings: cyclic consistency loss based on the BLEU score plus a language modeling loss (based on n-gram LMs), NMT hybridization, etc.

- The method is competitive to supervised systems from 2014, we’re moving forward so quickly...

Two (and a half) open questions:

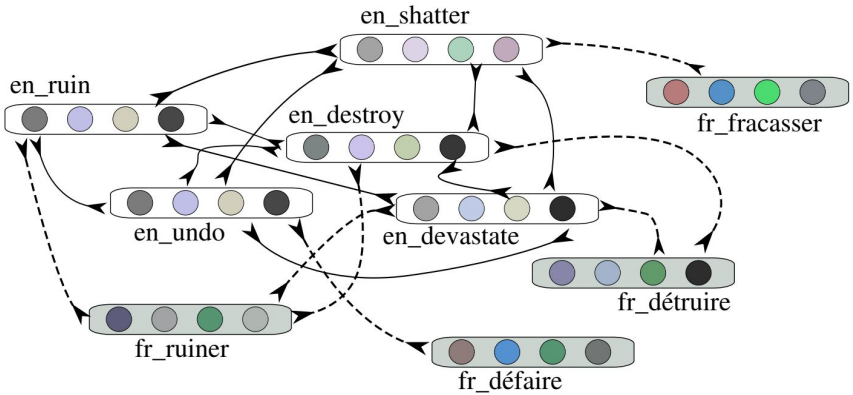
- What about unsupervised MT for truly distant language pairs? *(It seems to work for English-Urdu…)*
- How different (unsupervised) CLWE-s affect unsupervised MT?
  - *How important is the actual initialisation?*

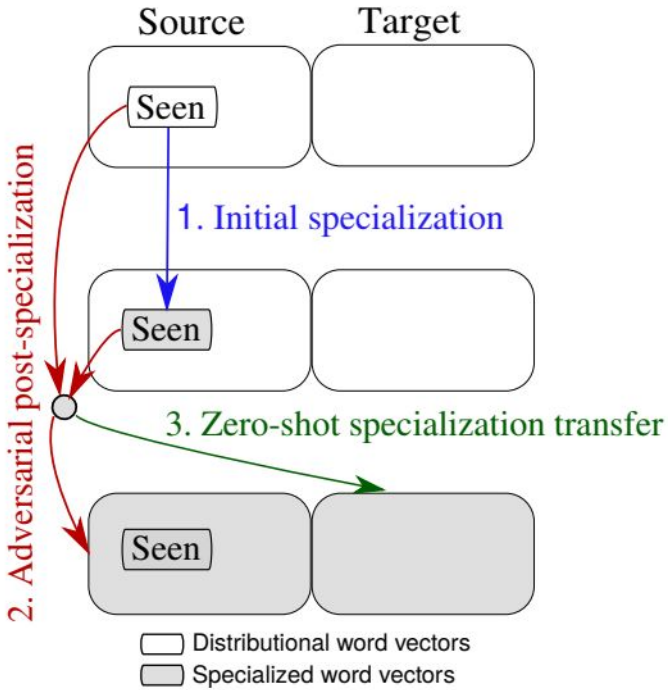
Why (Unsupervised) Cross-Lingual Word Representations?

- Recently: unsupervised cross-lingual transfer of lexical resources

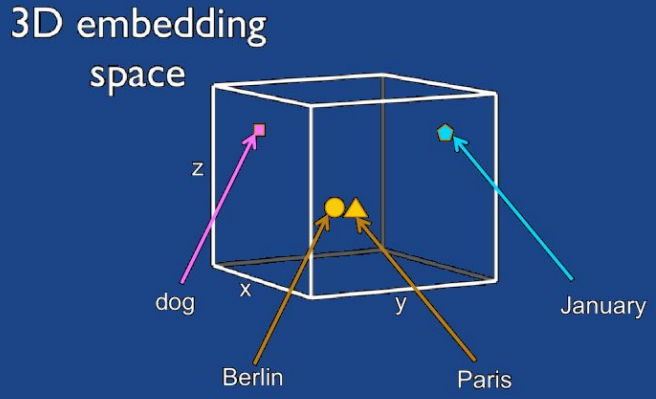
[Ponti et al., EMNLP-18; Glavaš and Vulić, NAACL-18, ACL-18; Jebbara and Cimiano, NAACL-19;...]

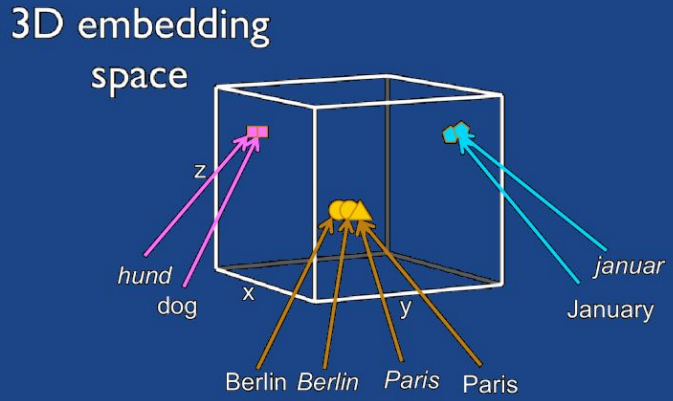
Means of transfer: unsupervised cross-lingual word embeddings





Cross-Lingual Word Embeddings





Monolingual vs. **Cross-lingual**

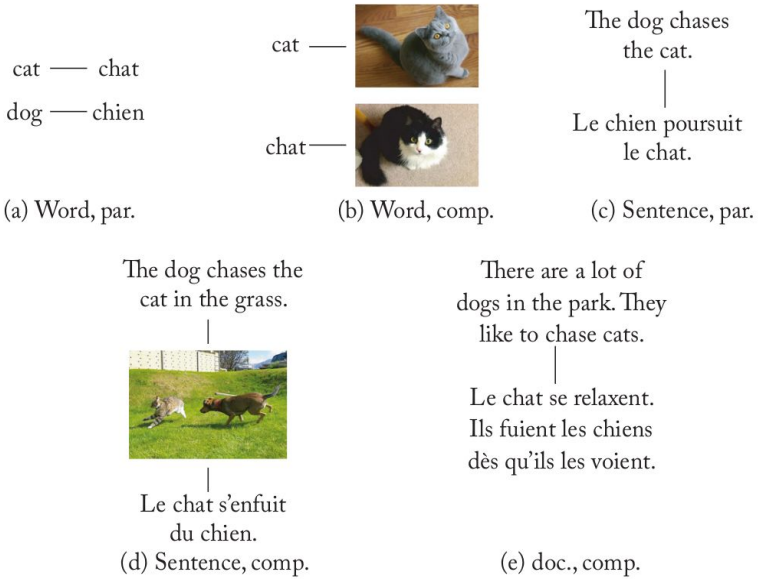
Q1 → Algorithm Design: How to align semantic spaces in two different languages?

Q2 → Data Requirements: Which **bilingual signals** are used for the alignment?

Cross-Lingual Word Embeddings

A large number of different methods, but **the same end goal**:

Induce a shared semantic vector space in which words with similar meaning end up with similar vectors, regardless of their actual language.



We need some bilingual supervision to learn CLWE-s.

Fully unsupervised CLWE-s: they rely only on monolingual data

Cross-Lingual Word Embeddings

A large number of different methods, but **the same end goal**:

Induce a shared semantic vector space in which words with similar meaning end up with similar vectors, regardless of their actual language.

Typology of methods for inducing CLWE-s [Ruder et al., JAIR-19; Søgaard et al., M&C Book]

- **Type of bilingual signal**
  - Document-level, sentence-level, **word-level, no signal (i.e., unsupervised)**
- **Comparability**
  - Parallel texts, comparable texts, non-comparable
- **Point/Time of alignment**
  - Joint embedding models vs. **Post-hoc alignment** vs. post-specialisation/retrofitting
- **Modality**
  - Text only vs. using images for alignment, e.g., [Kiela et al., EMNLP-15; Vulić et al., ACL-16; Gella et al., EMNLP-17]

General (Simplified) CLWE Methodology

- Previously: (bilingual) data sources seem more important than the chosen algorithm [Levy et al, EACL-17]
- Most CLWE algorithms are formulated as: $\mathcal{L}_S(\mathbf{X}_S) + \mathcal{L}_T(\mathbf{X}_T) + \Omega(\mathbf{X}_S, \mathbf{X}_T)$

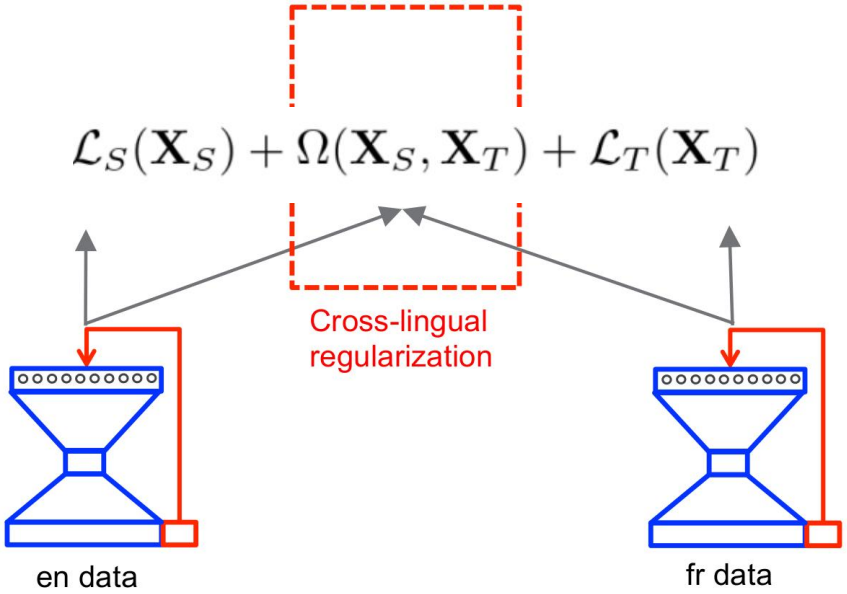
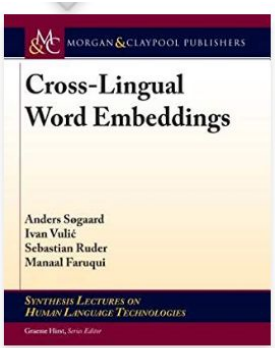


Image adapted from [Gouws et al., ICML-15]

Joint CWLE Models *(selection)*

- **Using word-level cross-lingual signal: word alignments, bilingual dictionaries...**
  - Bilingual extensions of the monolingual skip-gram and CBOW models
    [Ammar et al., arXiv-15; Luong et al., NAACL-15; Guo et al., ACL-15; Shi et al., ACL-15]
  - Creating pseudo-bilingual corpora + training monolingual WE methods on such corpora
    [Gouws and Søgaard, NAACL-15; Duong et al., EMNLP-16; Adams et al., EACL-17]
- **Using sentence-level cross-lingual signal**
  - Compositional sentence model [Hermann and Blunsom, ACL-14]
  - Sentence-level bilingual skip-gram [Coulmance et al., EMNLP-15; Gouws et al., ICML-15]
  - Bilingual sentence autoencoders [Chandar et al., NeurIPS-14]
- **Using document-level cross-lingual signal**
  [Vulić and Moens, ACL-15, JAIR-16; Søgaard et al., ACL-15]

A Commercial Break



Want to Read

Rate this book

Cross-Lingual Word Embeddings

by Anders Sogaard, Ivan Vulic, Sebastian Ruder, Manaal Faruqui, Graeme Hirst (Editor)

0.00 · Rating details · 0 ratings · 0 reviews

The majority of natural language processing (NLP) is English language processing, and while there is good language technology support for (standard varieties of) English, support for Albanian, Burmese, or Cebuano--and most other languages--remains limited.

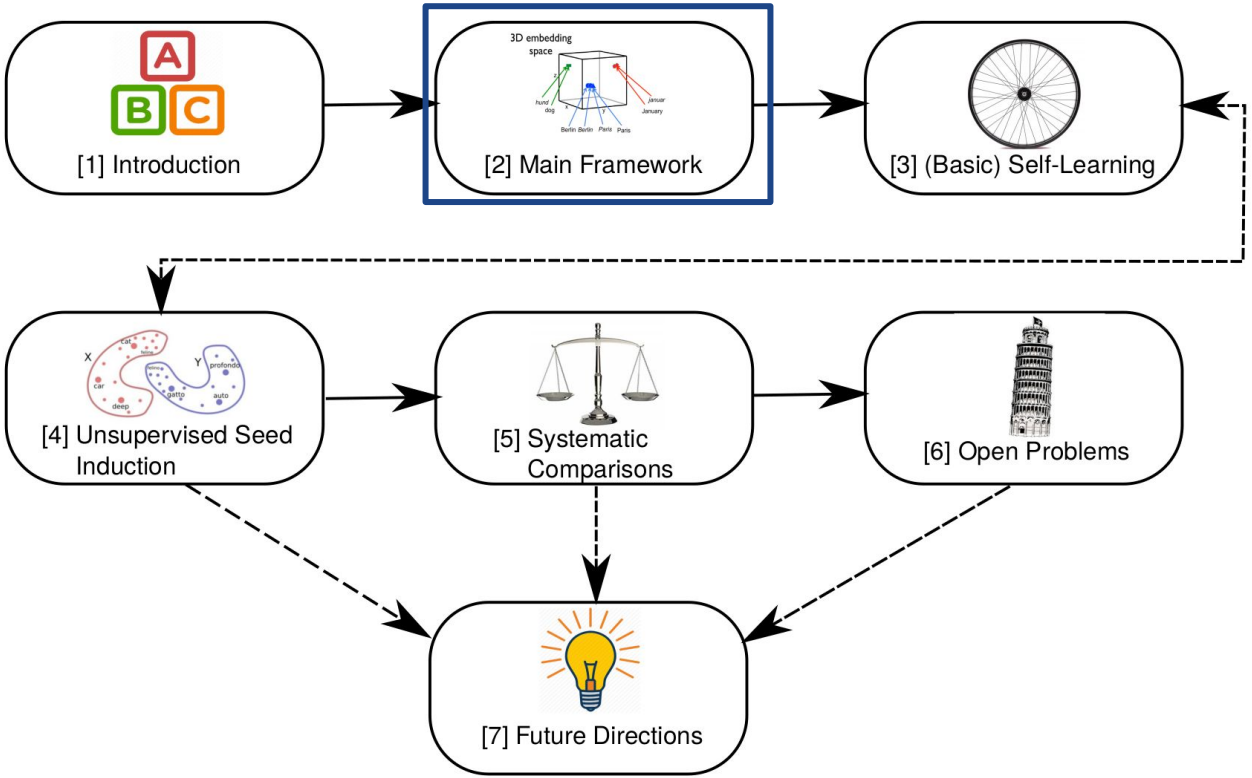
Being able to bridge this digital divide is important for scientific and democratic reasons but also represents an enor ...more

Book published in June 2019

It covers a lot of material we just skipped…

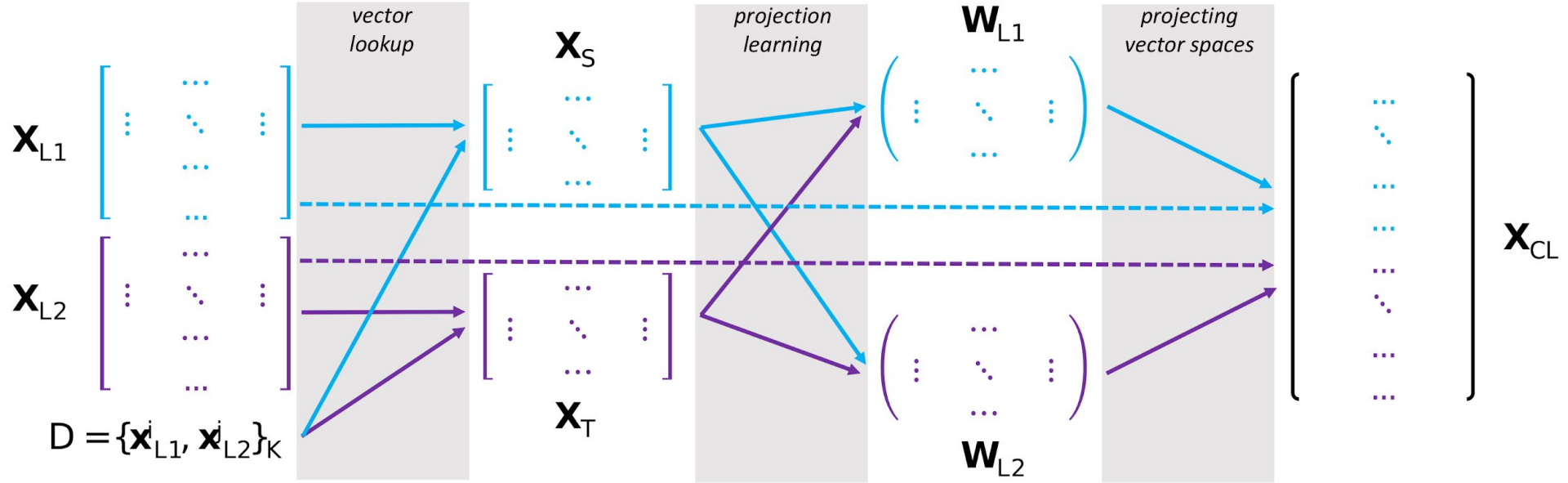
(An older overview is also in the EMNLP-17 tutorial)

2. Main Framework



Projection-based CLWE Learning

[Glavaš et al.; ACL-19]



Post-hoc alignment of independently trained monolingual distributional word vectors

Alignment based on **word translation pairs** (dictionary **D**)

Projection-based CLWE Learning

Most models learn a single projection matrix $\mathbf{W}_{L1}$ (i..e., $\mathbf{W}_{L2} = \mathbf{I}$), but **bidirectional learning** is also common.

$$\begin{array}{r} \\ \text{bird} \\ \text{pretty} \\ \dots \\ \text{eat} \end{array} \begin{bmatrix} -1.18 & 0.21 & \dots & 0.11 \\ 0.23 & -0.53 & \dots & 0.34 \\ \dots & \dots & \dots & \dots \\ 0.78 & 1.33 & \dots & -0.47 \end{bmatrix} \mathbf{W}_{L1} = \begin{bmatrix} 0.59 & 1.01 & \dots & 0.37 \\ -0.34 & -0.27 & \dots & 0.41 \\ \dots & \dots & \dots & \dots \\ 0.81 & -0.31 & \dots & 0.29 \end{bmatrix} \begin{array}{l} \mathbf{X}_T \\ \text{Vogel} \\ \text{schön} \\ \dots \\ \text{essen} \end{array}$$

$\mathbf{X}_S$

How do we find the "optimal" projection matrix $\mathbf{W}_{L1}$?

- **Mean square error:** [Mikolov et al., arXiv-13] and most follow-up work
  ...except…
- **Canonical methods** [Faruqui et al., EACL-14; Lu et al., NAACL-15; Rotman et al., ACL-18]
- **Max-margin framework:** [Lazaridou et al., ACL-15; Mrkšić et al., TACL-17]
- **Relaxed Cross-Domain Similarity Local Scaling**: [Joulin et al., EMNLP-18]

Minimising Euclidean Distance

[Mikolov et al., arXiv-13] minimize the Euclidean distances for translation pairs after projection

$$\mathbf{W}_{L1} = \arg\min_{\mathbf{W}} \| \mathbf{X}_{\mathrm{S}} \mathbf{W} - \mathbf{X}_{\mathrm{T}} \|_2$$

The optimisation problem has no closed-form solution

- Iterative SGD-based optimisation was used initially

More complex mappings: e.g., non-linear DFFNs instead of linear projection matrix yield worse performance

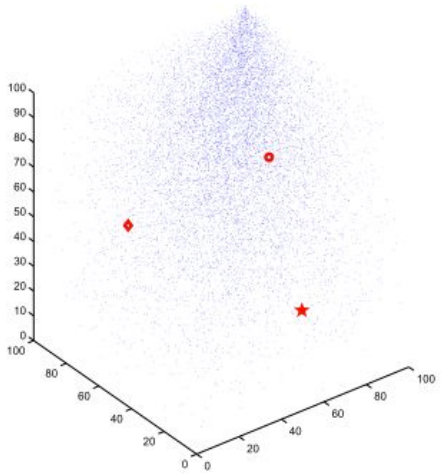
Better (word translation) results when $\mathbf{W}_{L1}$ is constrained to be **orthogonal**

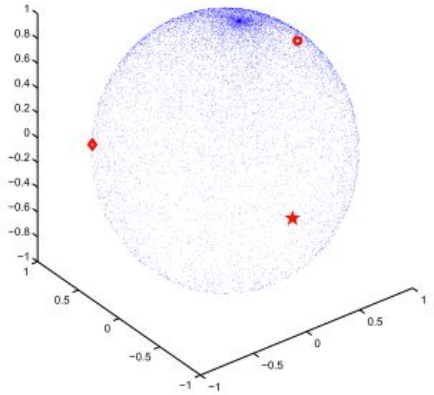
- This preserves monolingual vector space topology

Minimising Euclidean Distance or Cosine?

[Xing et al., NAACL-15]: there is a mismatch between the initial objective function, the distance measure, and the transformation objective

Solution: 1. Normalization of word vectors to unit length + 2. Replacing mean-square error with cosine similarity for learning the mapping





$$\max_{W} \sum_{i=1}^{n} cos(Wx_S^i, x_T^i)$$

Orthogonal Mapping: Solving the Procrustes Problem

If **W** is orthogonal, the optimisation problem is the so-called **Procrustes** problem:

- It has a closed form solution [Schönemann, 1966; Artetxe et al., EMNLP-16]

$$\mathbf{W}_{L1} = \mathbf{U}\mathbf{V}^\top, \text{ with}$$
$$\mathbf{U}\mathbf{\Sigma}\mathbf{V}^\top = SVD(\mathbf{X}_T\mathbf{X}_S{}^\top)$$

Important! Almost all projection-based CLWE methods, supervised and unsupervised alike, solve the Procrustes problem in the final step or during self-learning...

What is Hubness?

Hubness: the tendency of some vectors (i.e., “hubs”) to appear in the ranked lists of nearest neighbours of many other vectors in high-dimensional spaces
[Radovanović et al., JMLR-10; Dinu et al., ICLR-15]

Solutions:

- **Globally-corrected retrieval:** : instead of returning the nearest neighbour of the query, it returns the target element which has the query ranked highest
  [Dinu et al., ICLR-15; Smith et al., ICLR-17]

- A **margin-based ranking loss** instead of mean-squared error
  [Lazaridou et al., ACL-15]

- Scaled similarity measures instead of cosine: discounting similarity in dense areas: **CSLS**
  [Conneau et al., ICLR-18]

Relaxed Cross-Domain Similarity Local Scaling (RCSLS)

If our goal is to optimise for the word translation performance…

Improved word translation retrieval if the retrieval procedure is corrected for **hubness** [Radovanović et al., JMLR-10; Dinu et al., ICLR-15]

Cross-domain similarity local scaling (CSLS) proposed by [Conneau et al., ICLR-18]

$$\mathrm{CSLS}(Wx_s, y_t) = 2\cos(Wx_s, y_t) - r_{\mathrm{T}}(Wx_s) - r_{\mathrm{S}}(y_t)$$

$$r_{\mathrm{T}}(Wx_s) = \frac{1}{K} \sum_{y_t \in \mathcal{N}_{\mathrm{T}}(Wx_s)} \cos(Wx_s, y_t),$$

[Joulin et al., EMNLP-18] maximise CSLS (after projection) instead of minimising Euclidean distance; they relax the orthogonality constraint: **Relaxed CSLS**

How Important is (the Noise in the) the Seed Lexicon?

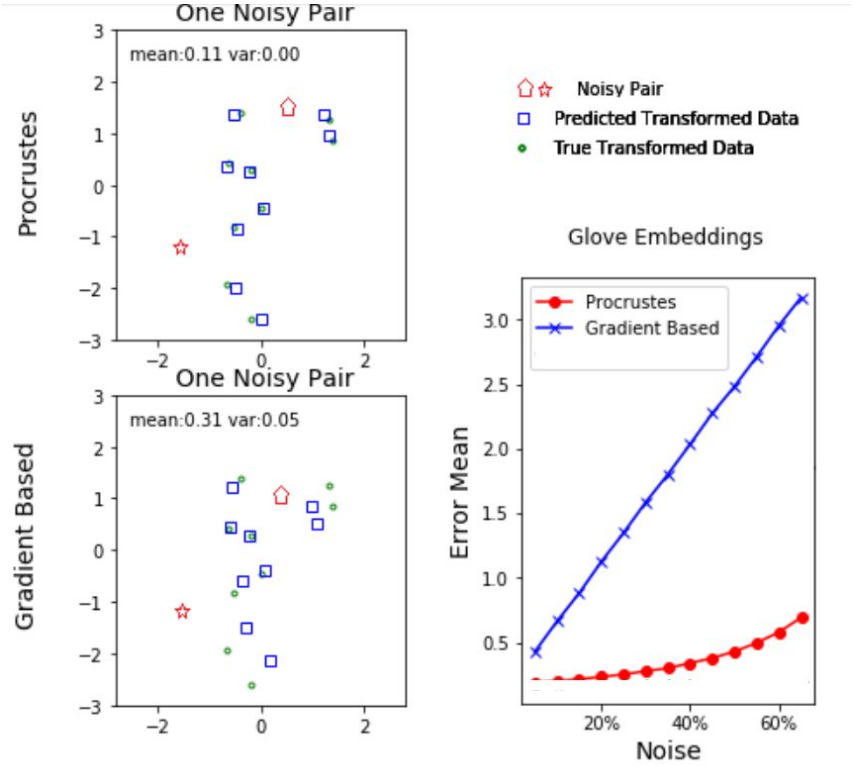


Image adapted from [Lubin et al., NAACL-19]

Orthogonal Procrustes is less affected by noise (i.e., incorrect translation pairs in the training set)

Open question: how to reduce the noise in seed lexicons?

Are larger seed lexicons necessarily better seed lexicons?

How Important is (the Size of) the Seed Lexicon?

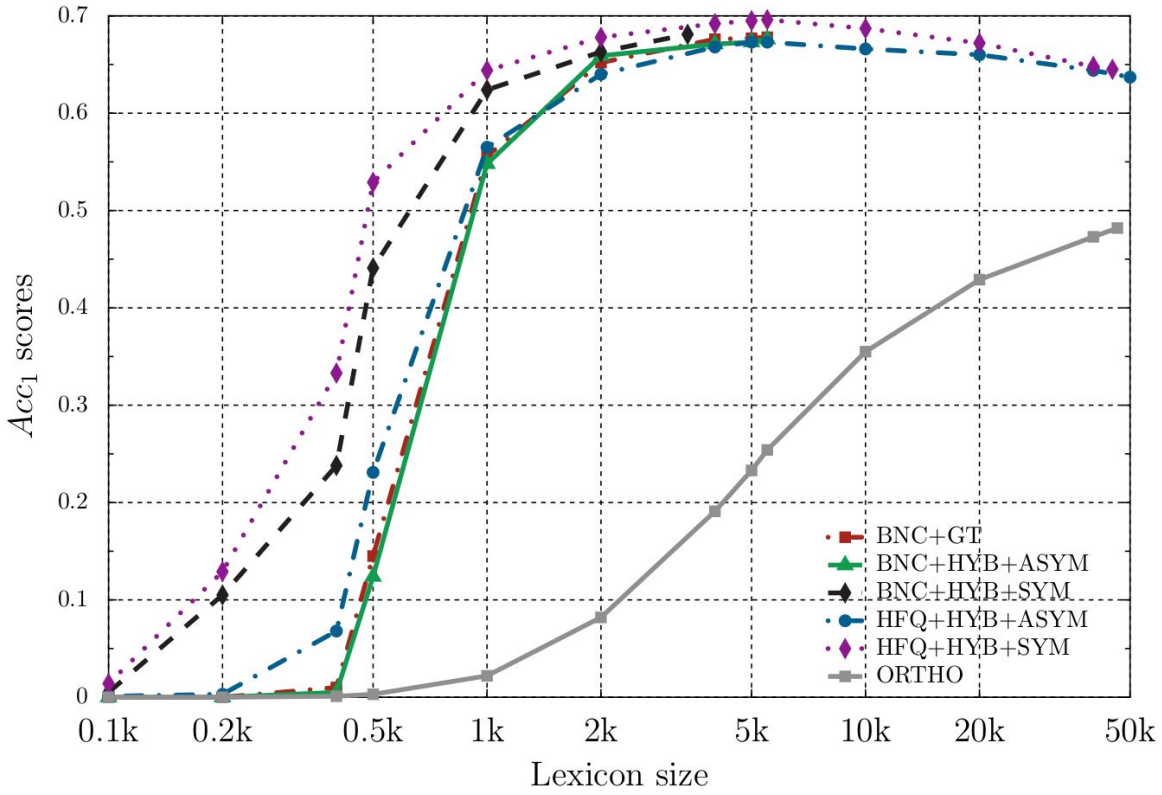


Image adapted from [Vulić and Korhonen, ACL-16]

Performance saturates or even drops when adding lower-frequency words into dictionaries.

Better results with fewer translation pairs (but less noisier pairs): **Symmetric/mutual nearest neighbours**

Identical words can also be useful, but they are heavily dependent on language proximity and writing scripts.

Can we reduce the requirements further?

Reducing the Cross-Lingual Signal Requirements Further

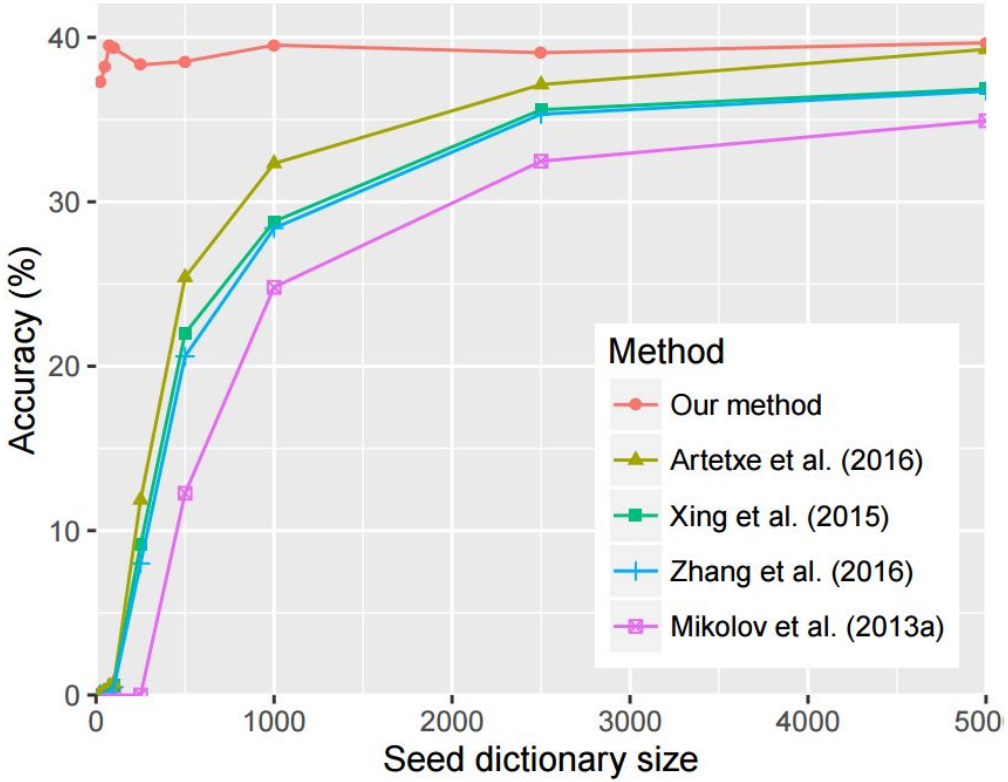


Image from [Artetxe et al., ACL-17]

Other sources of supervision:

- **Shared words and cognates**
- **Numerals**
- **...**

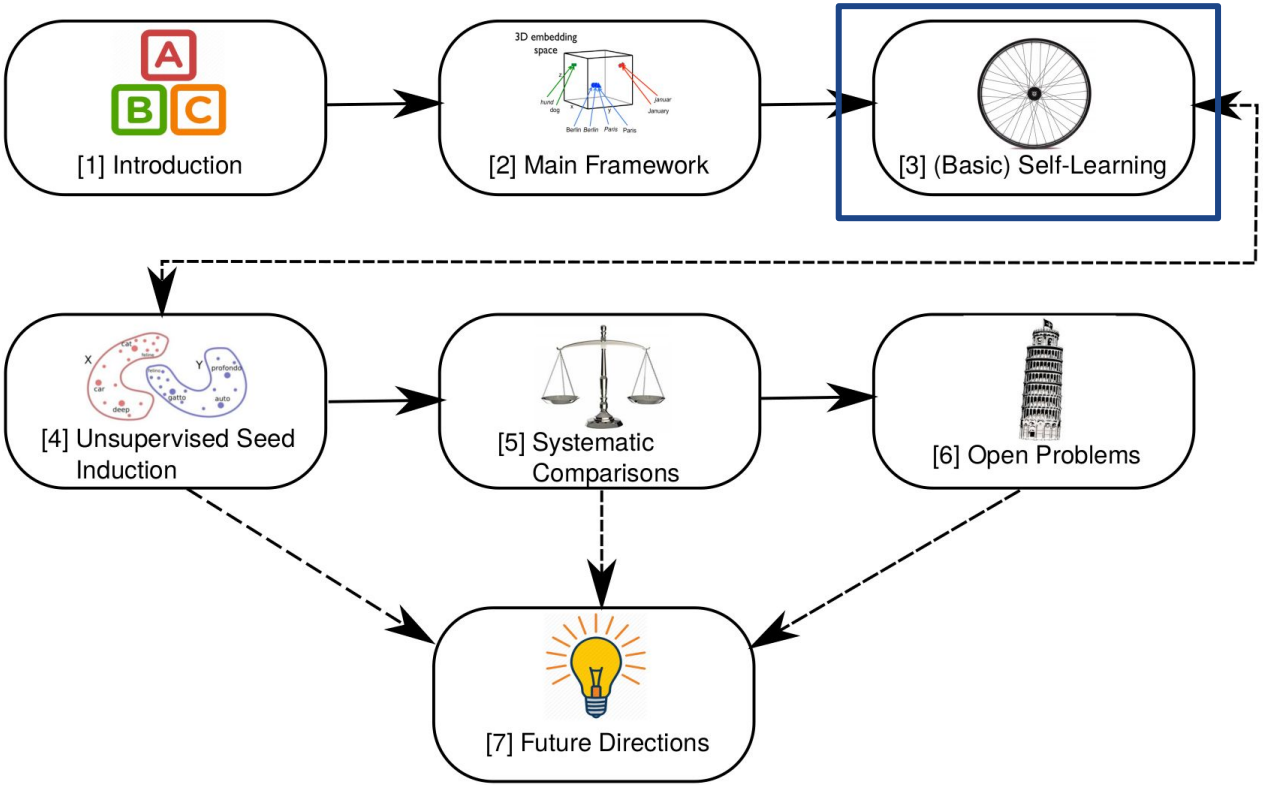
[Vulić and Moens, EMNLP-13; Vulić and Korhonen, ACL-16; Zhang et al., NAACL-17; Artetxe et al., ACL-17, Smith et al., ICLR-17]

Reducing the dictionaries only to 25-40 examples.

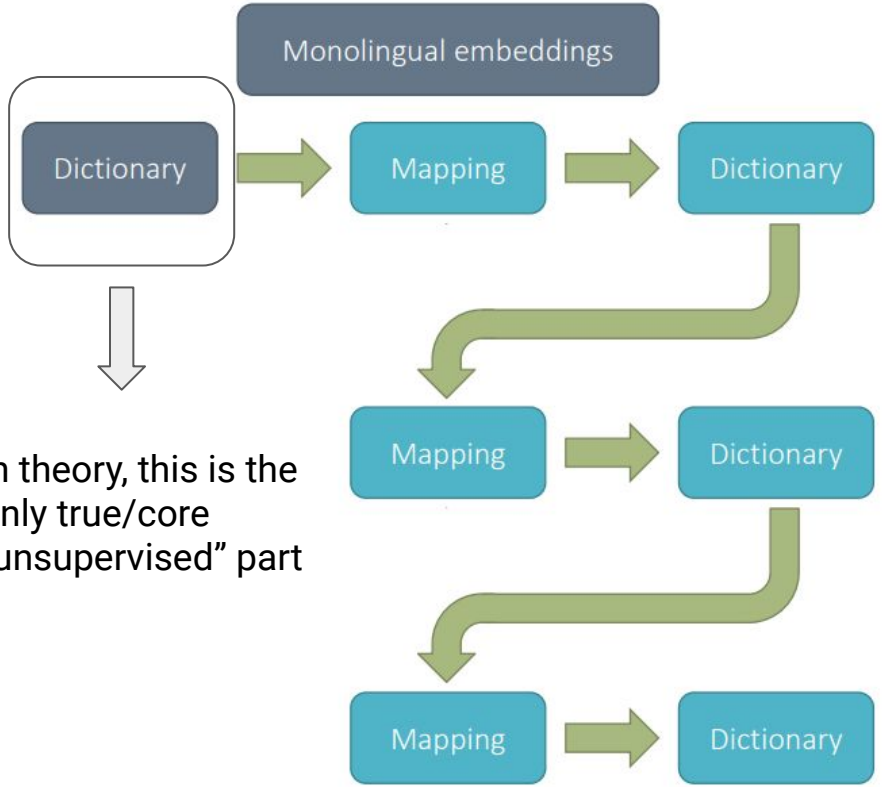
The crucial idea with limited supervision is **bootstrapping** or **self-learning**

The final frontier… Unsupervised methods...

3. (Basic) Self-Learning



Self-Learning in a Nutshell



In theory, this is the only true/core “unsupervised” part

- The seed dictionary improves over time, but…
- How do we start?
- How do we choose new candidates?
- How do we guarantee that we do not introduce noise?
- How does the method converge?

Image courtesy of Eneko Agirre

Self-Learning in a Nutshell

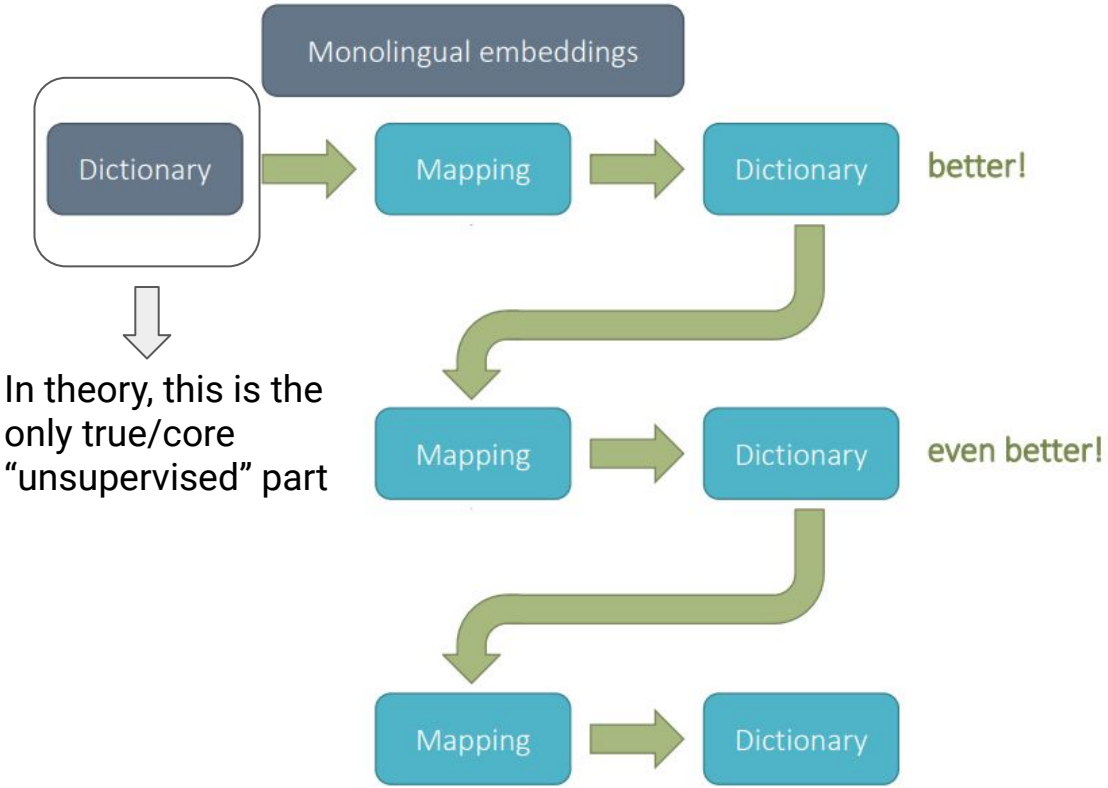


Image courtesy of Eneko Agirre

Bootstrapping with Orthogonal Procrustes

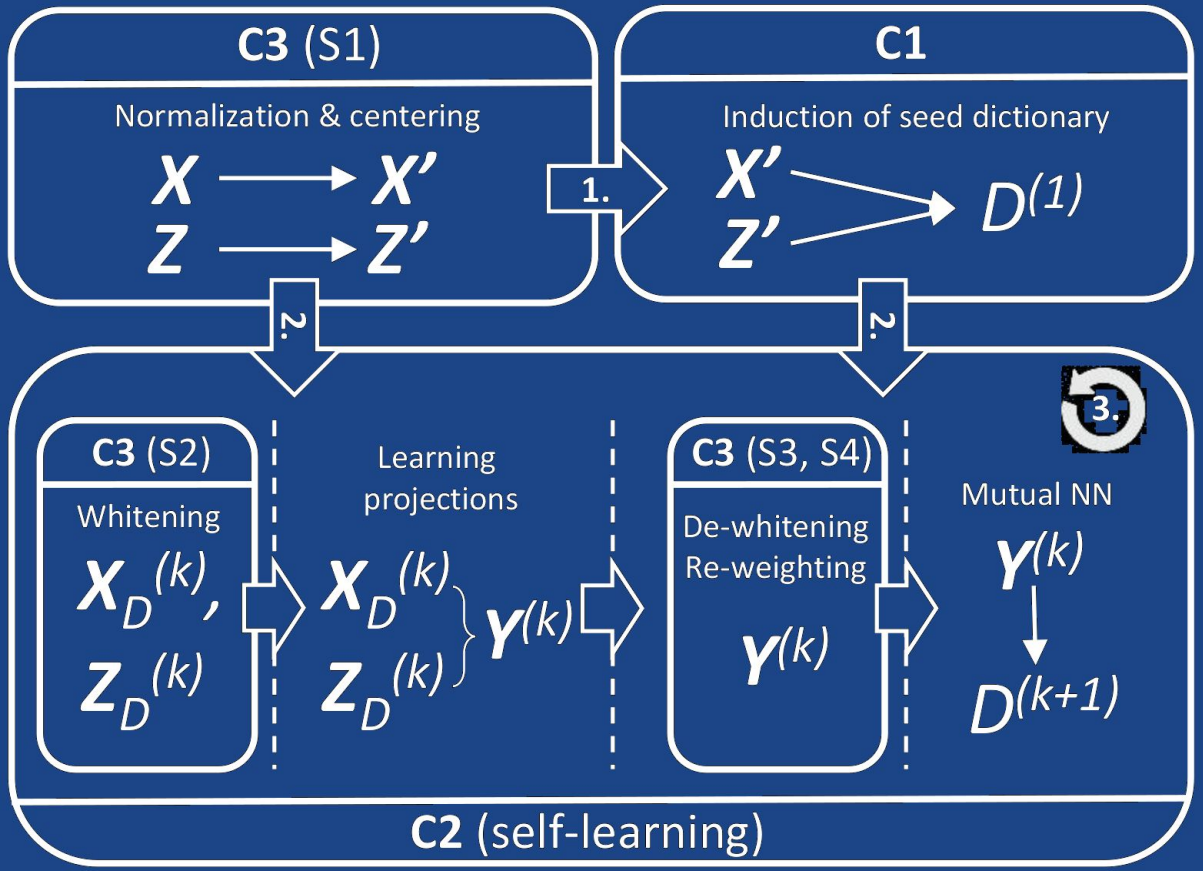
Algorithm 1: Bootstrapping Procrustes (PROC-B)

$\mathbf{X}_{L1}, \mathbf{X}_{L2} \leftarrow$ monolingual embeddings of $L1$ and $L2$
$D \leftarrow$ initial word translation dictionary
$n \leftarrow$ number of bootstrapping iterations
for *each of* n *iterations* **do**
  $\mathbf{X}_S \leftarrow$ L1 vectors for left words in D
  $\mathbf{X}_T \leftarrow$ L2 vectors for right words in D
  $\mathbf{W}_{L1} \leftarrow \arg\min_W \|\mathbf{X}_S\mathbf{W} - \mathbf{X}_T\|_2$
  $\mathbf{W}_{L2} \leftarrow \arg\min_W \|\mathbf{X}_T\mathbf{W} - \mathbf{X}_S\|_2$
  if *last iteration* **then**
    break
  $\mathbf{X'}_{L1} \leftarrow \mathbf{X}_{L1}\mathbf{W}_{L1}$
  $\mathbf{X'}_{L2} \leftarrow \mathbf{X}_{L2}\mathbf{W}_{L2}$
  $D_{1\rightarrow2} \leftarrow$ most-similar$(V_{L1}, \mathbf{X'}_{L1}, \mathbf{X}_{L2})$
  $D_{2\rightarrow1} \leftarrow$ most-similar$(V_{L2}, \mathbf{X'}_{L2}, \mathbf{X}_{L1})$
  $D \leftarrow D \cup (D_{1\rightarrow2} \cap D_{2\rightarrow1})$
return: $\mathbf{W}_{L1}$ (and/or $\mathbf{W}_{L2}$)

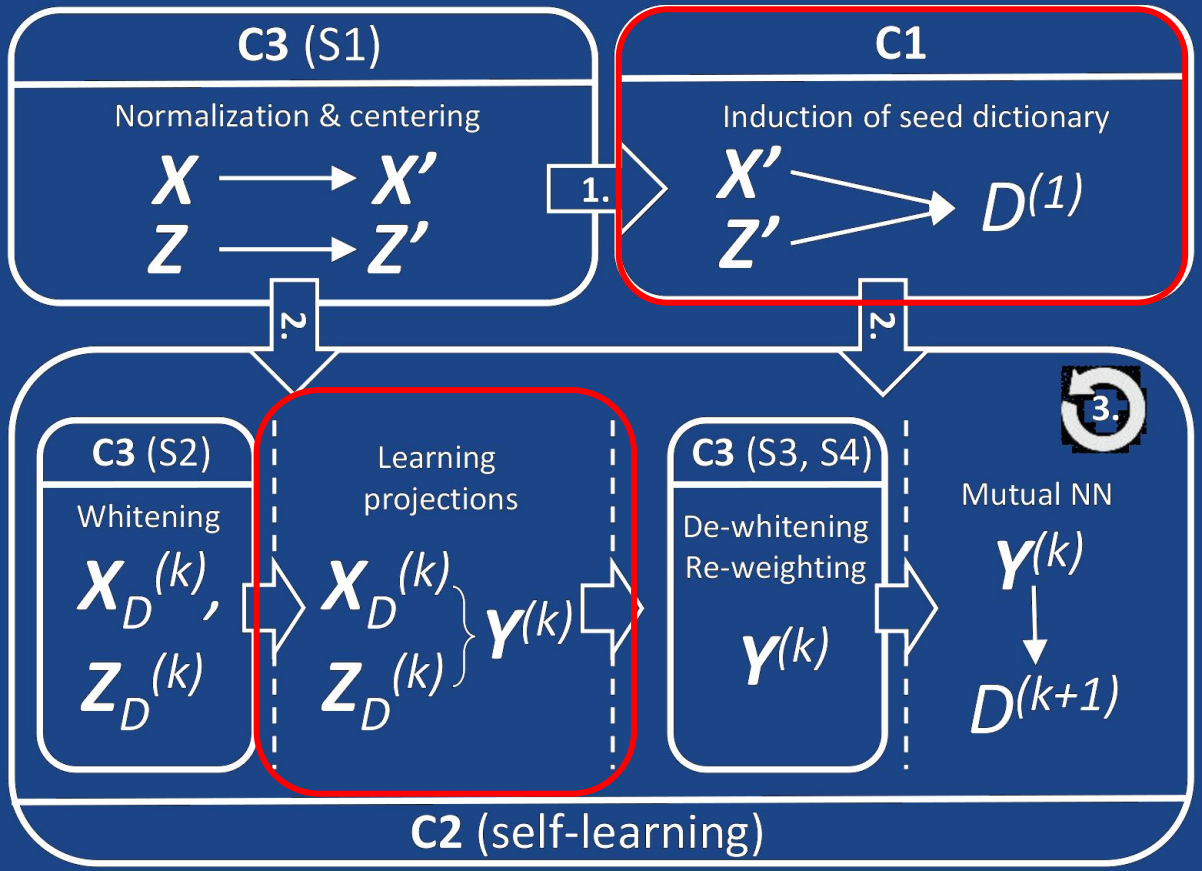
- Source-to-target and target-to-source mappings
- Symmetric nearest neighbours
- Nothing useful is learned after the first iteration *(only lower-frequency words and noise)*

[Artetxe et al., ACL-17, Glavaš et al., ACL-19]

A General Framework (with all the “tricks of the trade”…)



A General Framework (with all the “tricks of the trade”…)



Typical focus: Seed dictionary induction and learning projections.

The only difference between weakly supervised and fully unsupervised methods is than in C1.

A General Framework (with all the “tricks of the trade”…)

C1. Seed Lexicon Extraction

- **Supervised** models assume that the lexicon is available (at least some pairs…)
- **Fully unsupervised** models: automatically induce the lexicon

C2. Self-Learning Procedure

- iteratively apply the **Procrustes procedure** (or something else)
- different tricks to avoid suboptimal solutions (e.g., stochastic dropout of translation pairs, carefully-tuned frequency cut-offs)

C3. Preprocessing and Postprocessing Steps [Artetxe et al., AAAI-18]

- length normalization
- mean centering
- whitening and de-whitening

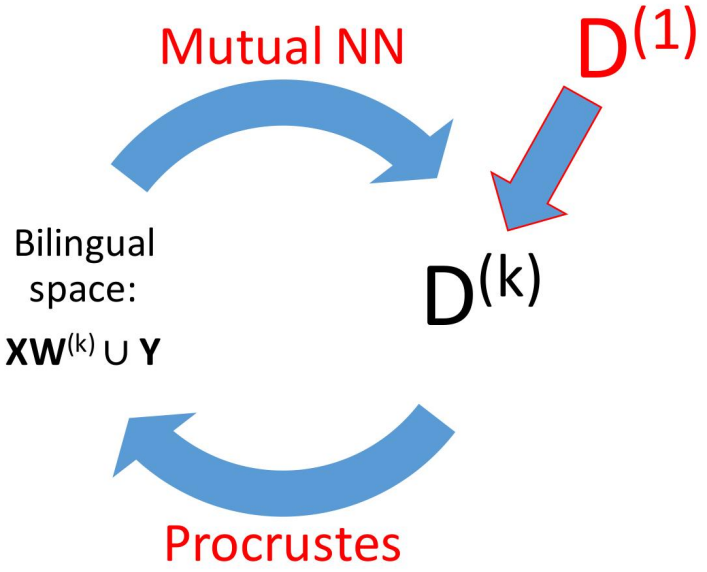
C1. Seed Lexicon Extraction and C2. Self-Learning

The **same general framework** for all unsupervised CLWE methods:

1. Induce (automatically) initial seed lexicon $\mathbf{D}^{(1)}$

Repeat:

2. Learn the projection $\mathbf{W}^{(k)}$ using $\mathbf{D}^{(k)}$
3. Induce a new dictionary $\mathbf{D}^{(k+1)}$ from $\mathbf{XW}^{(k)} \cup \mathbf{Y}$



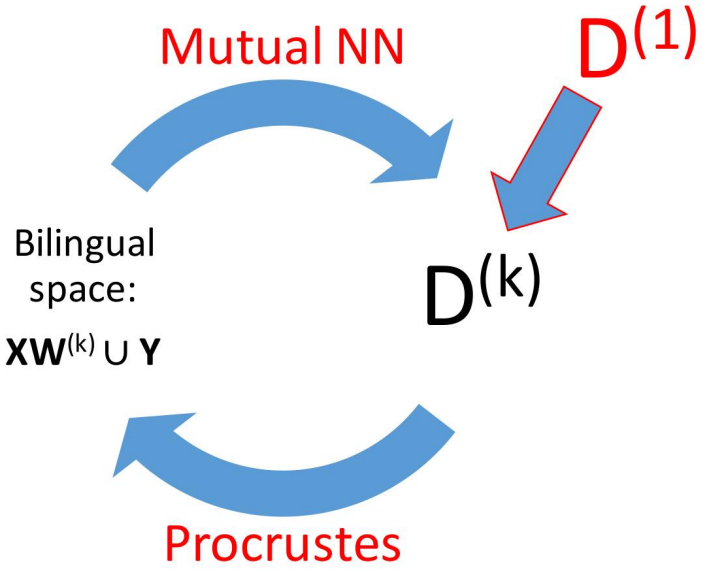
C1. Seed Lexicon Extraction and C2. Self-Learning

The **same general framework** for all unsupervised CLWE methods.

Different approaches to C1 (i.e., to obtain $\mathbf{D^{(1)}}$), e.g.,:

- **Adversarial learning**
- **Similarity of monolingual similarity distributions**
- **PCA-based similarity**
- **Solving optimal transport problem**

All solutions assume **approximate isomorphism** of monolingual embedding spaces



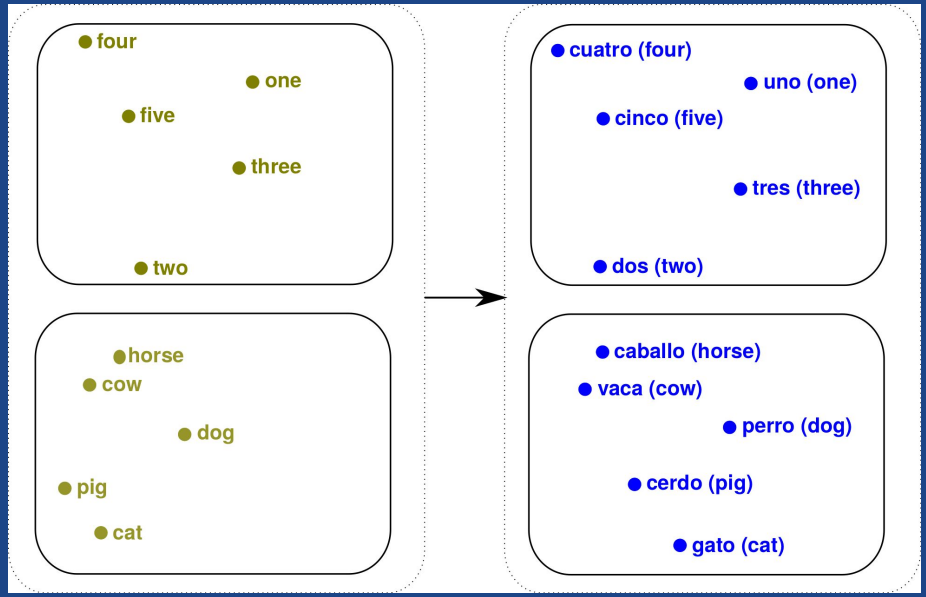
(Approximate) Isomorphism

“. . . we hypothesize that, if languages are used to convey thematically similar information in similar contexts, these random processes should be approximately isomorphic between languages, and that this isomorphism can be learned from the statistics of the realizations of these processes, the monolingual corpora, in principle without any form of explicit alignment.”

[Miceli Barone, RepL4NLP-16]

Fully unsupervised CLWE methods rely on this assumption **twice**

Is this problematic?
(...to be continued...)



[Mikolov et al., arXiv-13]

C3. Preprocessing and Postprocessing Steps

A general projection-based supervised framework of [Artetxe et al., AAAI-18]

- S0: **Normalisation:** Unit length normalisation, mean centering, or combination of both *(preproc)*
- S1: **Whitening**: turning covariance matrices into the identity matrix: each dimension obtains unit variance
- S2: **Re-weighting:** re-weigh each component according to its cross-correlation to increase the relevance of those that best match across languages *(after the mapping)*
- S3: **De-whitening**: use only if S1 was used; restore the original variance in each dimension *(after)*
- S4: **Dimensionality reduction**: keep only the first *n* components of the resulting embeddings (and set the rest to 0) *(after)*

C3. Preprocessing and Postprocessing Steps and Other Choices

A (most robust) unsupervised framework of [Artetxe et al., ACL-18]

- Unit length normalisation #1 + mean centering + unit length normalisation #2
- Re-weighting is done using the singular value matrix $\sum$ after the orthogonal Procrustes
- Bidirectional mapping and dictionary induction
- **Symmetric re-weighting** applied only once (and not in each iteration)
- CSLS used for word retrieval instead of the simple cosine
- Self-learning applied only on the top K (K=20,000) most frequent words
- **Stochastic dictionary induction**: dropout on current dictionary

$$\mathbf{W}_{L1} = \mathbf{U}\mathbf{V}^\top, \text{ with}$$
$$\mathbf{U}\mathbf{\Sigma}\mathbf{V}^\top = SVD(\mathbf{X}_T\mathbf{X}_S{}^\top)$$

Improving C3: Preprocessing Steps and Other “Tricks”

Another look into the future [Zhang et al., ACL-19]

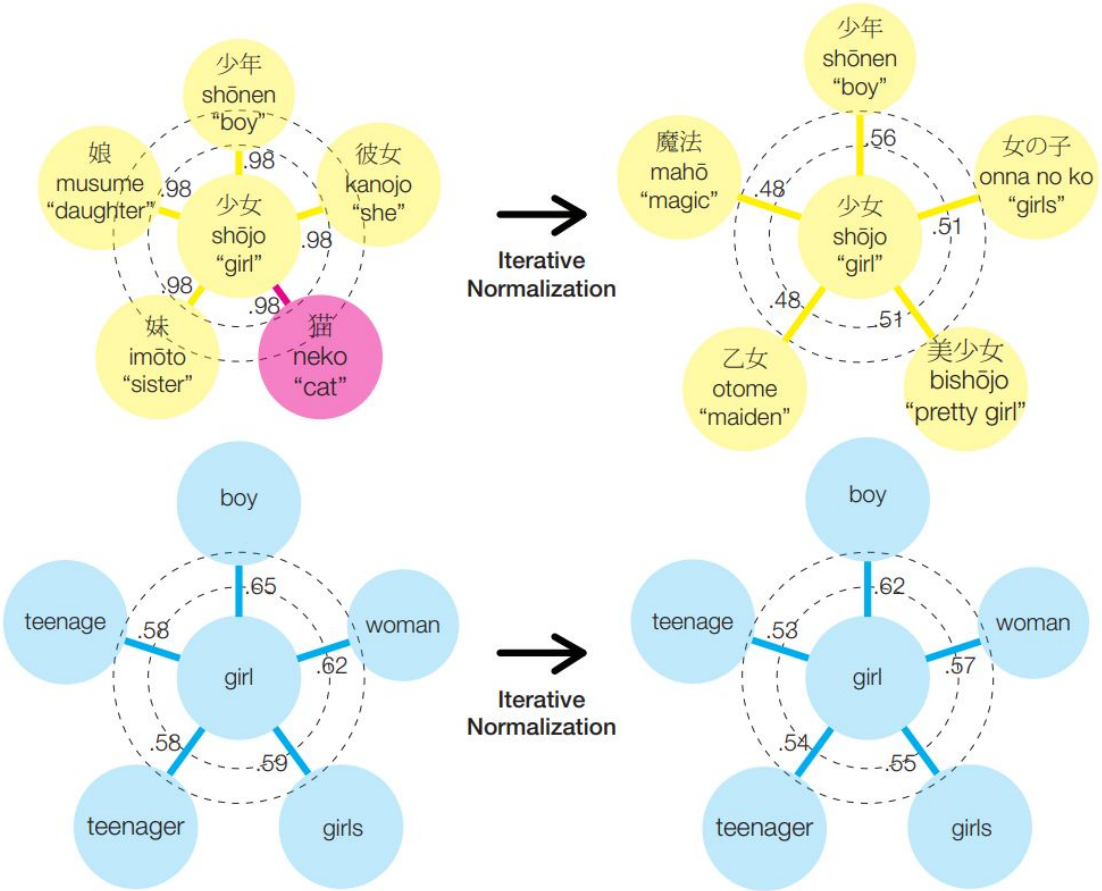
Based on [Artetxe et al., ACL-18]

Iterative Normalization guarantees:

- length-invariance
- center-invariance

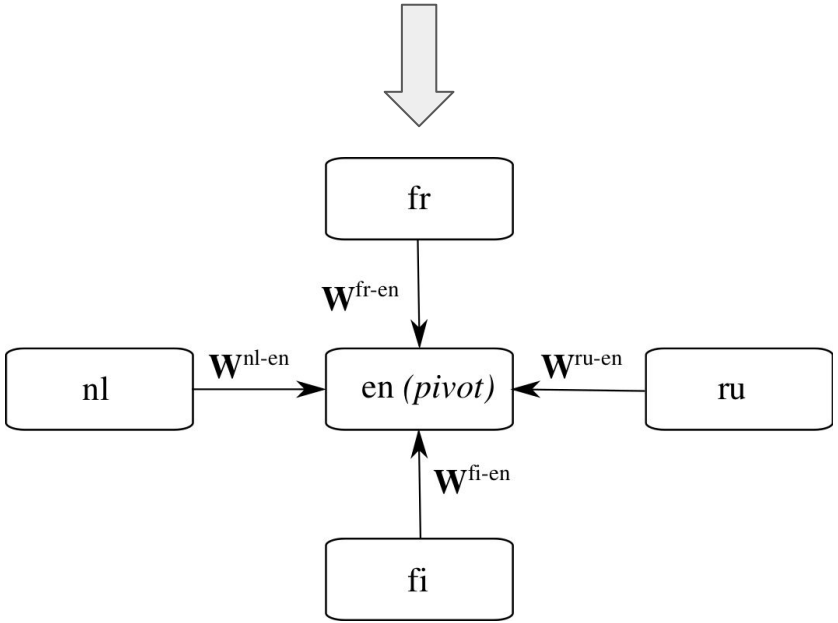
Preprocessing which makes two monolingual embedding spaces **more isomorphic**

Better results with orthogonal projections especially for **distant language pairs**.

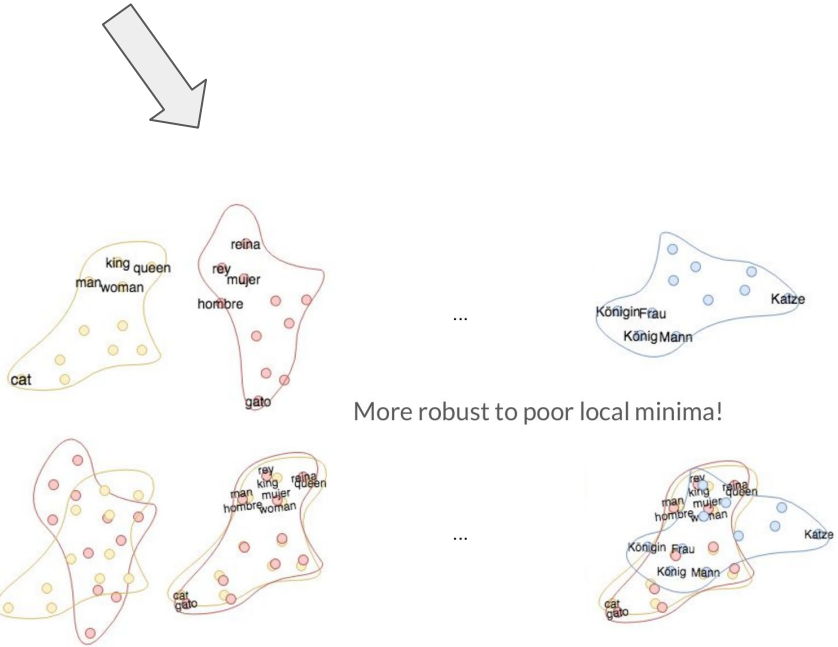


From Bilingual to Multilingual Word Embeddings

- **Single Hub Space (SHS)** versus **Iterative Hub Space (IHS)** [Heyman et al., NAACL-19]



N languages: N-1 seed lexicons needed...



N languages: (N-1)*N/2 seed lexicons needed...

A Quick Recap

Unsupervised CLWE-s are Projection-Based Methods (with some fancy improvements over the basic framework)

C1. Seed Lexicon Extraction

- How do we extract the initial signal from monolingual data only?
- How important is the initialisation actually?

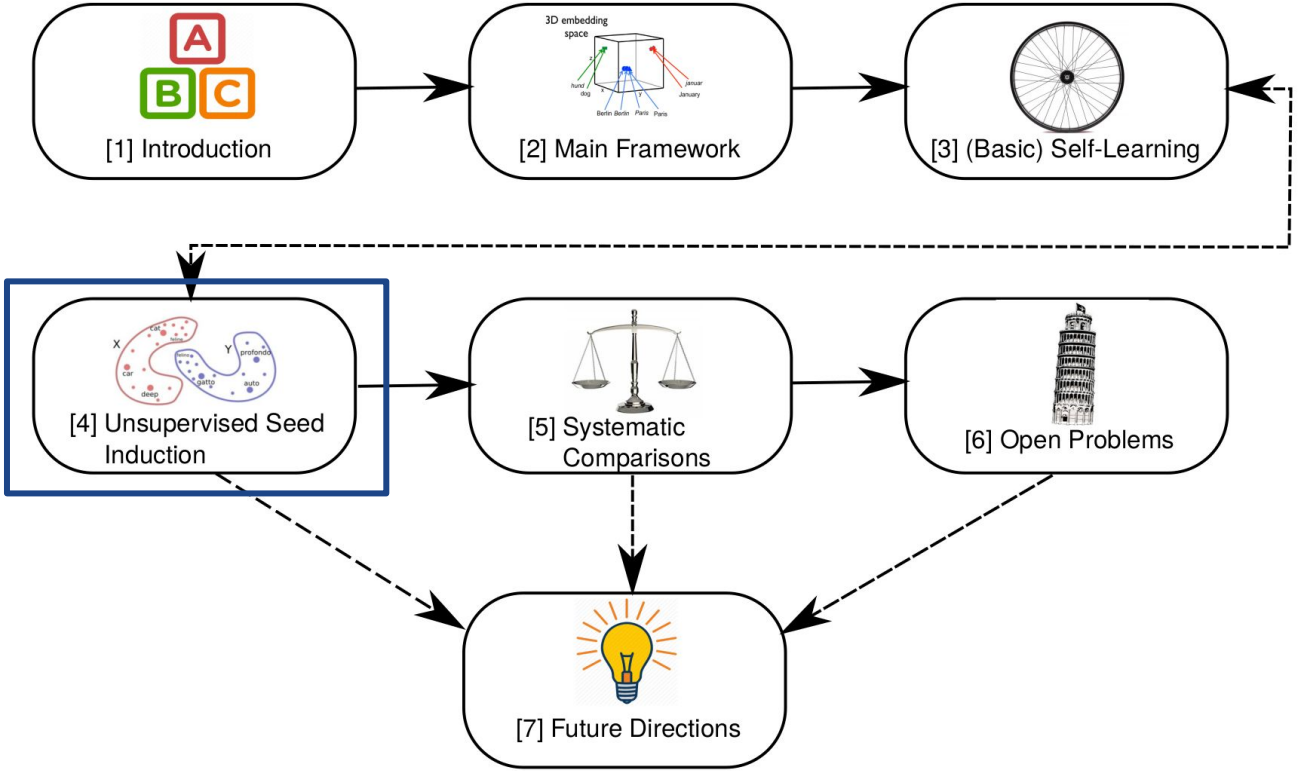
C2. Self-Learning Procedure

- How to denoisify the initial sub-optimal seed lexicon?
- How to design a robust procedure that works for virtually any language pair and avoids poor local optima?

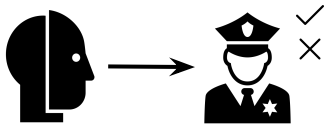
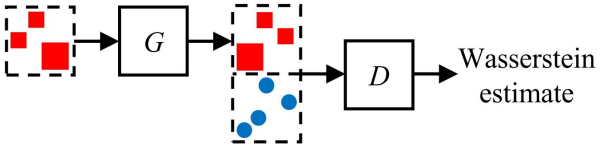
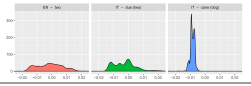
C3. Preprocessing and Postprocessing Steps

This comes next...

4. Unsupervised Seed Induction



Overview

Authors	Seed dictionary induction	
Barone (2016)	GAN 	Adversarial
Zhang et al. (2017a)		
Conneau et al. (2018)		
Zhang et al. (2017b)	Wasserstein GAN / Optimal transport 	
Xu et al. (2018)		
Alvarez-Melis and Jaakkola (2018)		
Artetxe et al. (2018)	Heuristic 	Non-adversarial
Hoshen and Wolf (2018)	Point Cloud Matching 	

Additional References (with Similar High-Level Ideas)

GAN

- Chen and Cardie (2018): a more robust and multilingual variant of the MUSE model

Wasserstein GAN / optimal transport

- Grave et al. (2018): convex relaxation for solving optimal transport with orthogonality constraint
- Alaux et al. (2019): extending the approach of Grave et al. to multilingual settings, using the RCSLS loss instead of the Orthogonal Procrustes formulation

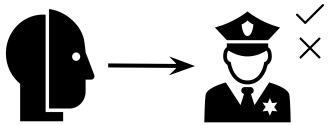
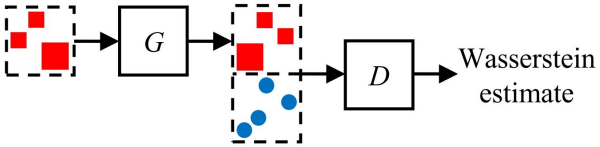
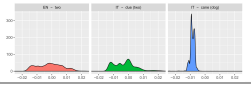
Heuristic

- Aldarmaki et al. (2018): structural similarity computed based on adjacency matrices and preserving relative distances in two monolingual spaces
- Heyman et al. (2019): multilingual generalisations of the robust framework from Artetxe et al. (later)

One promising direction for future research?

- Jawanpuria et al. (2019): simultaneously learning language-specific transformations to a shared latent space and a similarity metric in the shared space, *but still supervised...*

Overview

Authors	Seed dictionary induction	
Barone (2016)	GAN 	Adversarial
Zhang et al. (2017a)		
Conneau et al. (2018)		
Zhang et al. (2017b)	Wasserstein GAN / Optimal transport 	
Xu et al. (2018)		
Alvarez-Melis and Jaakkola (2018)		
Artetxe et al. (2018)	Heuristic 	Non-adversarial
Hoshen and Wolf (2018)	Point Cloud Matching 	

Adversarial mapping: Main idea

- Generator: projects source word embedding x into the target language
- Discriminator: differentiate between “fake” projected embeddings and “true” target language embeddings y

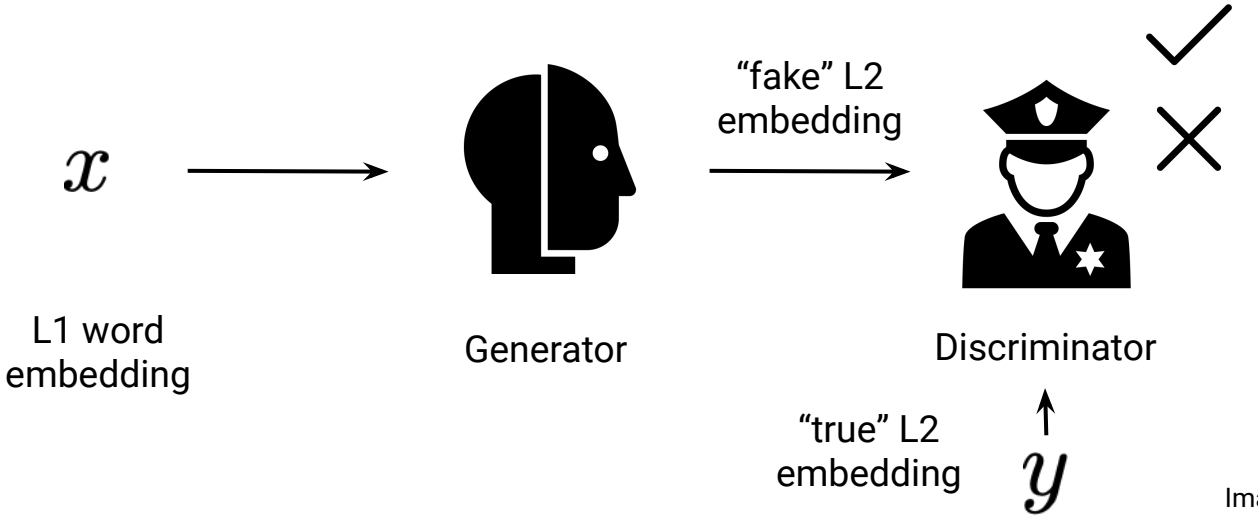


Image credit: Luis Prado, Jojo Ticon, Graphic Tigers

Adversarial mapping: Main idea

- Generator needs to match distribution of target language in order to fool discriminator consistently.
- *Hypothesis*: Best way to do this is to align words with their translations.

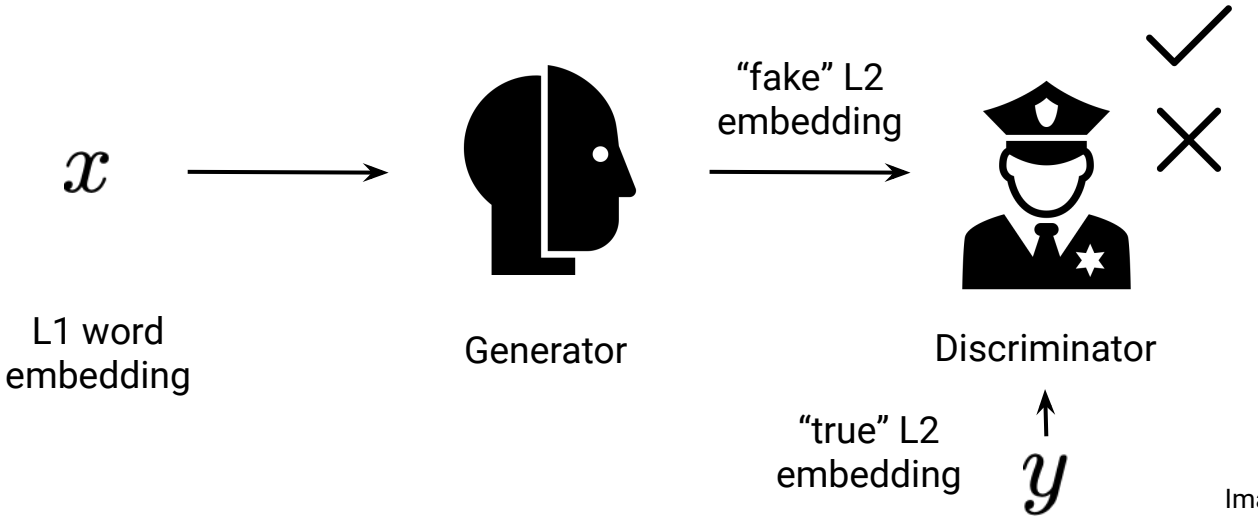
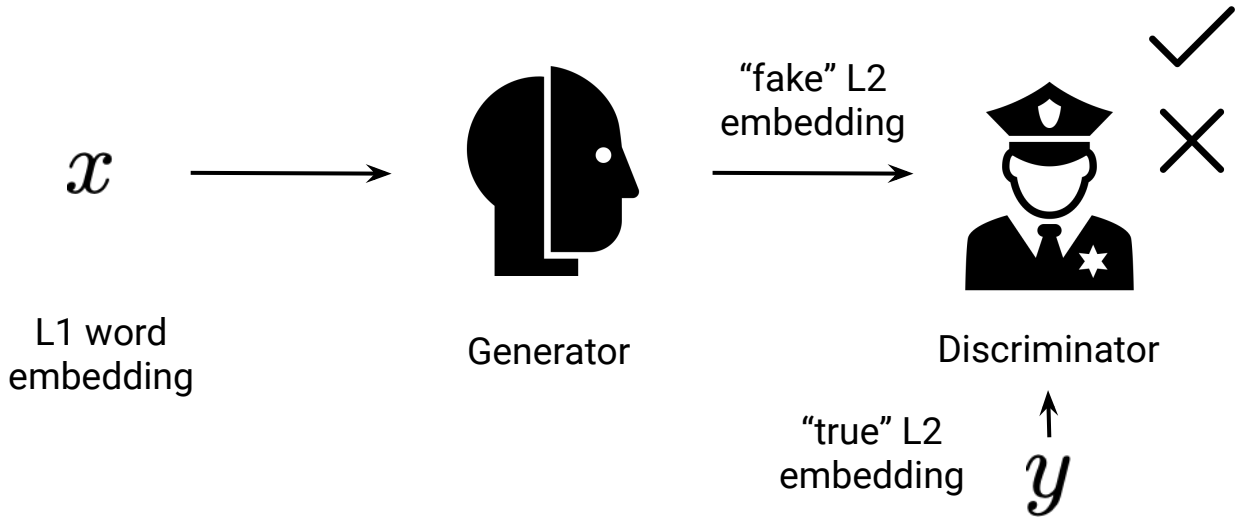


Image credit: Luis Prado, Jojo Ticon, Graphic Tigers

Adversarial mapping: Main idea

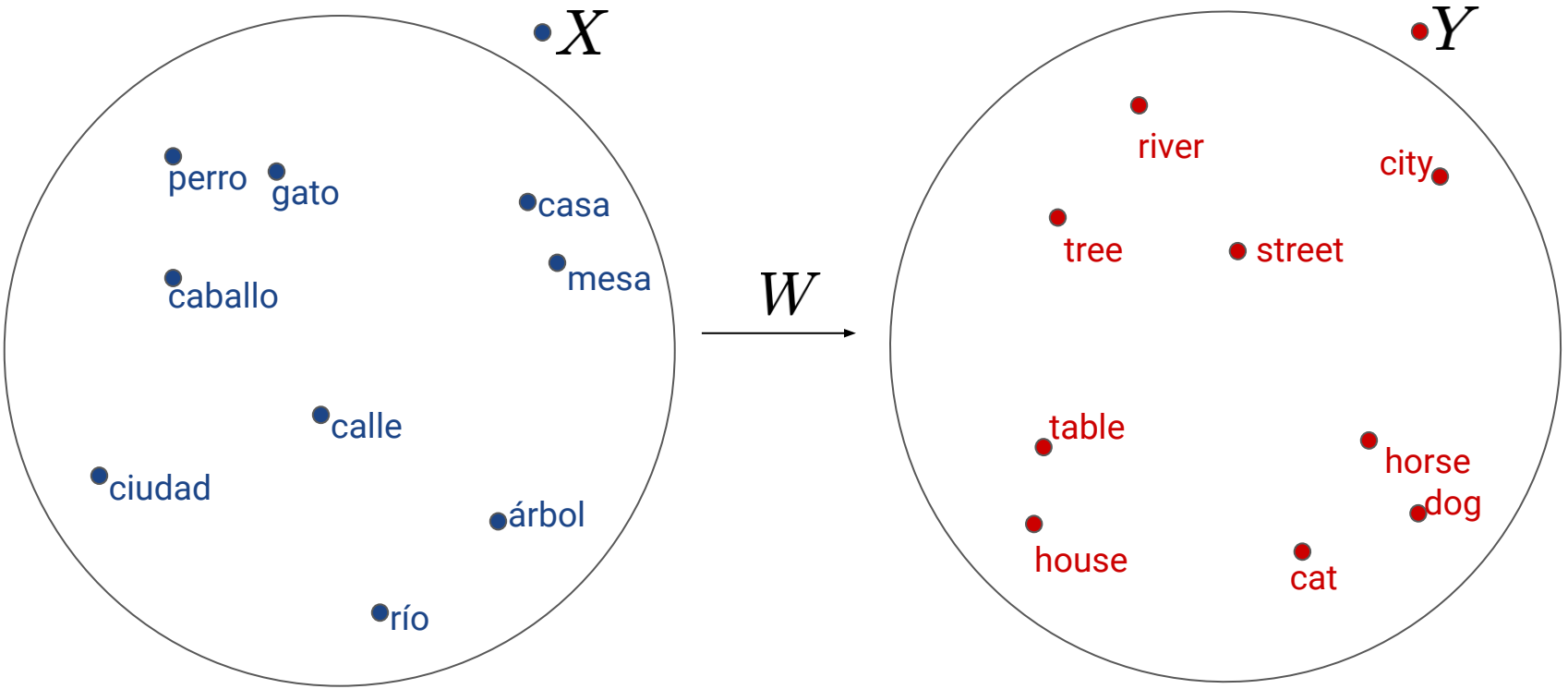


Why should this work at all?

Main assumption:
Embedding spaces in different languages have similar structure so that the same transformation can align source language words with target language words.

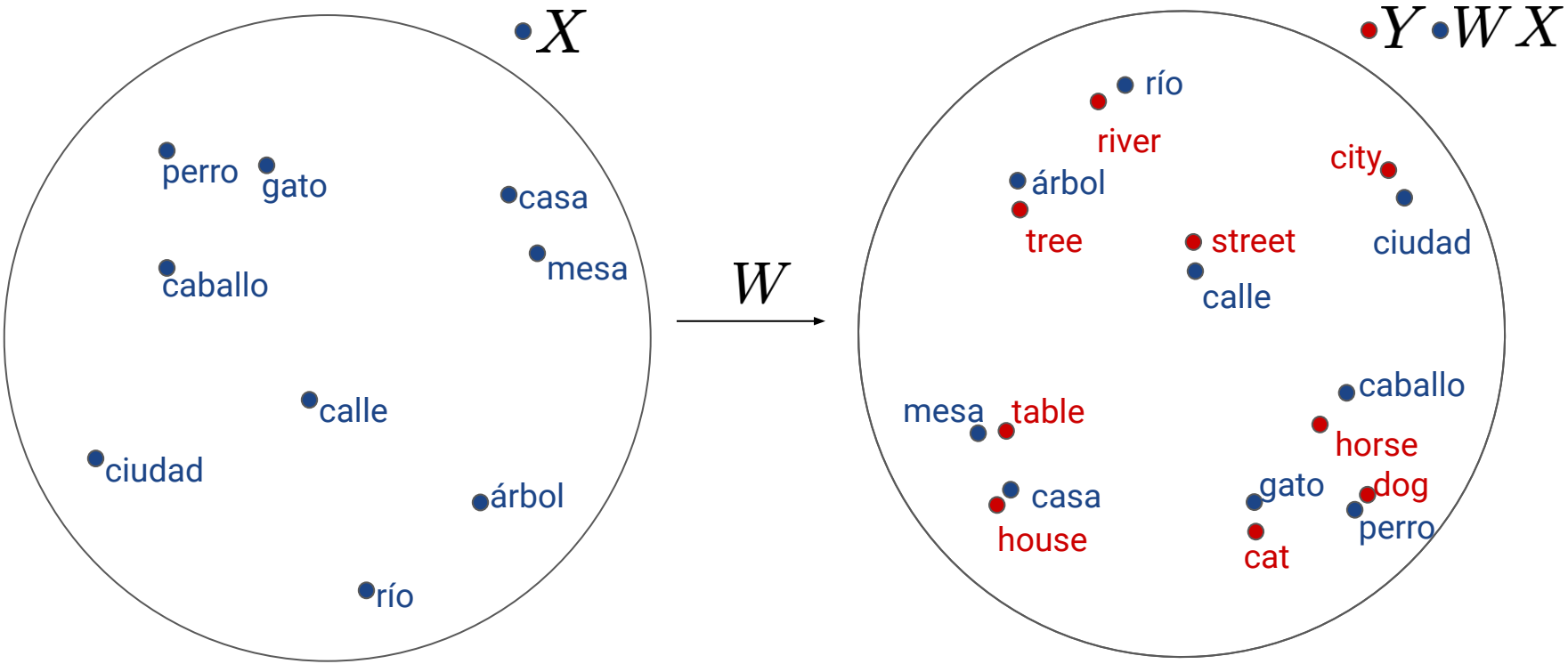
More specifically:
Embeddings spaces should be approximately isomorphic.

Supervised alignment



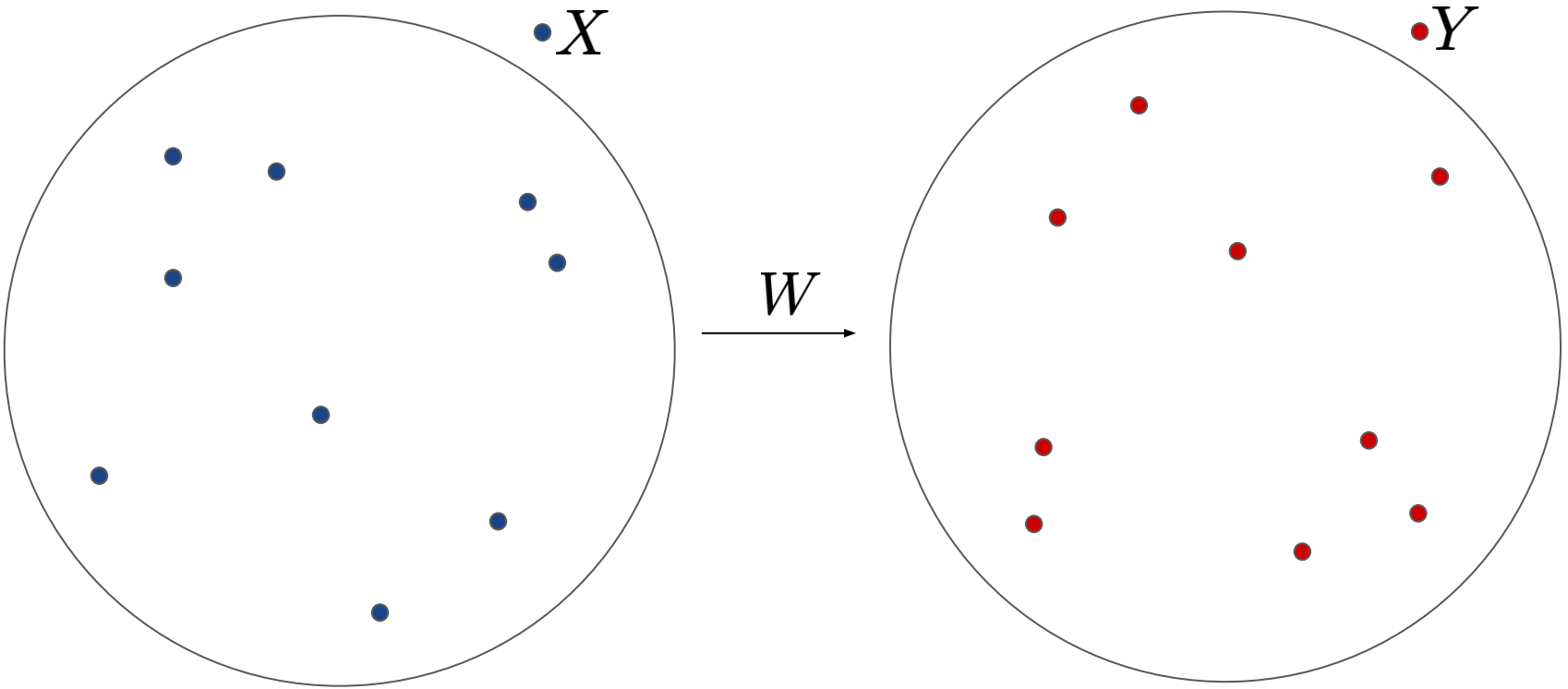
Inspired by: Slides from Eneko Agirre, Mikel Artetxe

Supervised alignment



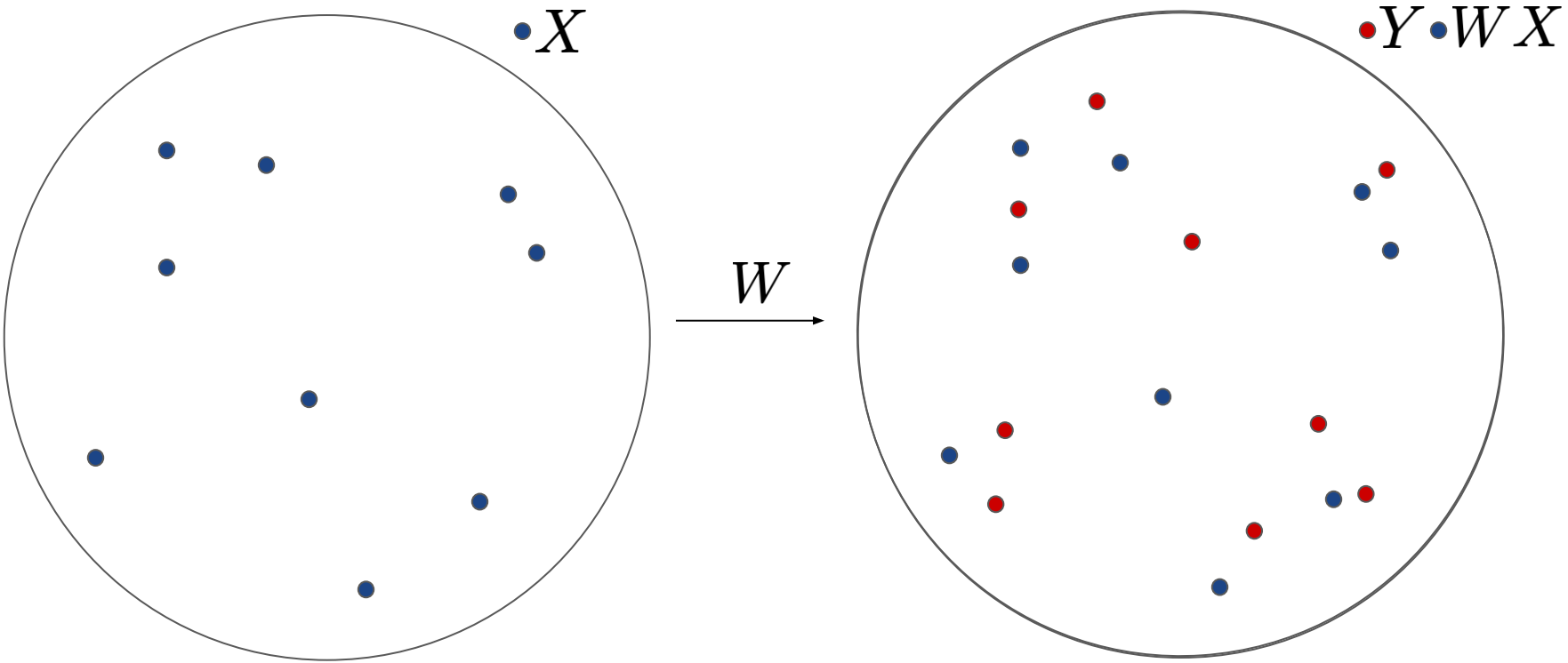
Inspired by: Slides from Eneko Agirre, Mikel Artetxe

Unsupervised alignment



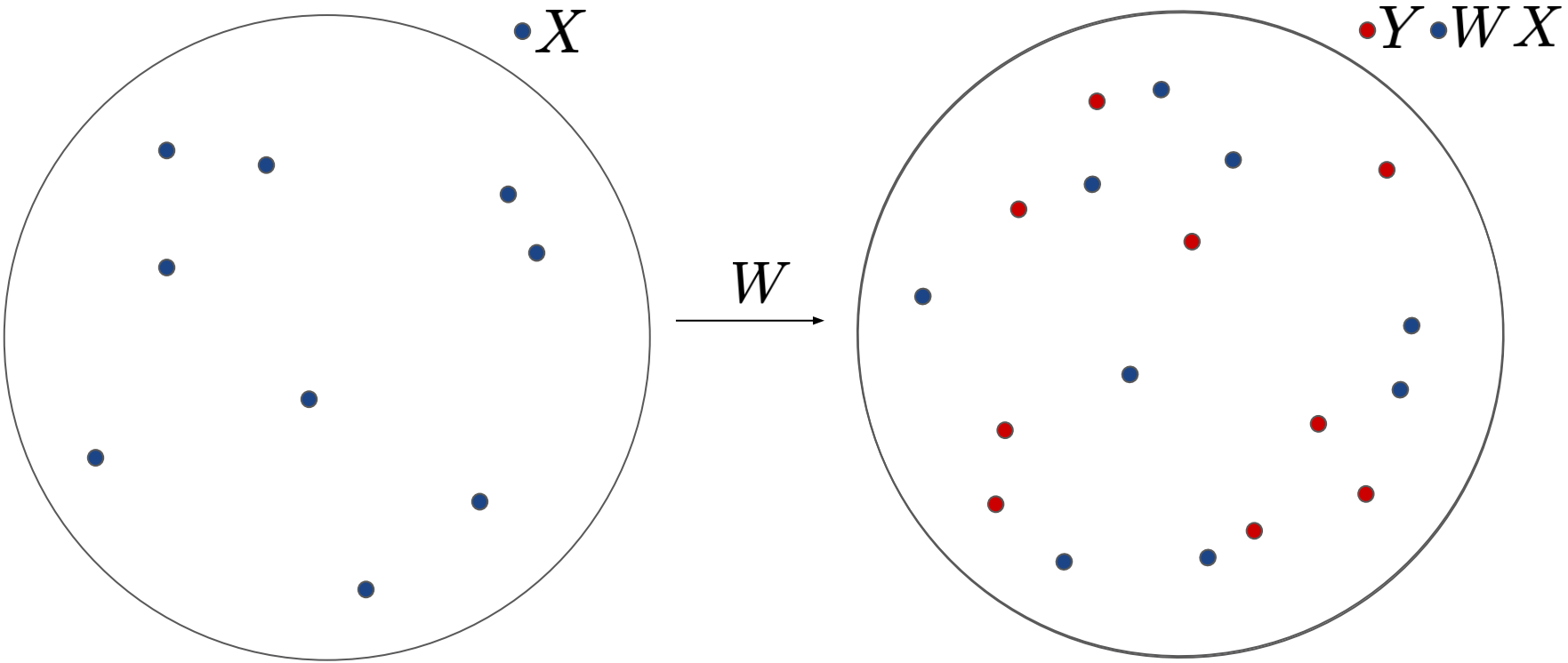
Inspired by: Slides from Eneko Agirre, Mikel Artetxe

Unsupervised alignment



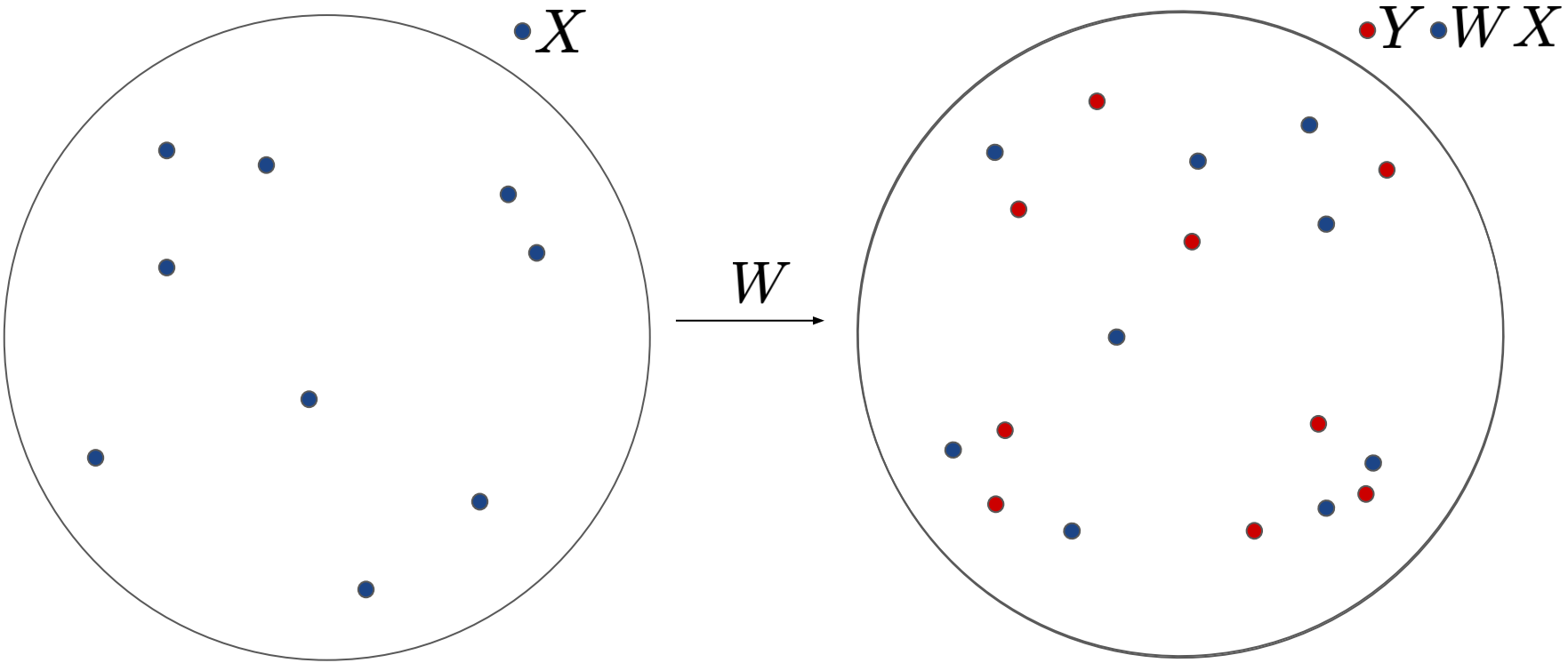
Inspired by: Slides from Eneko Agirre, Mikel Artetxe

Unsupervised alignment



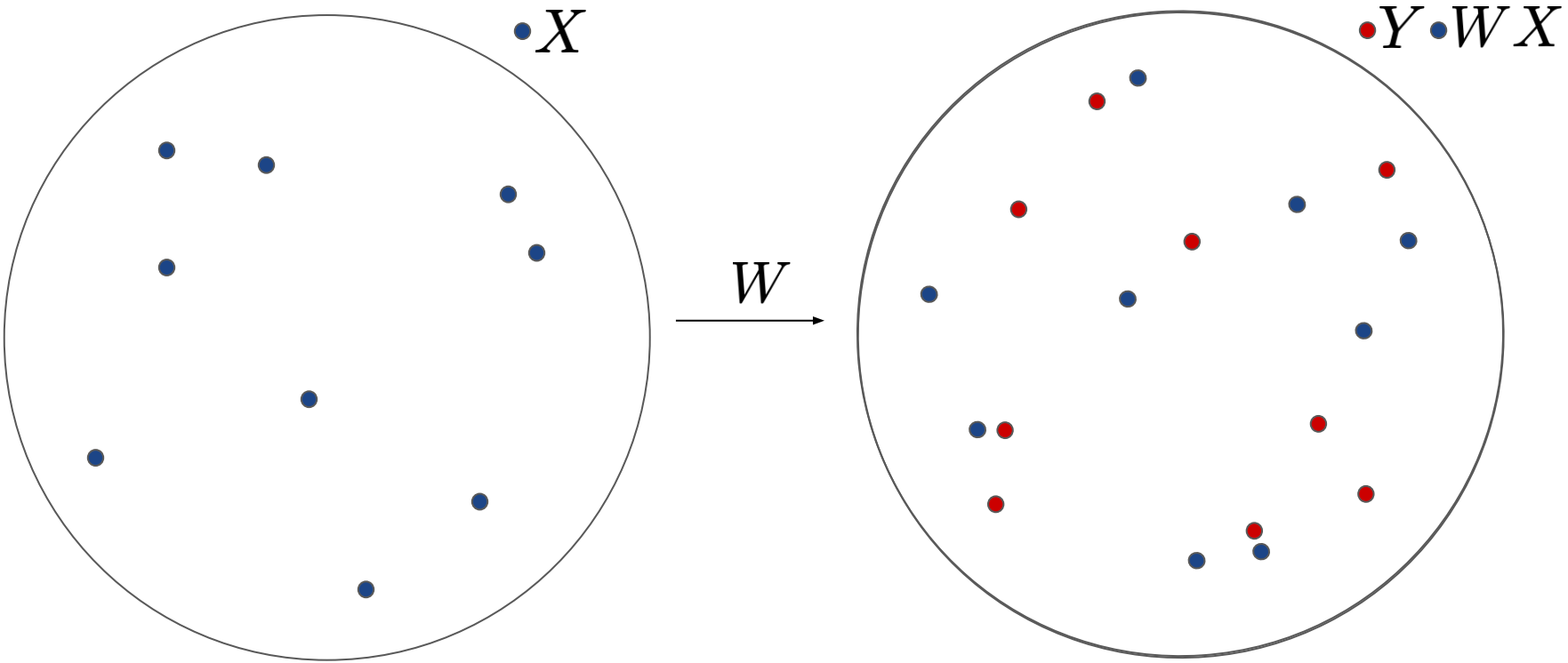
Inspired by: Slides from Eneko Agirre, Mikel Artetxe

Unsupervised alignment



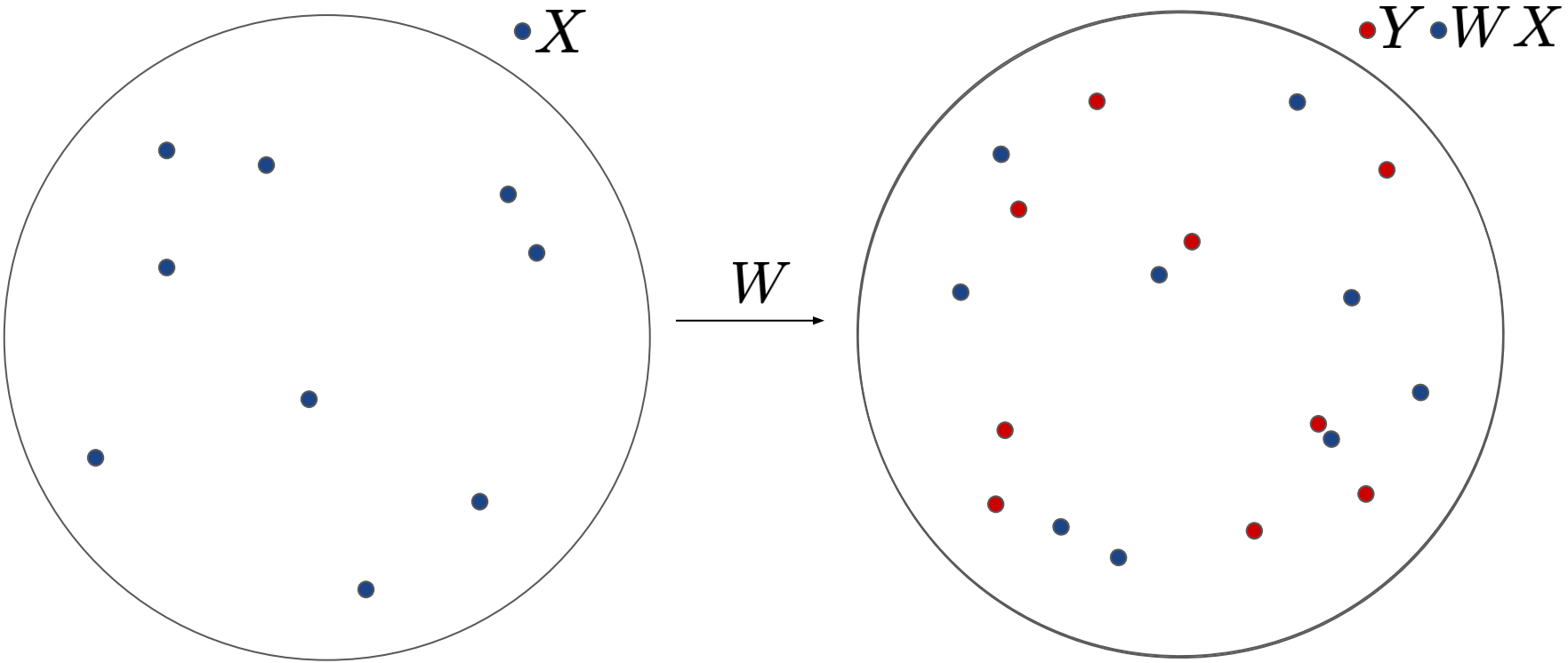
Inspired by: Slides from Eneko Agirre, Mikel Artetxe

Unsupervised alignment



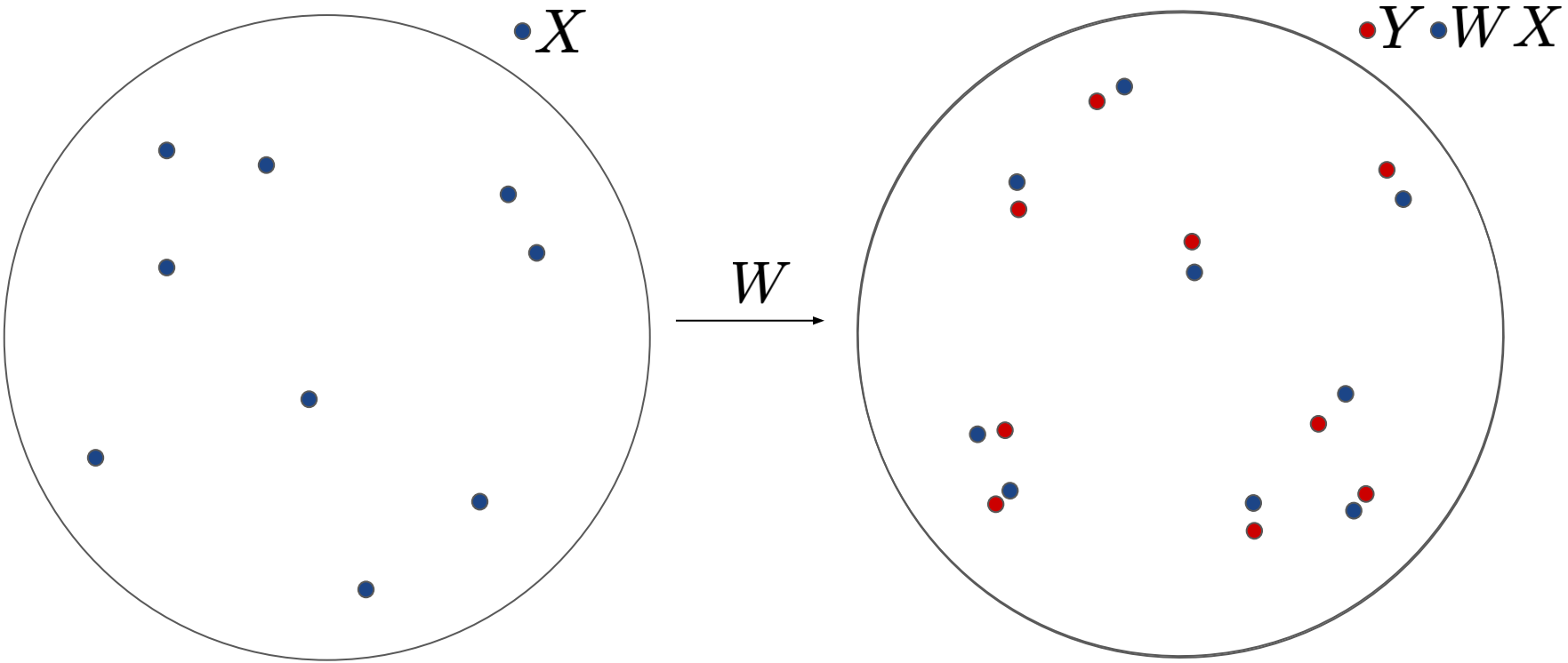
Inspired by: Slides from Eneko Agirre, Mikel Artetxe

Unsupervised alignment



Inspired by: Slides from Eneko Agirre, Mikel Artetxe

Unsupervised alignment



Inspired by: Slides from Eneko Agirre, Mikel Artetxe

Early adversarial approaches

Training GANs is often unstable, gets stuck in poor minima.

Early approaches were not as successful…

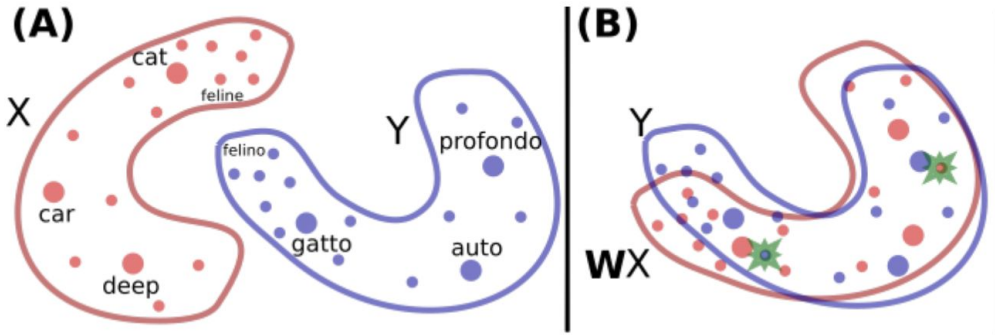
- The generator is not able to find a good alignment (Barone, 2016).
- Training requires careful regularization and hyper-parameter tuning; only works for small embedding sizes (50 dimensions) and on non-standard benchmarks (Zhang et al., 2017a).
- More robust approach in certain settings (Conneau et al., 2018)

For more on GANs: NAACL 2019 Deep Adversarial Learning tutorial

Conneau et al. (2018)

1. **Monolingual word embeddings:**
   Learn monolingual vector spaces X and Y.

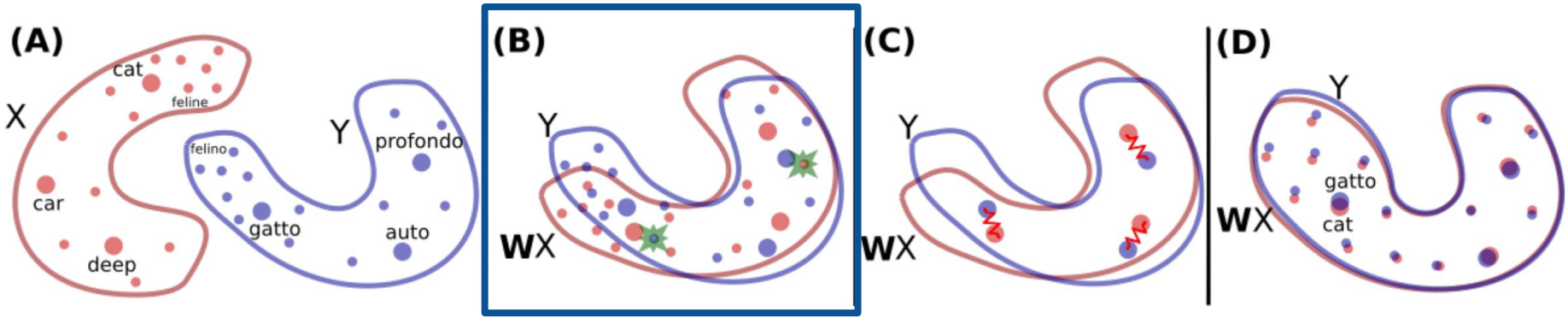
2. **Adversarial mapping:**
   Learn a translation matrix W. Train discriminator to discriminate samples from WX and Y.



Conneau et al. (2018)

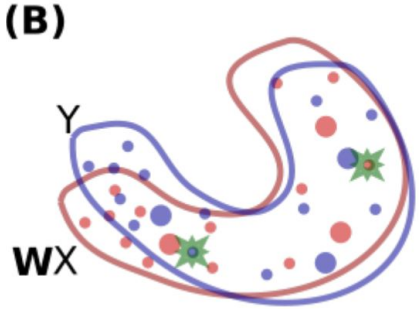
3. **Refinement (Procrustes analysis):**
   Build bilingual dictionary of frequent words using W. Learn a new W based on frequent word pairs.

4. **Cross-domain similarity local scaling (CSLS):**
   Use similarity measure that increases similarity of isolated word vectors, decreases similarity of vectors in dense areas.



Adversarial mapping in detail

- **Generator**: the projection matrix W
- **Discriminator**: differentiate between the “fake” projected source samples WX and the “true” target samples Y.



an embedding in X

an embedding in Y

$$\mathcal{L}_D(\theta_D|W) = -\frac{1}{n}\sum_{i=1}^{n} \log P_{\theta_D}\left(\text{source} = 1 \middle| Wx_i\right) - \frac{1}{m}\sum_{i=1}^{m} \log P_{\theta_D}\left(\text{source} = 0 \middle| y_i\right)$$

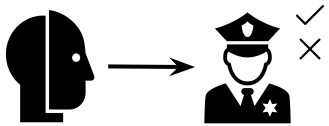
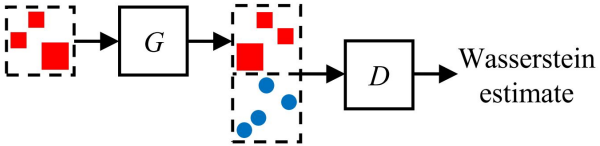
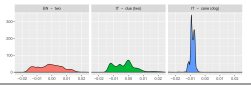
Maximize probability of predicting correct source

$$\mathcal{L}_W(W|\theta_D) = -\frac{1}{n}\sum_{i=1}^{n} \log P_{\theta_D}\left(\text{source} = 0 \middle| Wx_i\right)$$

Maximize probability of “fooling” discriminator

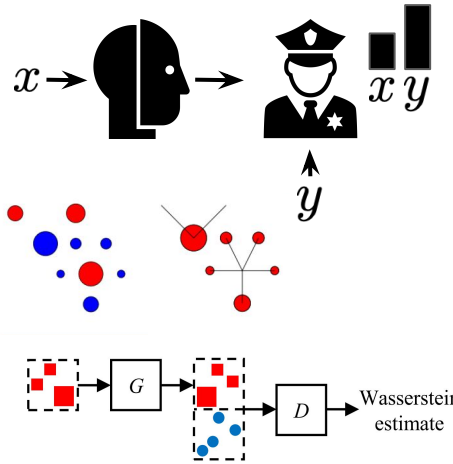
For every input sample, generator and discriminator are trained successively with SGD to minimize their losses.

Overview

Authors	Seed dictionary induction	
Barone (2016)	GAN 	Adversarial
Zhang et al. (2017a)		
Conneau et al. (2018)		
Zhang et al. (2017b)	Wasserstein GAN / Optimal transport 	
Xu et al. (2018)		
Alvarez-Melis and Jaakkola (2018)		
Artetxe et al. (2018)	Heuristic 	Non-adversarial
Hoshen and Wolf (2018)	Point Cloud Matching 	

Wasserstein GAN / Optimal Transport

- Wasserstein GAN
- Earth Mover's Distance (EMD)
- EMD and Wasserstein Distance
- Solving the optimal transport problem
- Gromov Wasserstein Distance

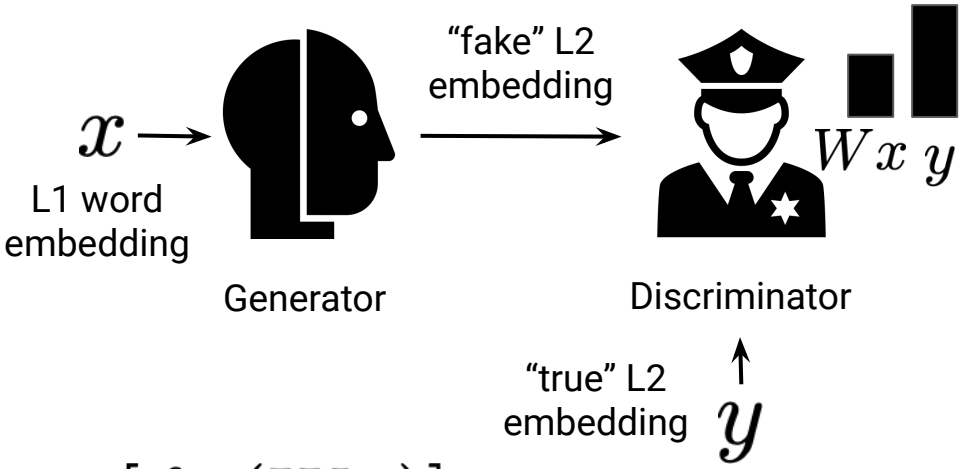


$$d_T^{\lambda}(P_1, P_2) = \min_{T \in \mathcal{U}(p,q)} TC - \frac{1}{\lambda}\mathcal{H}(T)$$

$$GW(C, C', p, q) = \min_{T \in \mathcal{U}(p,q)} \sum_{i,j,k,l} T_{ij} T_{kl} L_{ijkl}$$

Wasserstein GAN

Wasserstein GAN (WGAN; Arjovsky et al., 2017) has been shown to be more stable than the original GAN



Discriminator: assign higher scores to "true" than to "fake" samples

$$\max_D \mathbb{E}_{y \sim P^T}[f_D(y)] - \mathbb{E}_{x \sim P^S}[f_D(Wx)]$$

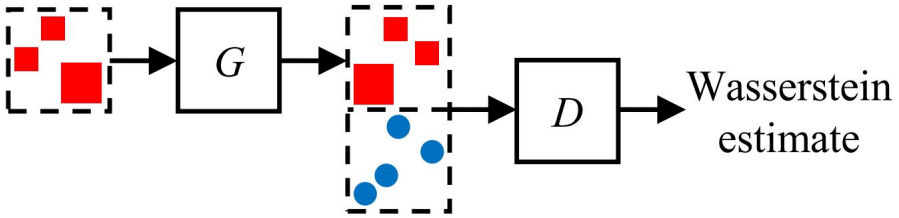
Generator: "fool" discriminator to predict high scores for "fake" samples

$$\min_W -\mathbb{E}_{x \sim P^S}[f_D(Wx)]$$

Wasserstein GAN

Wasserstein GAN (WGAN; Arjovsky et al., 2017) implicitly minimizes Wasserstein distance

- Discriminator *D* takes source and target word embeddings as input and estimates Wasserstein distance
- Generator G uses this information to try to minimize Wasserstein distance



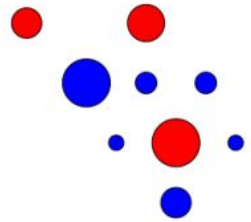
Zhang et al. (2017b)

- Wasserstein distance is the Earth Mover's distance for continuous distributions

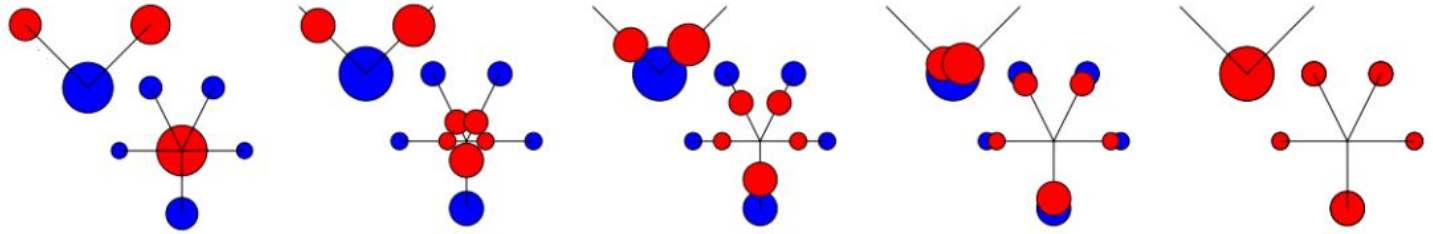
→ What is the Earth Mover's Distance?

Earth Mover's Distance (EMD)

- Distance measure between (discrete) probability distributions.

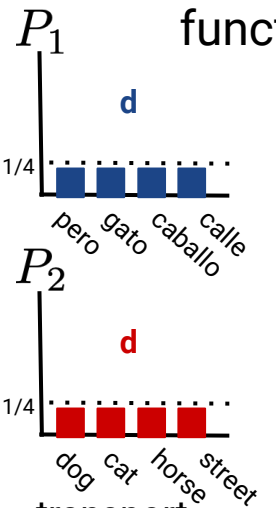


- Red distribution: "dirt"
- Blue distribution: "holes"
- Minimize overall distance of dirt moved into holes
- Distance between points can be any distance (Euclidean, etc.)



Earth Mover's Distance (EMD)

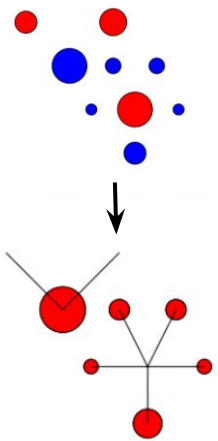
- Distance measure between (discrete) probability distributions.
- Discrete probability distributions can be represented as sum of Dirac delta functions: $P_1 = \sum_i p_i \delta_{x_i}$ and $P_2 = \sum_j q_j \delta_{y_j}$



probability of *i*-th word, assumed to be uniform

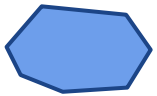
Dirac delta function: unit point mass at point x_i

transport matrix: indicates assignment, i.e. how much probability mass is transported from x_i to y_j



$$EMD(P_1, P_2) = \min_{T \in \mathcal{U}(p,q)} \sum_i \sum_j T_{ij} c(x_i, y_j)$$

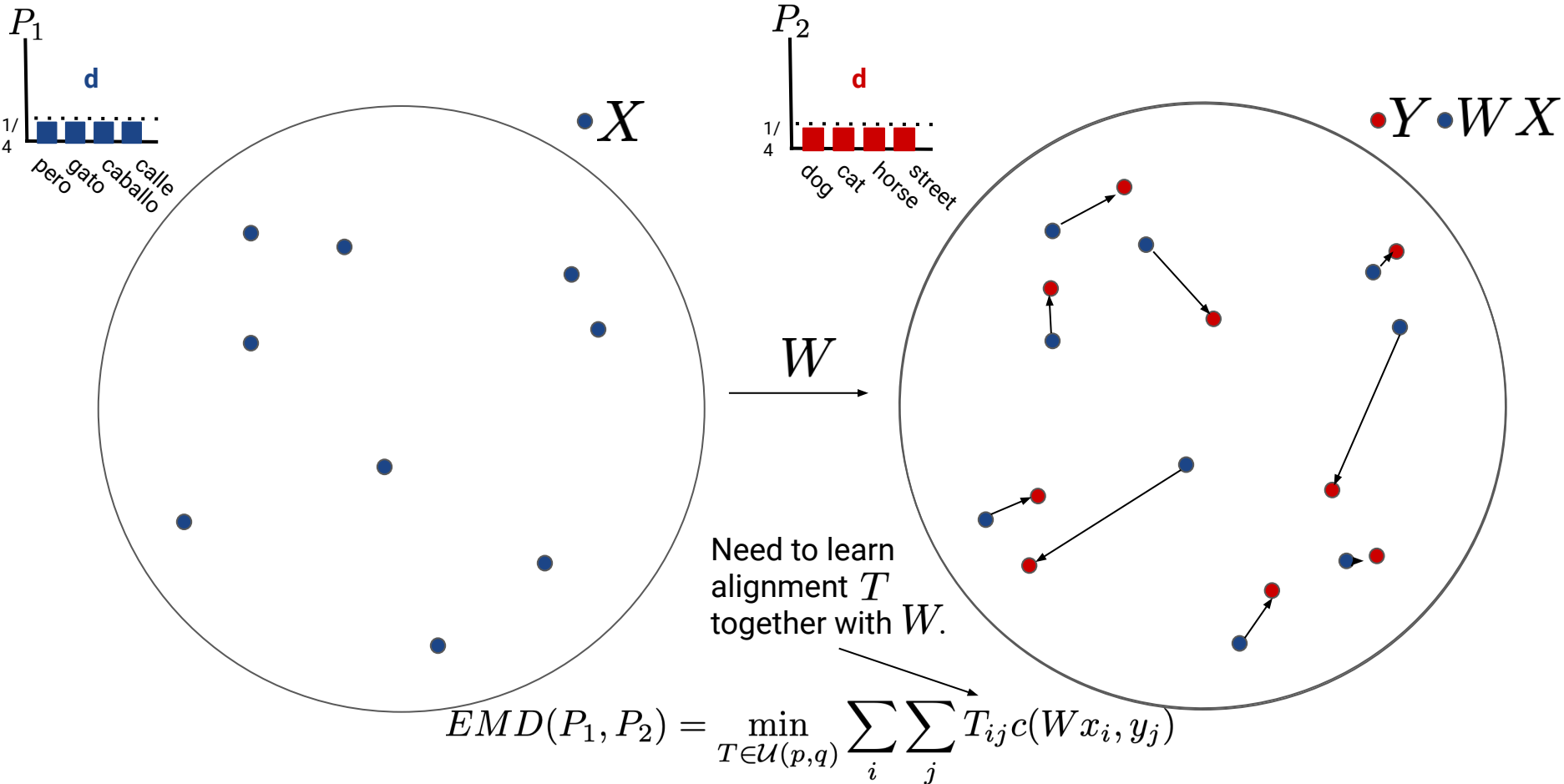
transport polytope:



$$\mathcal{U}(p,q) = \left\{ T \mid T_{ij} \geq 0, \sum_j T_{ij} = p_i, \sum_i T_{ij} = q_j \right\}$$

cost: distance between x_i and y_j

Learning cross-lingual embeddings with EMD



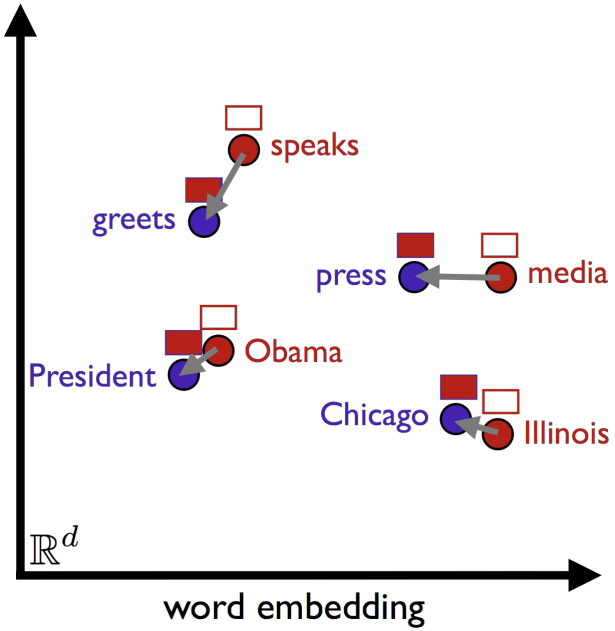
Learning cross-lingual embeddings with EMD

In linear optimization, solutions to the OT problem are always on a vertex of $\mathcal{U}(p, q)$ i.e. T is a sparse $d \times d$ matrix with only $2d - 1$ (Brualdi, 2006; §8.1.3).

→ Good fit for matching words across languages.

→ Probability mass preservation: We want to translate all words.

- EMD has been used to learn an alignment between words and their translations based on seed words (Zhang et al., 2016a; 2016b)
- EMD has also been used for measuring document similarity (Kusner et al., 2015)



EMD and Wasserstein distance

- Wasserstein distance generalizes EMD to continuous distributions.
- Not necessary for cross-lingual word embeddings, but theoretically motivates WGAN.

$$EMD(P_1, P_2) = \min_{T \in \mathcal{U}(p,q)} \sum_i \sum_j T_{ij} c(x_i, y_j)$$

infimum: greatest lower bound

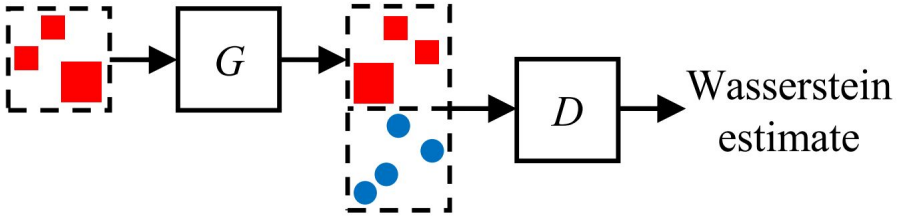
$$W(P_1, P_2) = \inf_{\gamma \in \Gamma(P_1, P2)} \int_x \int_y \gamma(x, y) c(x, y) dy\, dx$$

set of all joint distributions $\gamma(x, y)$

$$= \inf_{\gamma \in \Gamma(P_1, P2)} \mathbb{E}_{(x,y) \sim \gamma}[c(x, y)]$$

Wasserstein GAN

Objective to minimize: Wasserstein distance between transformed source and target embeddings:



Zhang et al. (2017b)

$$\min_{G \in \mathbb{R}^{d \times d}} W\left(P^{G(S)}, P^{T}\right)$$

If cost c is Euclidean distance, can be written as (Kantorovich-Rubinstein duality, Villani, 2006):

supremum: least upper bound

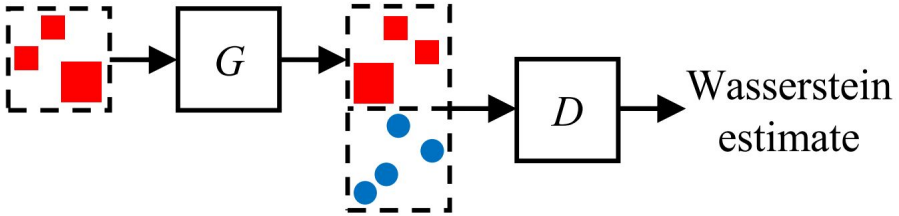
distributions of transformed source and target word embeddings

can replace these with neural network discriminator with weight clipping

$$= \frac{1}{K} \sup_{\|f\|_L \leq K} \mathbb{E}_{y \sim P^T}[f(y)] - \mathbb{E}_{y \sim P^{G(S)}}[f(y)]$$

Wasserstein GAN

Discriminator: assign higher scores to "true" than to "fake" samples



$$\max_D \mathbb{E}_{y\sim P^T}[f_D(y)] - \mathbb{E}_{x\sim P^S}[f_D(Wx)]$$

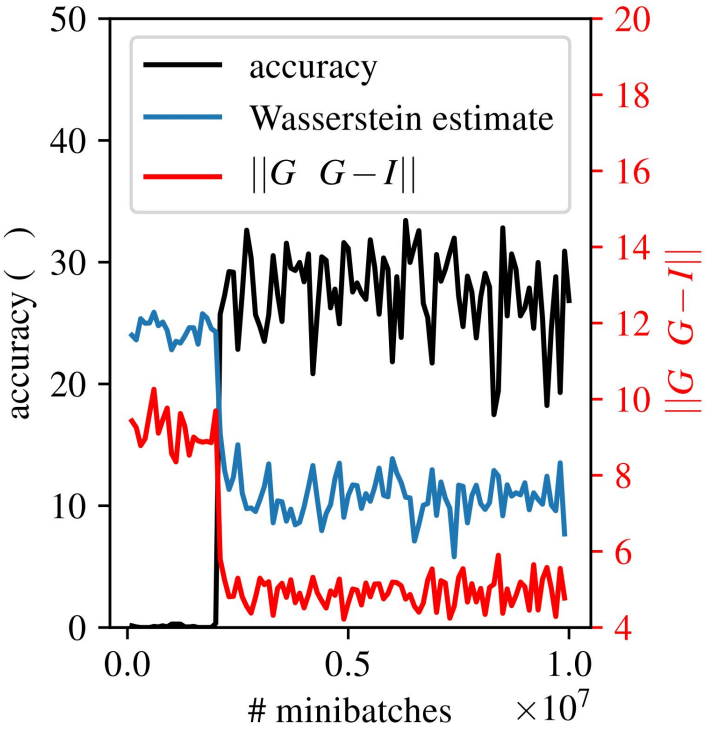
Zhang et al. (2017b)

Generator: minimize approximate Wasserstein distance

$$\min_W -\mathbb{E}_{x\sim P^S}[f_D(Wx)]$$

How to track performance

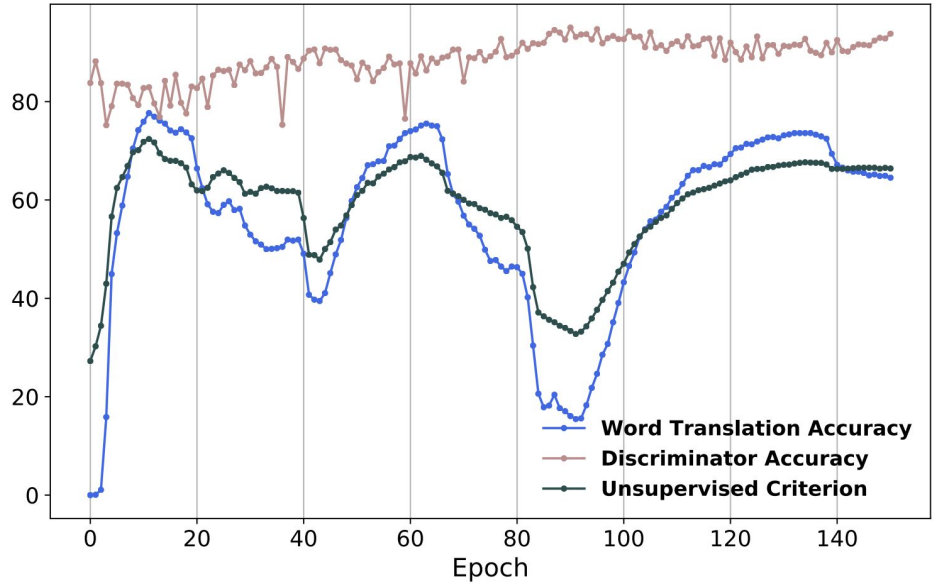
- **Accuracy** is not available in the unsupervised setting
- **Wasserstein estimate**: for free in WGAN, correlates with accuracy
- **Orthogonality of generator**: $\left\|G^\top G - I\right\|_F$ (if orthogonality is not strictly enforced)



Zhang et al. (2017b)

How to track performance

- **Accuracy** is not available in the unsupervised setting
- **Wasserstein estimate**: for free in WGAN, correlates with accuracy
- **Orthogonality of generator**: $\|G^\top G - I\|_F$ (if orthogonality is not strictly enforced)
- **Average cosine distance of translations / reconstruction cost**
- **Number of mutual nearest neighbours**

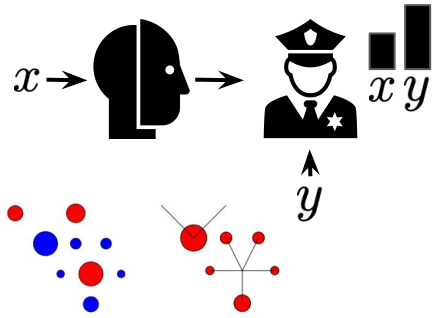


Conneau et al. (2018)

Hoshen & Wolf (2018)

Wasserstein GAN / Optimal Transport

- Wasserstein GAN
- Earth Mover's Distance (EMD)
- EMD and Wasserstein Distance
- Solving the optimal transport problem
- Gromov Wasserstein Distance

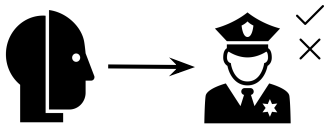
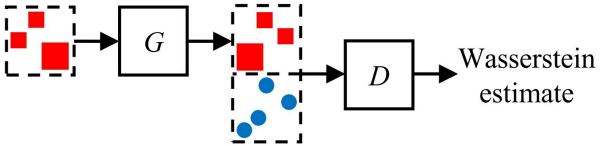
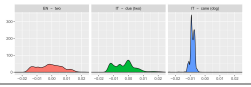


$$d_T^{\lambda}(P_1, P_2) = \min_{T \in \mathcal{U}(p,q)} TC - \frac{1}{\lambda}\mathcal{H}(T)$$

$$GW(C, C', p, q) = \min_{T \in \mathcal{U}(p,q)} \sum_{i,j,k,l} T_{ij} T_{kl} L_{ijkl}$$

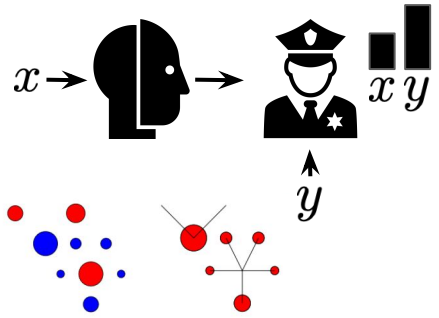
Break

Overview

Authors	Seed dictionary induction	
Barone (2016)	GAN 	Adversarial
Zhang et al. (2017a)		
Conneau et al. (2018)		
Zhang et al. (2017b)	Wasserstein GAN / Optimal transport 	
Xu et al. (2018)		
Alvarez-Melis and Jaakkola (2018)		
Artetxe et al. (2018)	Heuristic 	Non-adversarial
Hoshen and Wolf (2018)	Point Cloud Matching 	

Wasserstein GAN / Optimal Transport

- Wasserstein GAN
- Earth Mover's Distance (EMD)
- EMD and Wasserstein Distance
- Solving the optimal transport problem
- Gromov Wasserstein Distance



$$d_T^{\lambda}(P_1, P_2) = \min_{T \in \mathcal{U}(p,q)} TC - \frac{1}{\lambda}\mathcal{H}(T)$$

$$GW(C, C', p, q) = \min_{T \in \mathcal{U}(p,q)} \sum_{i,j,k,l} T_{ij} T_{kl} L_{ijkl}$$

EMD with orthogonality constraint

In order to incorporate the orthogonality constraint, we need to alternate optimizing the transport matrix T and the translation matrix W at every step k:

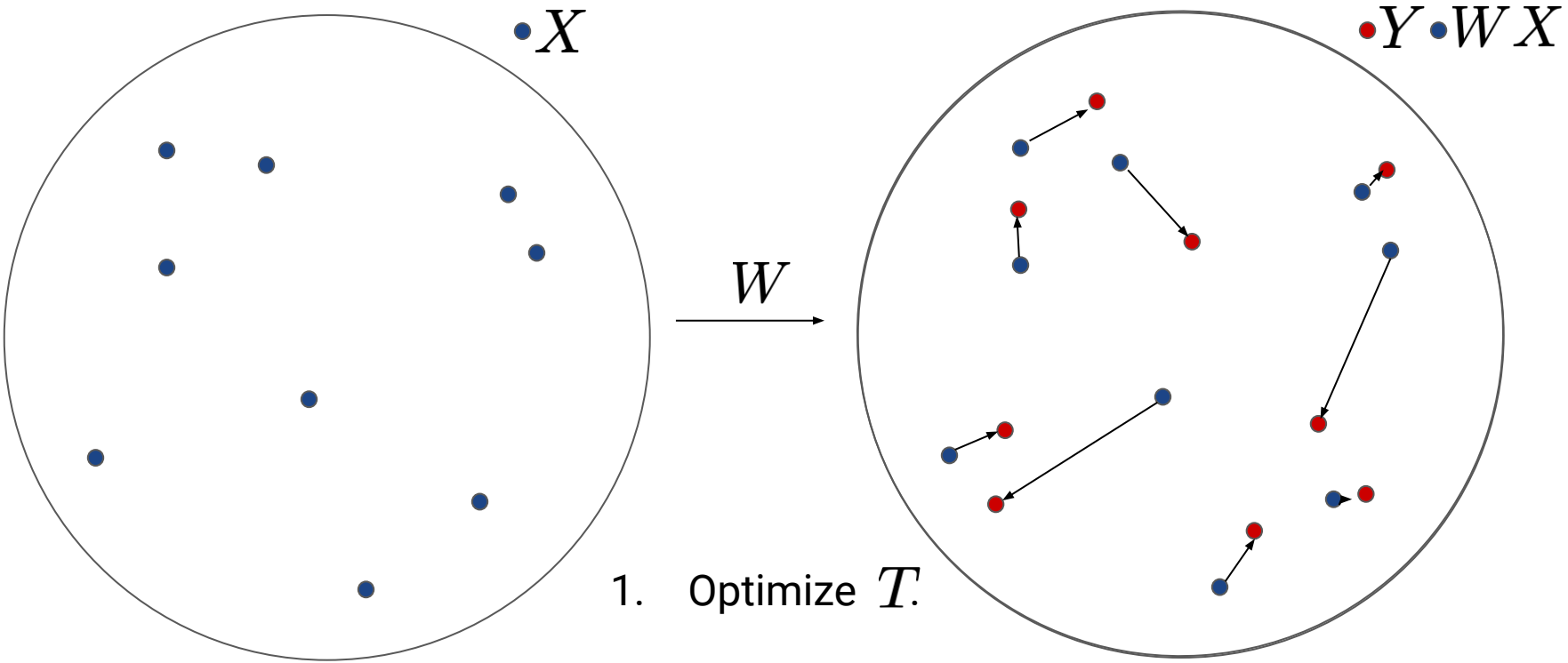
$$T^{(k)} = \underset{T \in \mathcal{U}(f^S, f^T)}{\arg\min} \overbrace{\sum_i \sum_j T_{ij} c(W^{(k)} x^i, y_j)}^{EMD(P^{G(S)}, P^T)}$$

keep this fixed

$$W^{(k+1)} = \underset{W}{\arg\min} \sum_i \sum_j T_{ij}^{(k)} c(W x_i, y_j)$$

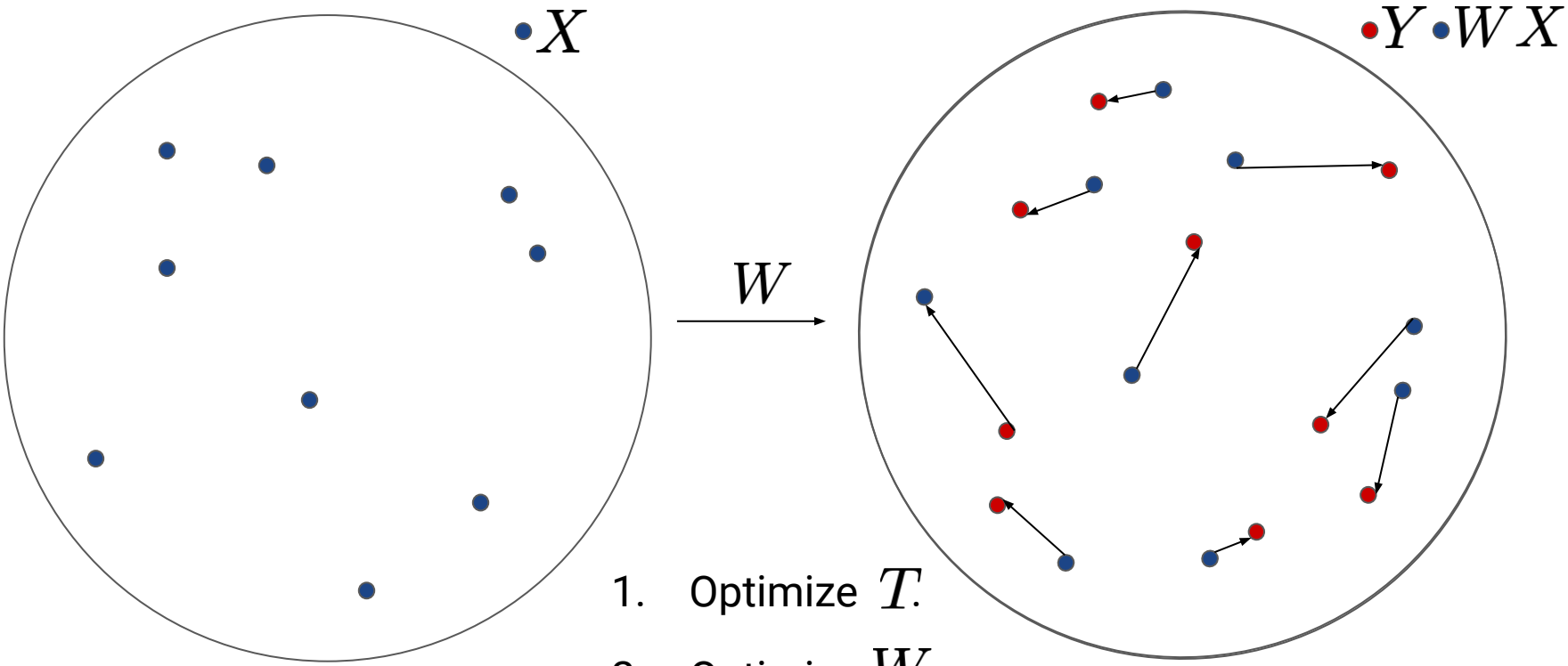
keep this fixed

Alternating optimization of T and W



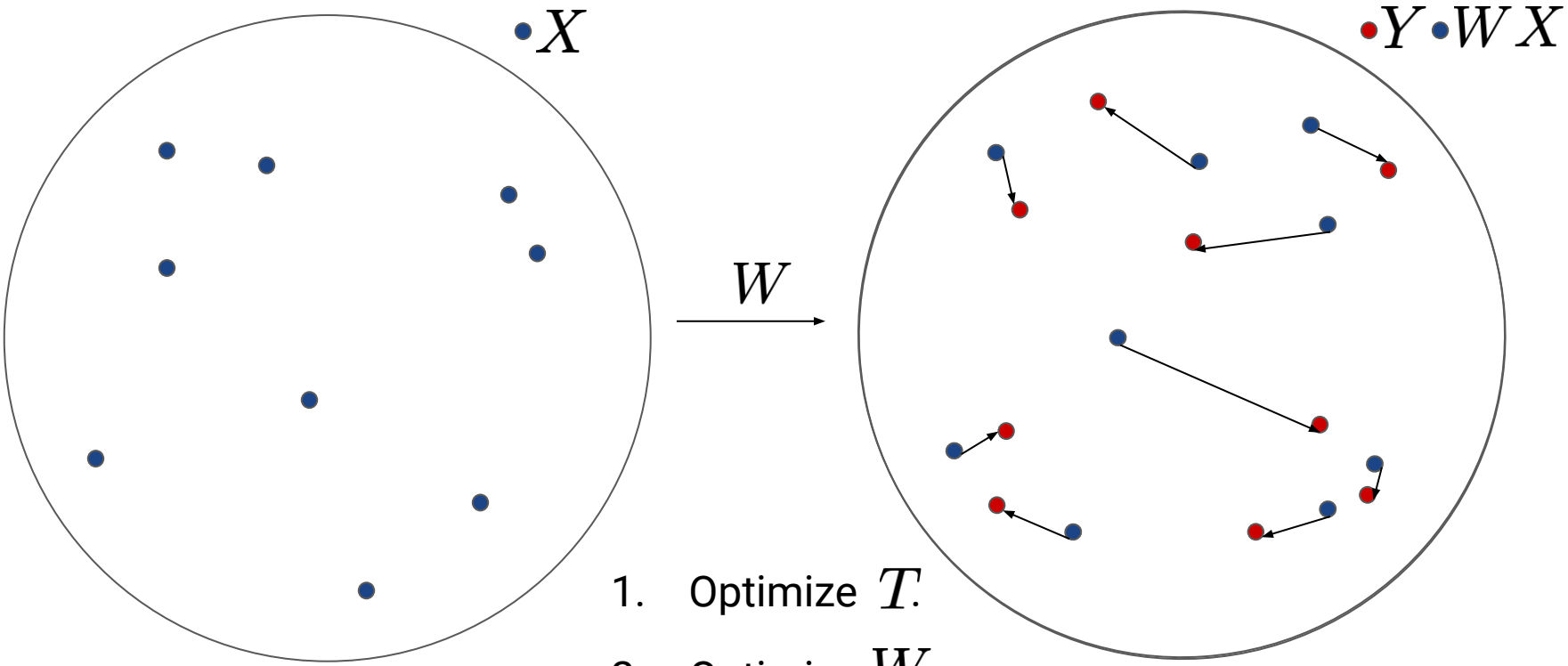
1. Optimize T.

Alternating optimization of T and W



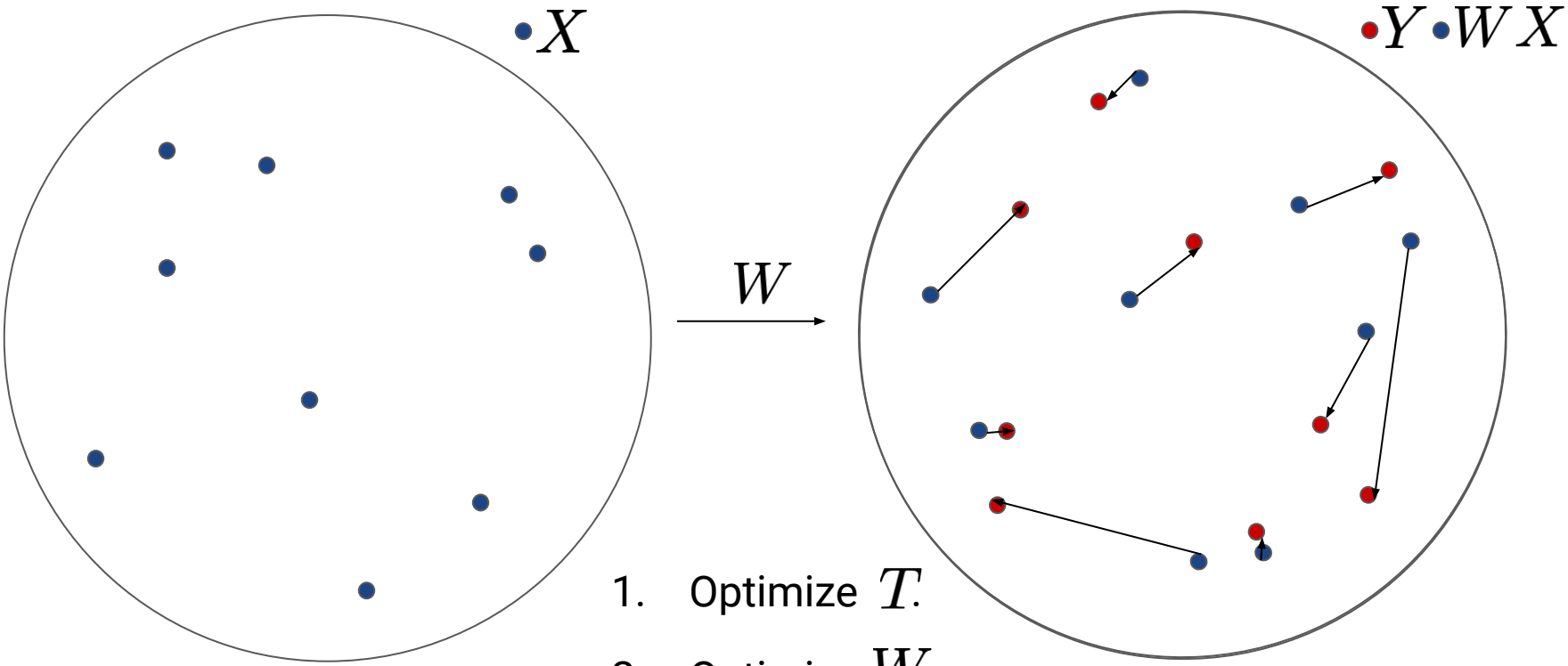
1. Optimize T.
2. Optimize W.

Alternating optimization of T and W



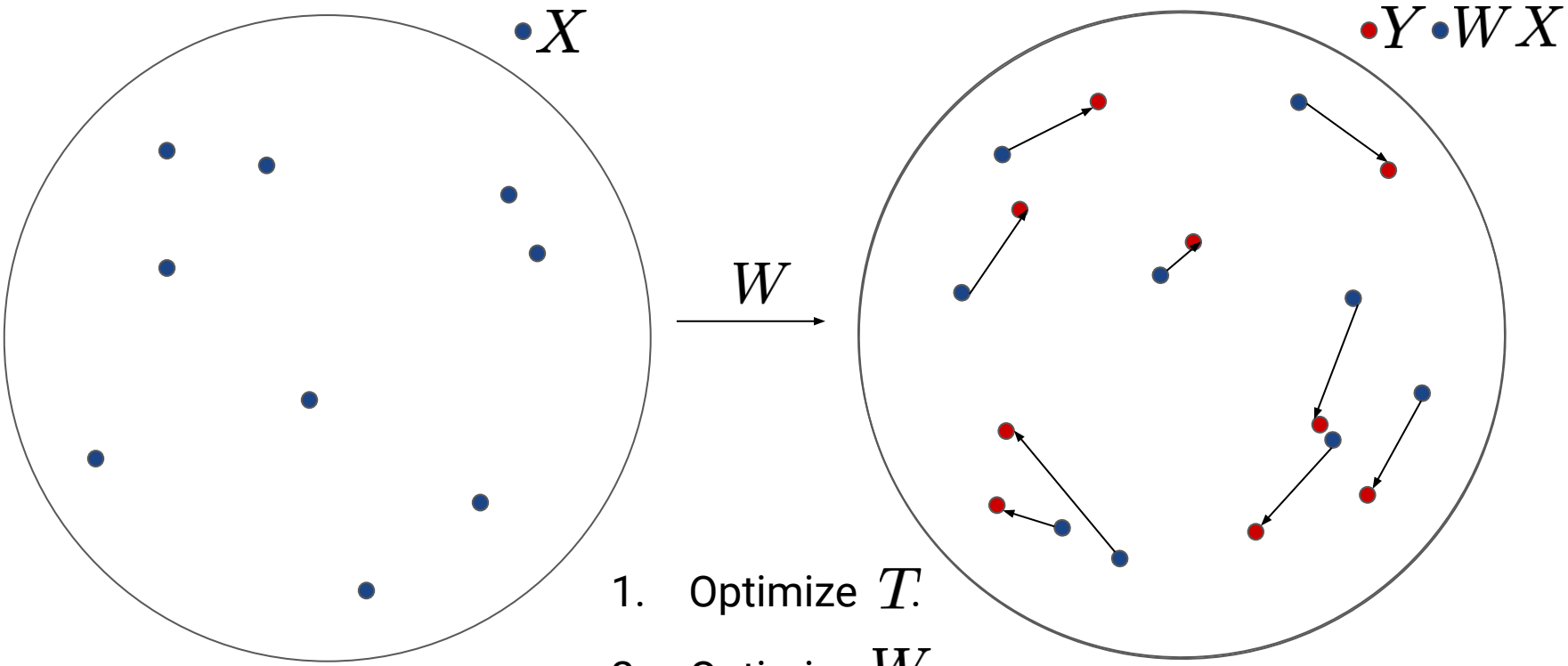
1. Optimize T.
2. Optimize W.

Alternating optimization of T and W



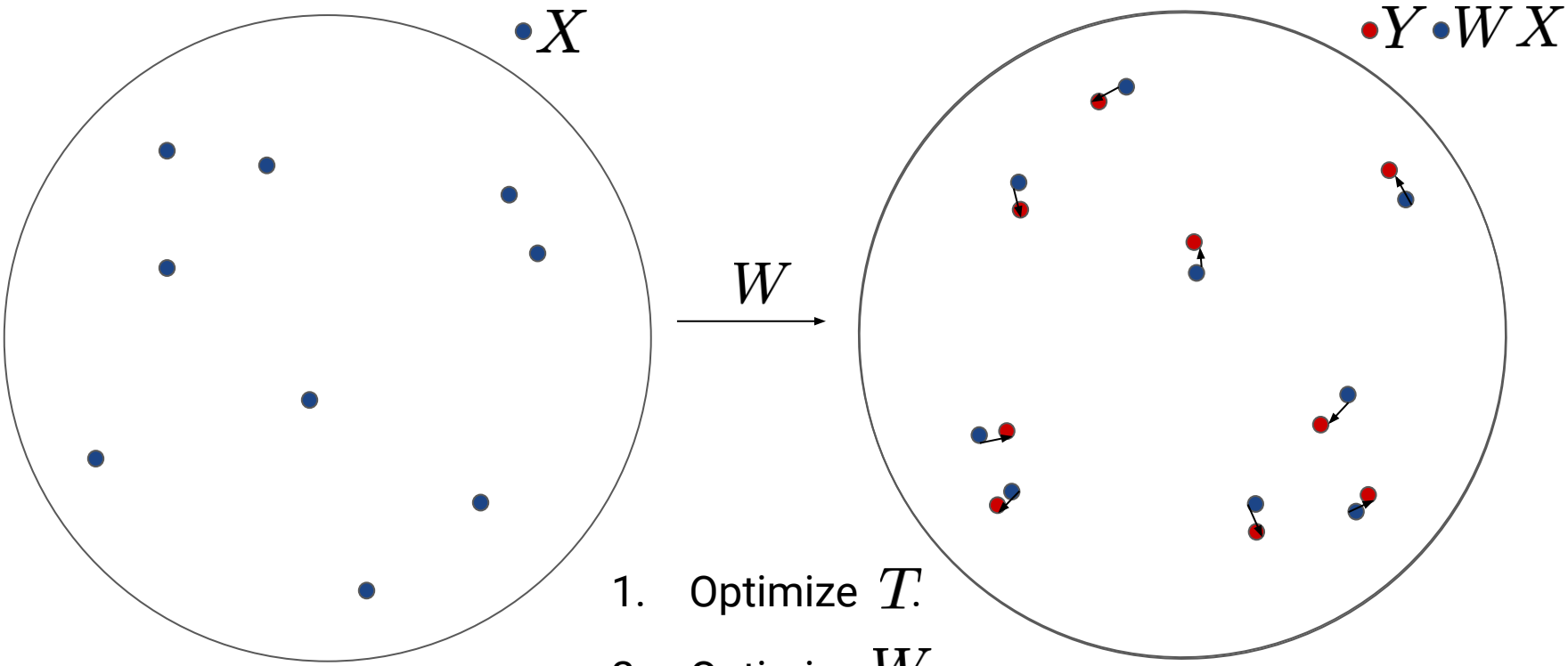
1. Optimize T.
2. Optimize W.

Alternating optimization of T and W



1. Optimize T.
2. Optimize W.

Alternating optimization of T and W



1. Optimize T.
2. Optimize W.

EMD with orthogonality constraint

In order to incorporate the orthogonality constraint, we need to alternate optimizing the transport matrix *T* and the translation matrix *W* at every step *k*:

$$T^{(k)} = \underset{T \in \mathcal{U}(f^S, f^T)}{\arg\min} \overbrace{\sum_i \sum_j T_{ij} c(W^{(k)} x^i, y_j)}^{EMD(P^{G(S)}, P^T)}$$

can be optimized with approximate optimal transport solver (Cuturi, 2013)

$$W^{(k+1)} = \underset{W}{\arg\min} \sum_i \sum_j T_{ij}^{(k)} c(W x_i, y_j)$$

if c is squared L2 distanc
this is just Procrustes
problem

Solving the optimal transport (OT) problem

$$EMD(P_1, P_2) = \min_{T \in \mathcal{U}(p,q)} \sum_i \sum_j T_{ij} c(x_i, y_j)$$

Solving the optimal transport (OT) problem

$$EMD(P_1, P_2) = \min_{T \in \mathcal{U}(p,q)} \underbrace{\sum_i \sum_j T_{ij} C_{ij}}_{TC}$$

Solving the optimal transport (OT) problem

$$EMD(P_1, P_2) = \min_{T \in \mathcal{U}(p,q)} TC$$

Solving the optimal transport (OT) problem

$$EMD(P_1, P_2) = \min_{T \in \mathcal{U}(p,q)} TC - \frac{1}{\lambda}\mathcal{H}(T)$$

Lagrange multiplier

Entropy constraint

Two reasons for adding an entropy constraint (Cuturi, 2013):

1. Enables efficient computation.
2. Encourages smooth solutions.

Solving the optimal transport (OT) problem

$$d_T^{\lambda}(P_1, P_2) = \min_{T \in \mathcal{U}(p,q)} TC - \frac{1}{\lambda}\mathcal{H}(T)$$

Known as Sinkhorn distance ("dual-Sinkhorn divergence" specifically)

Lagrange multiplier

Entropy constraint

Solution has the form (Cuturi, 2013):

$$T^* = \mathbf{diag}(u) K \mathbf{diag}(v)$$

where u and v are non-negative vectors and $K = e^{-\lambda C}$.

Solving the optimal transport (OT) problem

Can be efficiently computed with using only matrix-vector multiplication.
→ can back-propagate through it

Solution has the form (Cuturi, 2013):
$T^* = \mathbf{diag}(u) K \mathbf{diag}(v)$
where u and v are non-negative vectors and $K = e^{-\lambda C}$.

Algorithm 1 Computation of Sinkhorn Distance $d_{sh}(G)$

procedure SINKHORN($M^{(G)}, r, c, \lambda, I$)
  $K^{(G)} := e^{-\lambda M^{(G)}}$
  $v = \mathbb{1}_m / m$ ▷ normalized one vector
  $i = 0$
  while $i < I$ **do** ▷ iterate for I times
    $u = r./K^{(G)} v$
    $v = c./{K^{(G)}}^T u$
    $i = i + 1$
  $d_{sh}(G) = u^T((K^{(G)} \otimes M^{(G)})v)$
  return $d_{sh}(G)$ ▷ The Sinkhorn distance

Zhang et al. (2017b)

Solving the optimal transport (OT) problem

$$d_T^{\lambda}(P_1, P_2) = \min_{T \in \mathcal{U}(p,q)} TC - \frac{1}{\lambda}\mathcal{H}(T)$$

Distance should be a valid metric, e.g. Euclidean distance

Cosine distance $1 - cos(a, b)$ is not a valid metric (does not satisfy triangle inequality).

→ Xu et al. (2018) use square root cosine distance $\sqrt{2 - 2cos(a, b)}$ instead.

Gromov Wasserstein distance

Main idea: Compare metric spaces directly (instead of just comparing samples) by comparing distances between pairs of points (*distance between distances*).

$$EMD(P_1, P_2) = \min_{T \in \mathcal{U}(p,q)} \sum_i \sum_j T_{ij} C_{ij}$$

$$GW(C, C', p, q) = \min_{T \in \mathcal{U}(p,q)} \sum_{i,j,k,l} T_{ij} T_{kl} L(C_{ik}, C'_{jl})$$

measures cost of matching x_i to y_j and x_k to y_l

Alvarez-Melis & Jaakkola (2018)

Gromov Wasserstein distance

Main idea: Compare metric spaces directly (instead of just comparing samples) by comparing distances between pairs of points (*distance between distances*).

$$EMD(P_1, P_2) = \min_{T \in \mathcal{U}(p,q)} \sum_i \sum_j T_{ij} C_{ij}$$

$$GW(C, C', p, q) = \min_{T \in \mathcal{U}(p,q)} \sum_{i,j,k,l} T_{ij} T_{kl} L_{ijkl}$$

requires operating over a fourth-order tensor!

Can be optimized efficiently with first-order methods (Peyré et al., 2016)

Alvarez-Melis & Jaakkola (2018)

Gromov Wasserstein distance

$$GW(C, C', p, q) = \min_{T \in \mathcal{U}(p,q)} \sum_{i,j,k,l} T_{ij} T_{kl} L_{ijkl}$$

Compute a pseudo-cost matrix:

$$\hat{C} = C_{st} - 2 C_s T C_t^\top$$

cross-language similarity
$C_{st} = C_s^2 p \mathbf{1}^\top + \mathbf{1} q (C_t^2)^\top$

intra-language similarities
$C_s = \cos(X, X)$ and $C_t = \cos(Y, Y)$

Repeat this process until convergence.

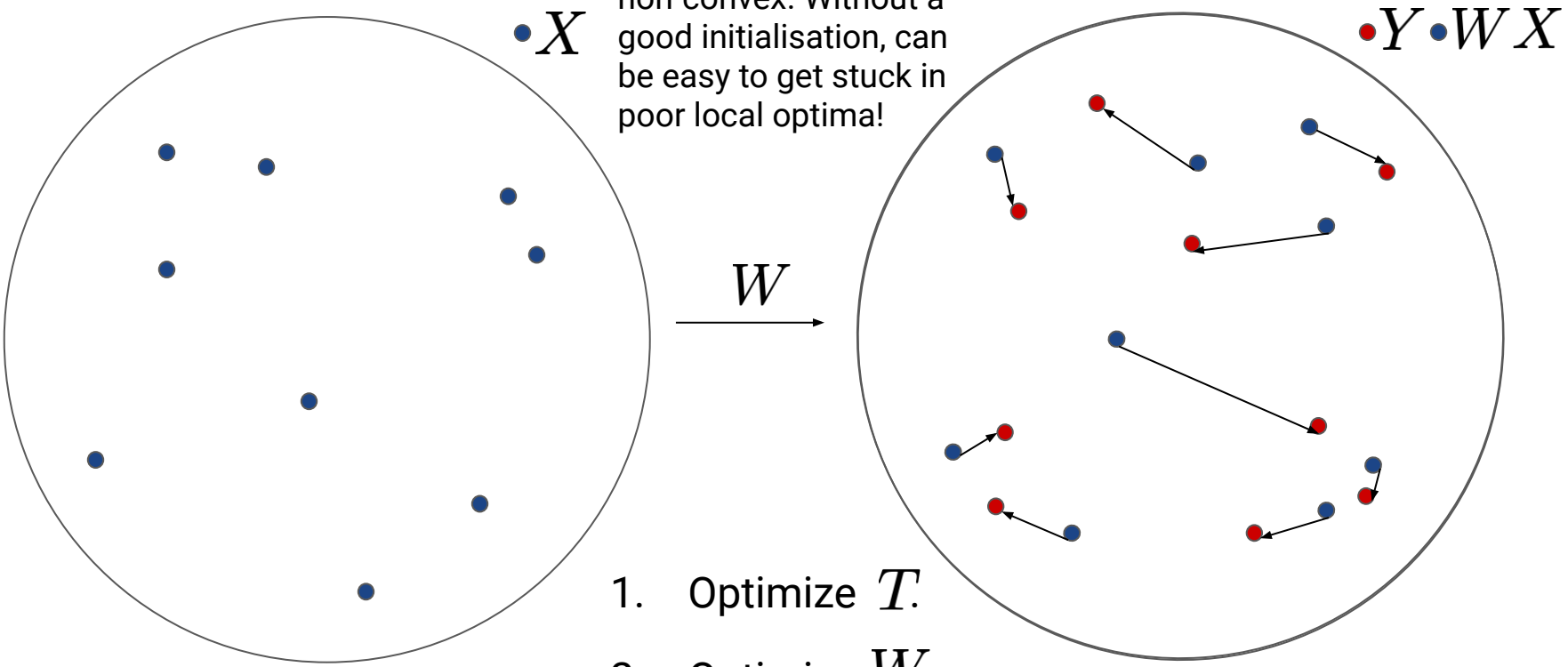
Solve optimal transport problem using $\hat{C}$ as cost matrix:

$$d_T^\lambda(P_1, P_2) = \min_{T \in \mathcal{U}(p,q)} T\hat{C} - \frac{1}{\lambda} \mathcal{H}(T)$$

Alvarez-Melis & Jaakkola (2018)

Finding a good initialisation

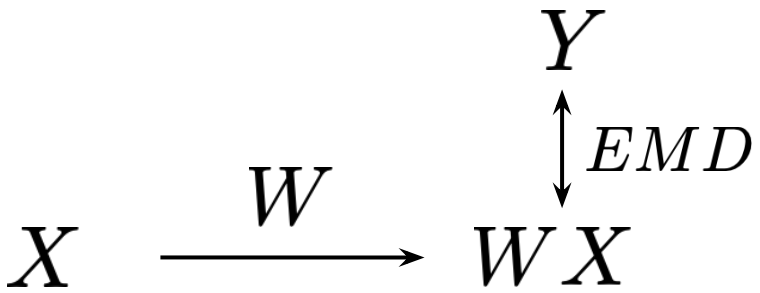
Problem is non-convex. Without a good initialisation, can be easy to get stuck in poor local optima!



1. Optimize T.
2. Optimize W.

Finding a good initialisation

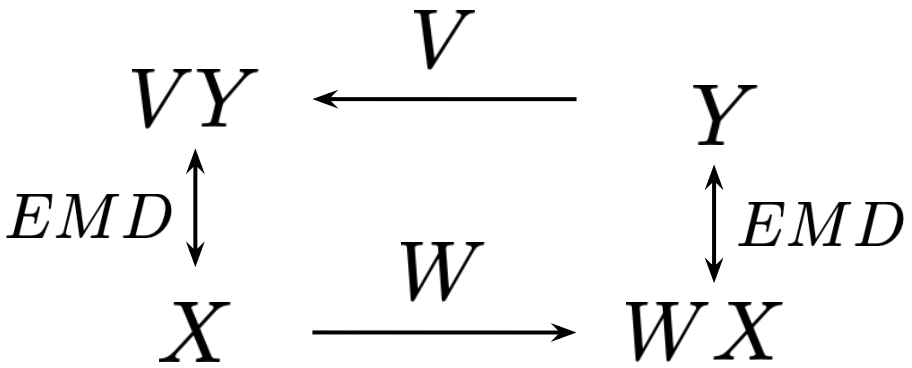
In practice, first use WGAN to find a good initialisation and then solve optimal transport problem with orthogonality constraint.



Zhang et al. (2017b)

Bidirectionality

- Can do projection in both ways.
- V can be a new transformation or $W^\top$.
- Can use a GAN instead of EMD.



Zhang et al. (2017a), Xu et al. (2018)

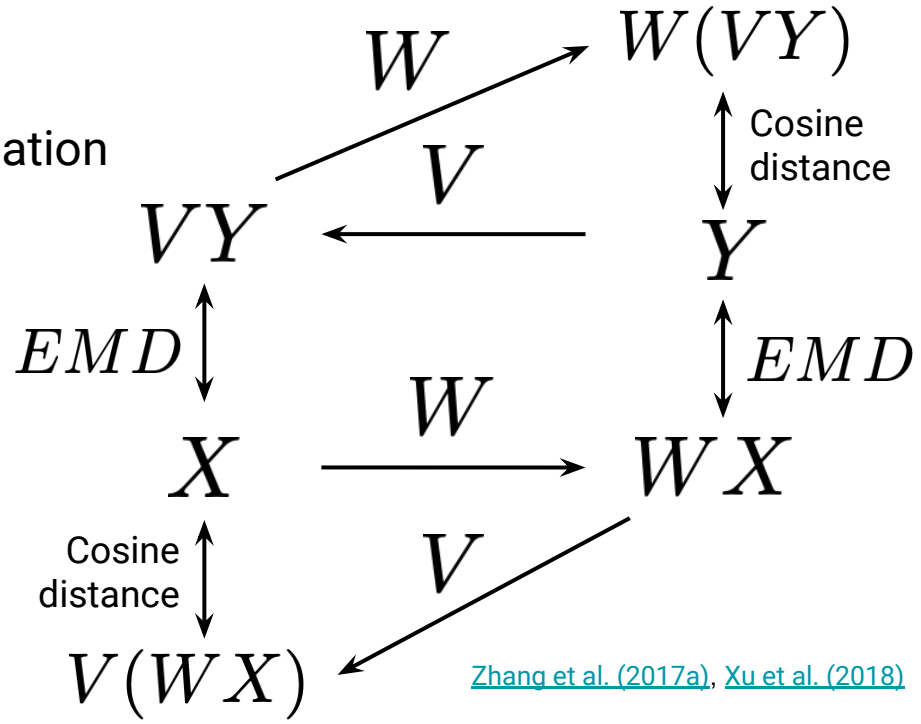
Reconstruction

- Enforce consistency by reconstructing the projected embeddings.
- Can also be seen as adversarial autoencoder, CycleGAN, or back-translation

Reconstruction loss

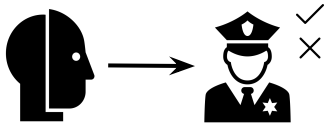
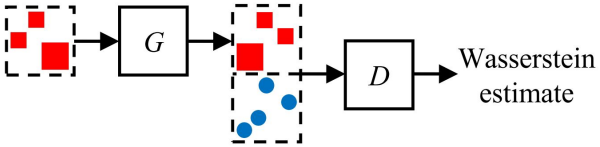
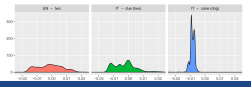
Minimise cosine distance between original and twice projected embeddings:

$$\arg\min_{V,W} \sum_i 1 - \cos(x_i, V(Wx_i)) + \sum_j 1 - \cos(y_j, W(Vy_i))$$



Zhang et al. (2017a), Xu et al. (2018)

Overview

Authors	Seed dictionary induction	
Barone (2016)	GAN 	Adversarial
Zhang et al. (2017a)		
Conneau et al. (2018)		
Zhang et al. (2017b)	Wasserstein GAN / Optimal transport 	
Xu et al. (2018)		
Alvarez-Melis and Jaakkola (2018)		
Artetxe et al. (2018)	Heuristic 	Non-adversarial
Hoshen and Wolf (2018)	Point Cloud Matching 	

Heuristic seed induction

We have seen heuristics for specifying a seed dictionary (“same spelling”, “numerals”) but these make strong assumptions about the writing system.

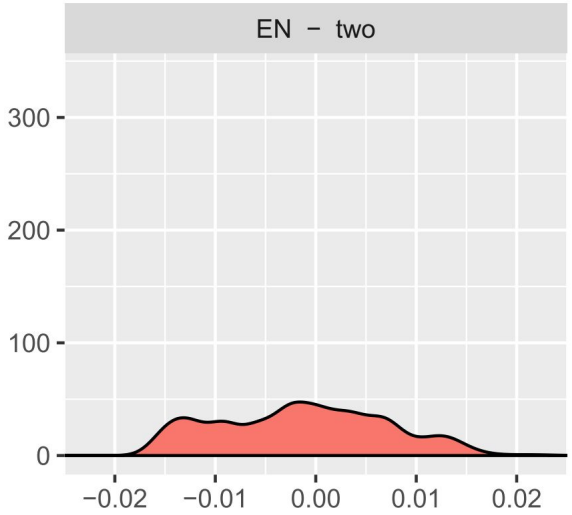
Can we come up with a heuristic that is independent of the writing system?

Heuristic seed induction

Artetxe et al. (2018)

Main idea: Translations are similar to other words in the same way across languages.

Specifically: Words with similar meaning have similar monolingual similarity distributions (i.e. distributions of similarity across all words of the same language).




```
for x in vocab:
  sim(x, "two")
```

intra-language similarity

Gromov-Wasserstein distance (Alvarez-Melis & Jaakkola, 2018) incorporates this, too

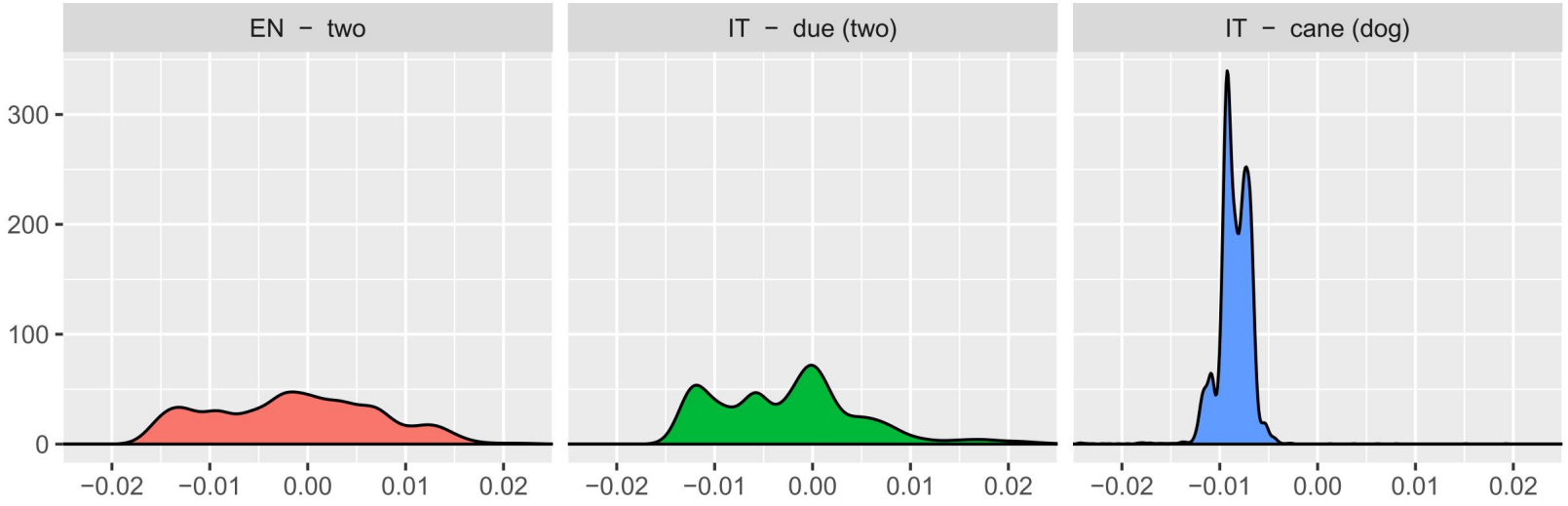
Similar to going from "*distance*" to "*distance between distances*", we now go from "*similarity*" to "*similarity between similarities*"…

Heuristic seed induction

Artetxe et al. (2018)

Main idea: Translations are similar to other words in the same way across languages.

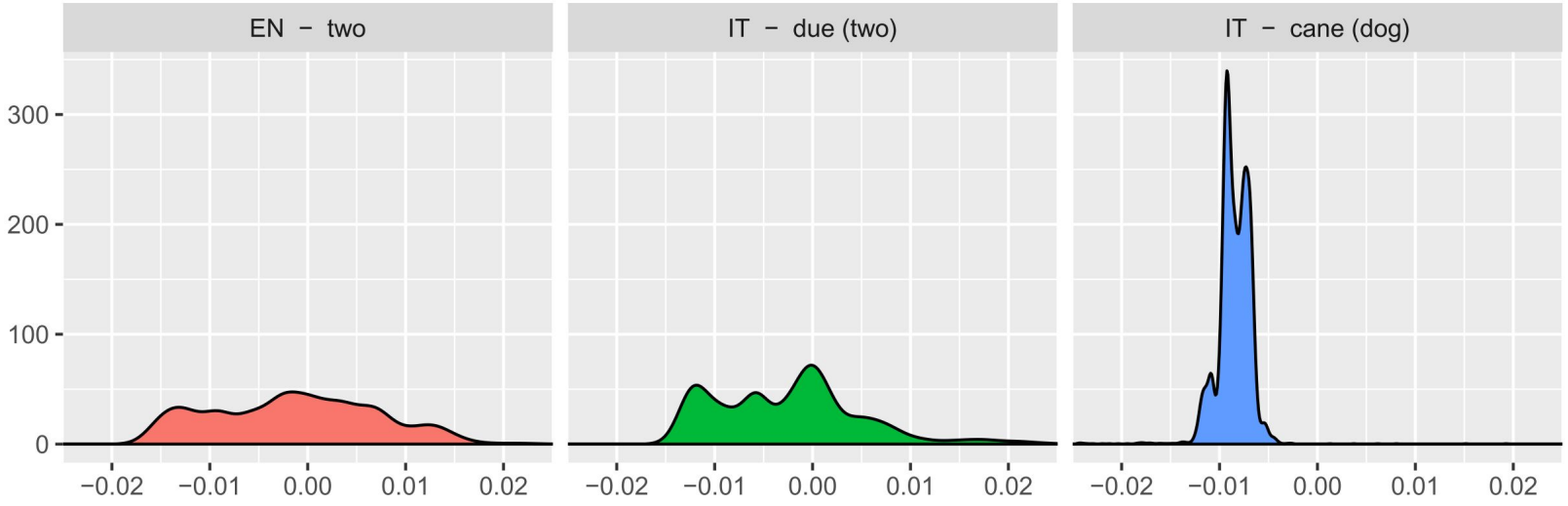
Specifically: Words with similar meaning have similar monolingual similarity distributions (i.e. distributions of similarity across all words of the same language).



Heuristic seed induction

Artetxe et al. (2018)

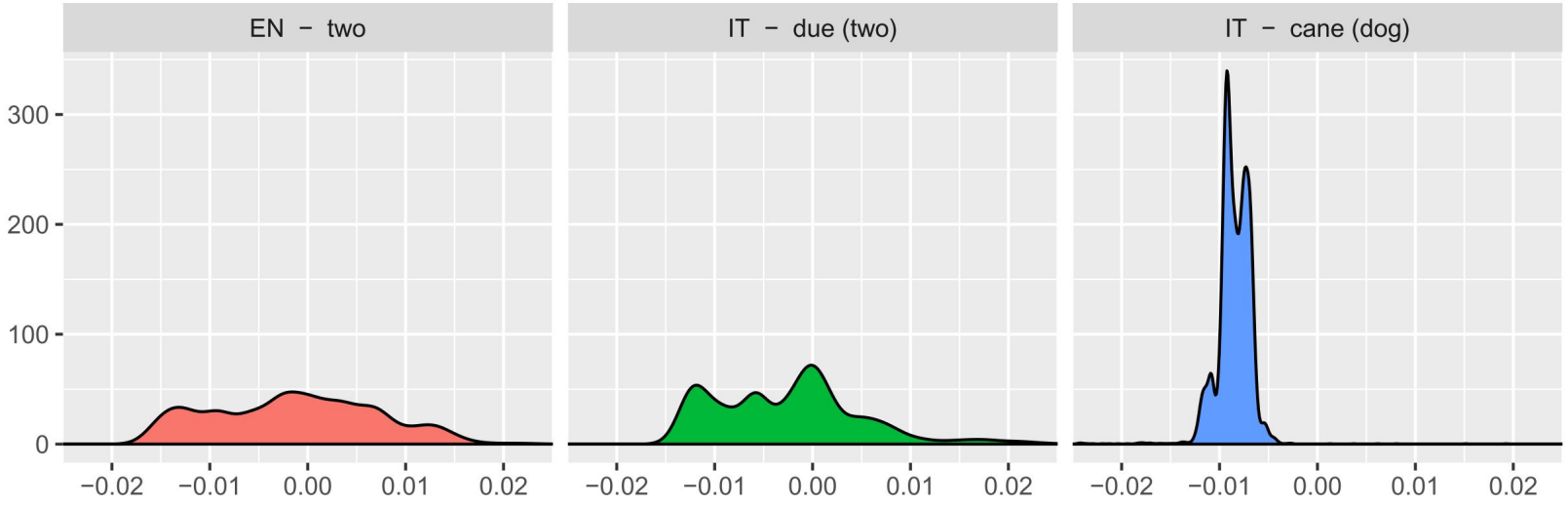
- Get similarity matrices $M_X = XX^\top$ and $M_Y = YY^\top$ for both languages
- Sort values in each row of M_X and M_Y to get similarity distribution for every word; take square root to get smoothed density estimate



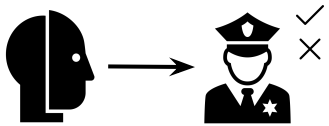
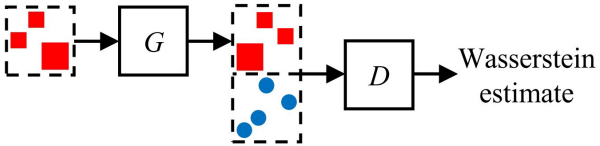
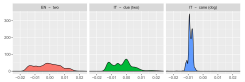
Heuristic seed induction

Artetxe et al. (2018)

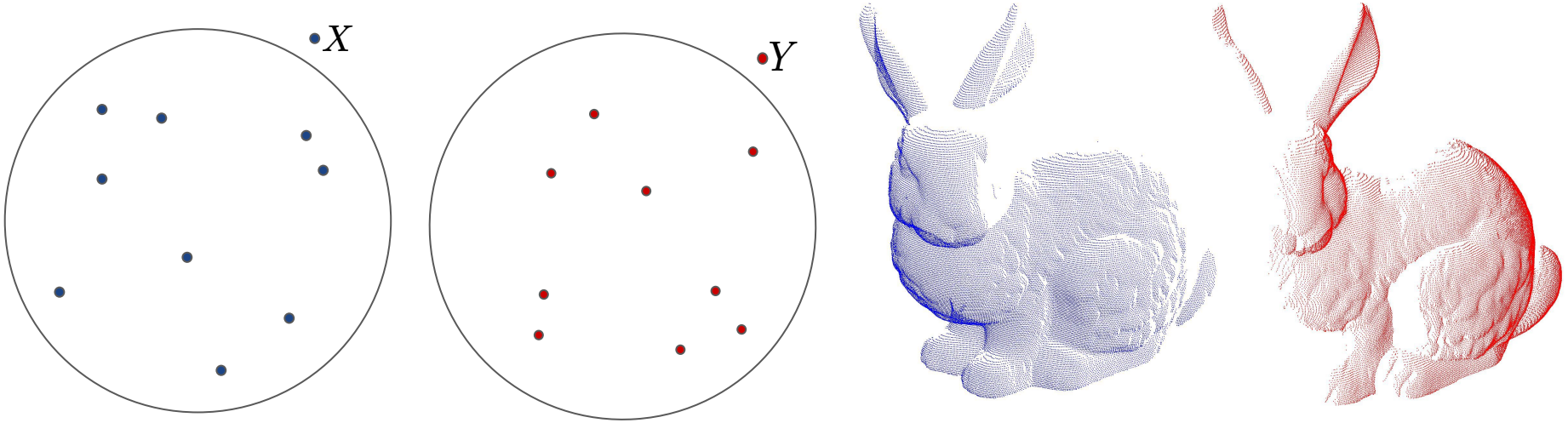
- Nearest neighbours from similarity distributions used as initial dictionary
- Additional tricks to make self-learning more robust



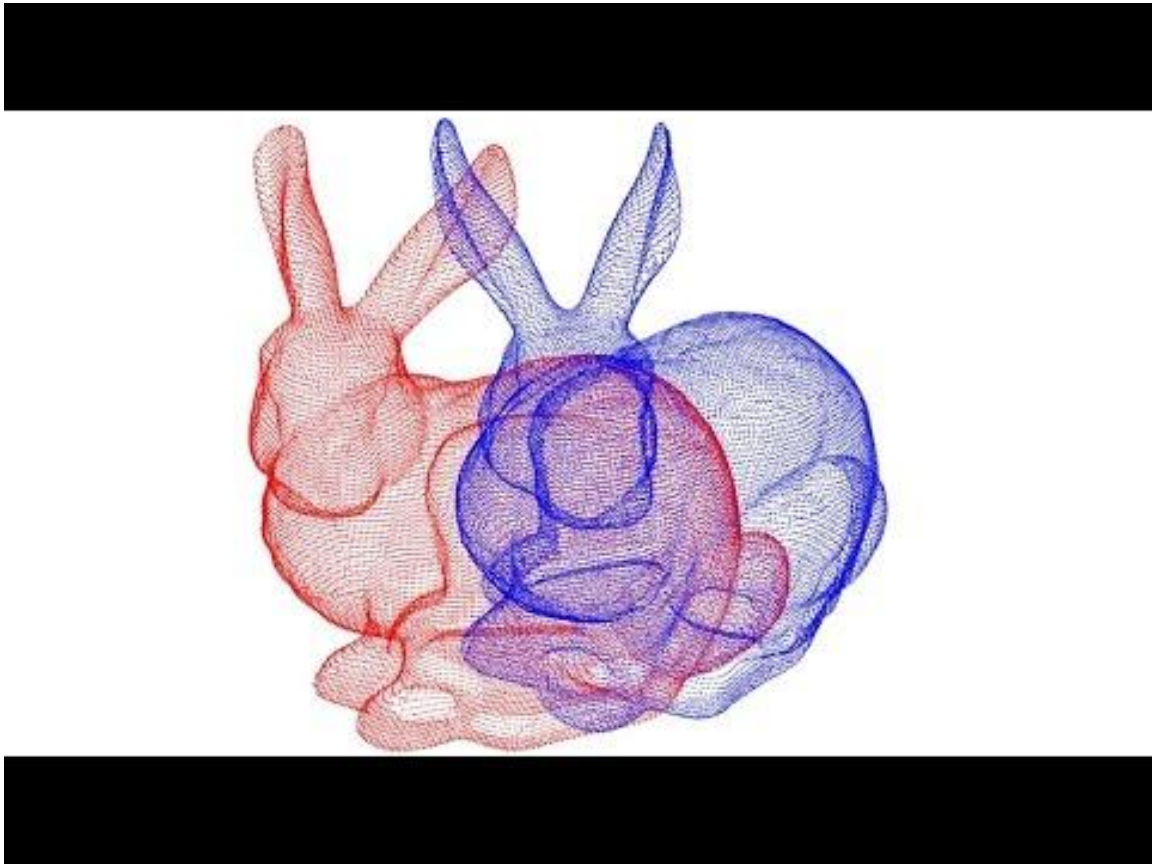
Overview

Authors	Seed dictionary induction	
Barone (2016)	GAN 	Adversarial
Zhang et al. (2017a)		
Conneau et al. (2018)		
Zhang et al. (2017b)	Wasserstein GAN / Optimal transport 	
Xu et al. (2018)		
Alvarez-Melis and Jaakkola (2018)		
Artetxe et al. (2018)	Heuristic 	Non-adversarial
Hoshen and Wolf (2018)	Point Cloud Matching 	

Word embedding spaces as point clouds

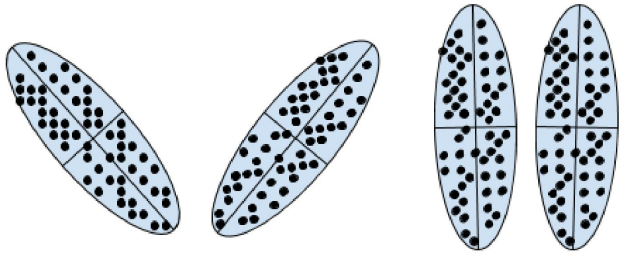


Point cloud matching



Approximate distribution alignment via PCA

- Project embeddings to principal axes of variation via PCA
- Then solve this easier problem first via self-learning
- Use the solution as initialization for solving the original problem
- PCA-based alignment is popular in point cloud matching



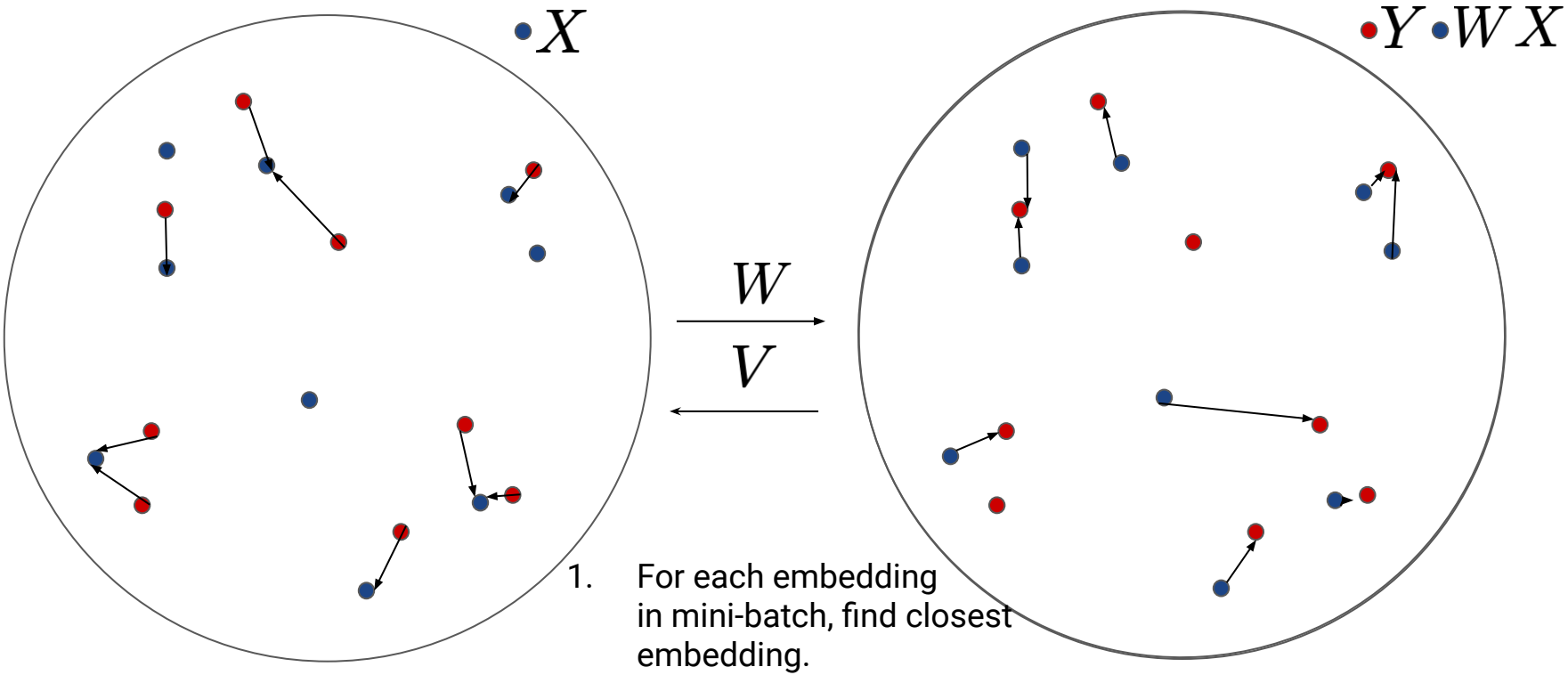
Iterative closest point (ICP)

ICP is very similar to our self-learning loop:

1. For each point in source space, find closest point in target space.
2. Estimate transformation that best aligns (minimizes distance between) source point and its match found in previous step.
3. Transform source points with estimated transformation.
4. Iterate.

Hoshen & Wolf (2018) do this in mini-batches, add bidirectionality + reconstruction.

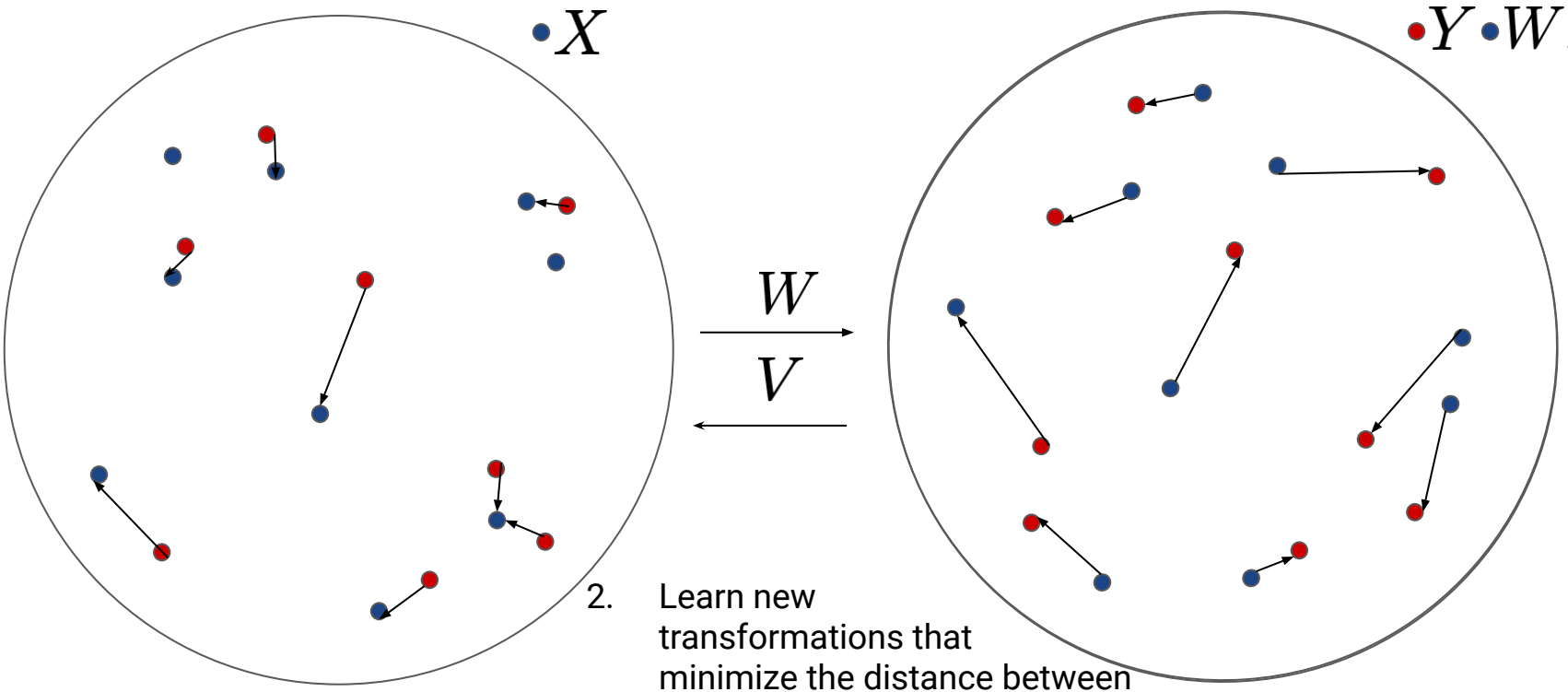
Mini-Batch Cycle Iterative Closest Point (MBC-ICP)



1. For each embedding in mini-batch, find closest embedding.
   Do this for both languages.

Hoshen & Wolf (2018)

Mini-Batch Cycle Iterative Closest Point (MBC-ICP)



2. Learn new transformations that minimize the distance between matched embeddings.

Hoshen & Wolf (2018)

Mini-Batch Cycle Iterative Closest Point (MBC-ICP)

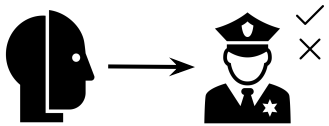
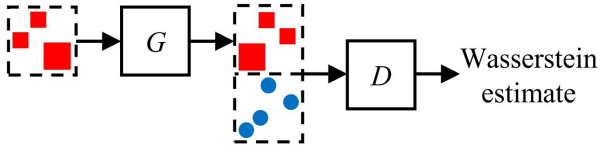
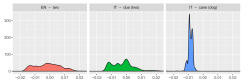
Initialization is important!

In practice, MBC-ICP is run three times:

1. On 5k most frequent words mapped to principal components.
2. On all words.
3. On mutual nearest neighbours (7.5k words).

Hoshen & Wolf (2018)

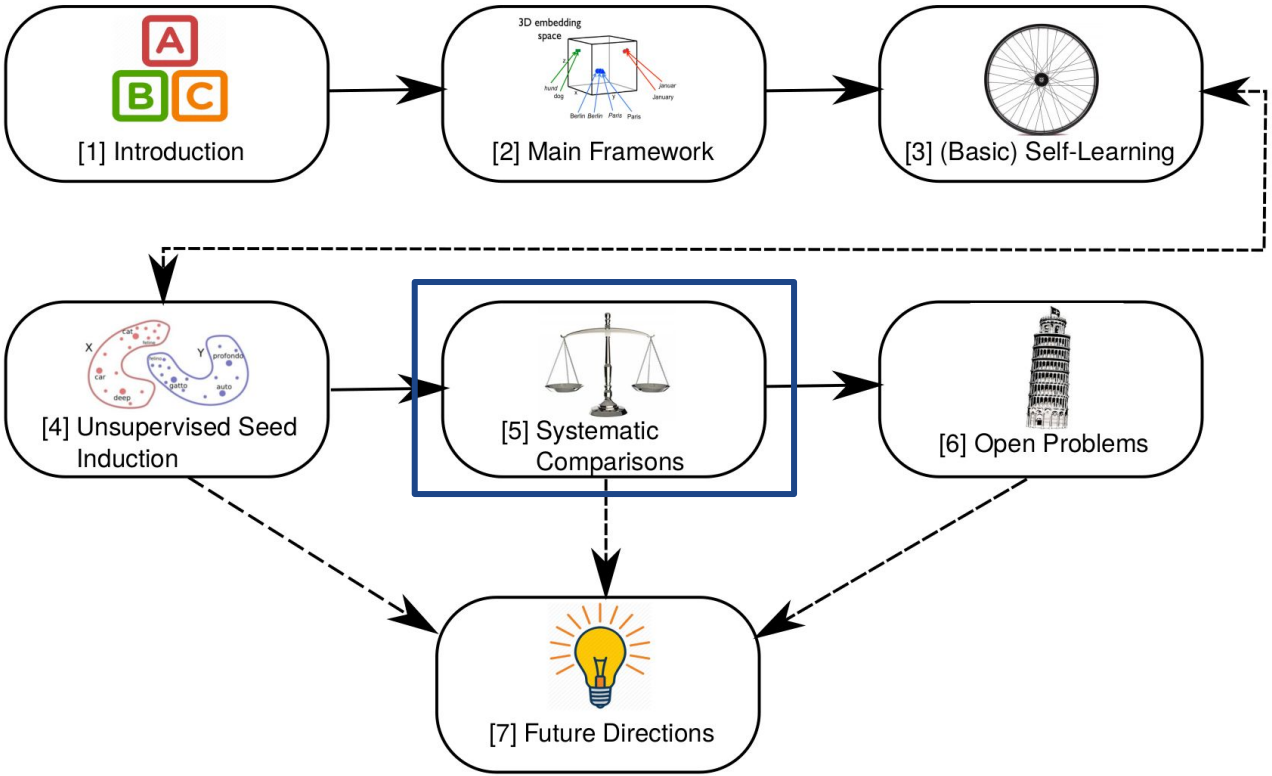
Overview

Authors	Seed dictionary induction
Barone (2016)	GAN 
Zhang et al. (2017a)	
Conneau et al. (2018)	
Zhang et al. (2017b)	Wasserstein GAN / Optimal transport 
Xu et al. (2018)	
Alvarez-Melis and Jaakkola (2018)	
Artetxe et al. (2018)	Heuristic 
Hoshen and Wolf (2018)	Point Cloud Matching 

Take-aways

- Existing methods for unsupervised seed induction share many commonalities:
  - Adversarial term
  - Optimal transport
  - Computing intra-language similarity
  - Iteratively optimizing a distance metric
- Learning an unsupervised alignment of word representations is a general problem
- Inspiration can come from many different domains:
  - Transportation theory
  - Computer vision
  - ...

5. Systematic Comparisons



State of the art

- New papers compare to previous work by reporting numbers of their systems on the same datasets.
- This systematically compares multi-component systems, but not modeling choices.
- We know VecMap > MUSE, but not whether heuristic initialization is better than GANs, for example.

State of the art

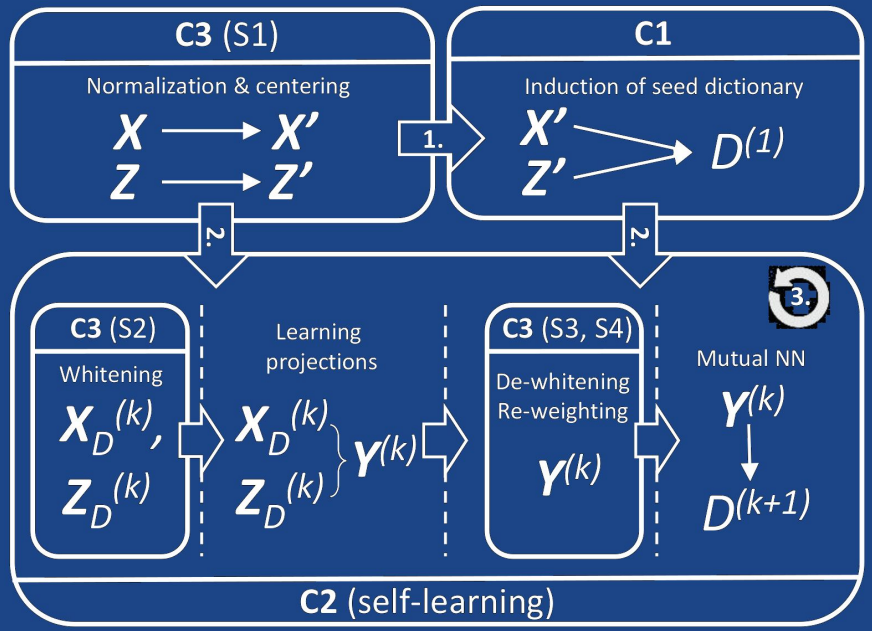
Common wisdom	Unsupervised is better than supervised (unless you have several thousands of good seeds)?
	VecMap (heuristic initialization and stochastic dictionary induction) is better than MUSE (GANs and Procrustes Analysis)?
Our message	

State of the art

Common wisdom	Unsupervised is better than supervised (unless you have several thousands of good seeds)?
	VecMap (heuristic initialization and stochastic dictionary induction) is better than MUSE (GANs and Procrustes Analysis)?
Our message	Unsupervised is mostly worse than supervised. Stochastic dictionary induction is a key component in the self-learning step. GANs are competitive with heuristic (second-order) initialization.

Reminder

- The difference between unsupervised and supervised approaches to cross-lingual learning is **how we obtain our seed alignments** (typically a dictionary).
- The choice of iterative refinement method is therefore **orthogonal** to the source of the seed alignments.



Footnote on Stochasticity and Optimization

Procrustes Analysis overfits the seed

- Procrustes Analysis (PA) minimizes the squared Euclidean distance between the seed points. If these are sampled at random, we minimize the expected overall distance.
- **Problem:** Seeds are not selected at random.

$$\mathbf{W}_{L1} = \mathbf{U}\mathbf{V}^\top, \text{ with}$$
$$\mathbf{U}\mathbf{\Sigma}\mathbf{V}^\top = SVD(\mathbf{X}_T\mathbf{X}_S{}^\top)$$

PA overfits the seed

- **Supervised:** Our dictionaries may be biased toward frequent words, common nouns, etc.
- **Unsupervised:** Our dictionaries are *both* biased and error-prone.

PA overfits the seed

- **Supervised:** Our dictionaries may be biased toward frequent words, common nouns, etc.
- **Unsupervised:** Our dictionaries are *both* biased and error-prone.
- While Lubin et al. (2019) have shown PA is relatively robust, they only investigated moderate noise levels.

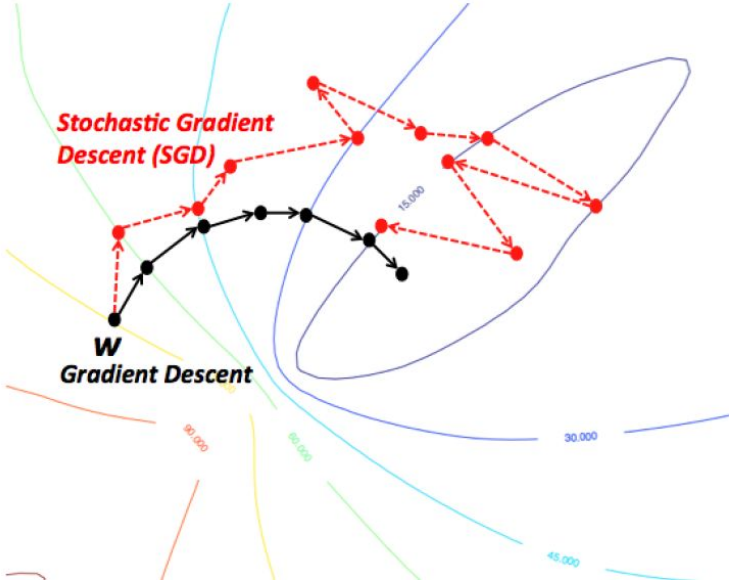
PA overfits the seed

- **Supervised:** Our dictionaries may be biased toward frequent words, common nouns, etc.
- **Unsupervised:** Our dictionaries are *both* biased and error-prone.
- While Lubin et al. (2019) have shown PA is relatively robust, they only investigated moderate noise levels.

- This may be why stochastic dictionary induction (i.e. dropout on current dictionary), a simple regularization technique (Artetxe et al., 2018) led to huge improvements over the state of the art.
- The variance induced by the regularization may prevent the model from getting stuck in poor local optima.

Reminder: Why SGD works so well

- Both GD and SGD iteratively update a set of parameters.
- Both GD and SGD iteratively update a set of parameters. **GD:** All parameters at once. **SGD:** Random samples (minibatches).
- **GD:** Great for convex, relatively smooth manifolds, but often trapped into local minima; the turbulence of **SGD** can get you out of such minima.



Reminder: Why SGD works so well

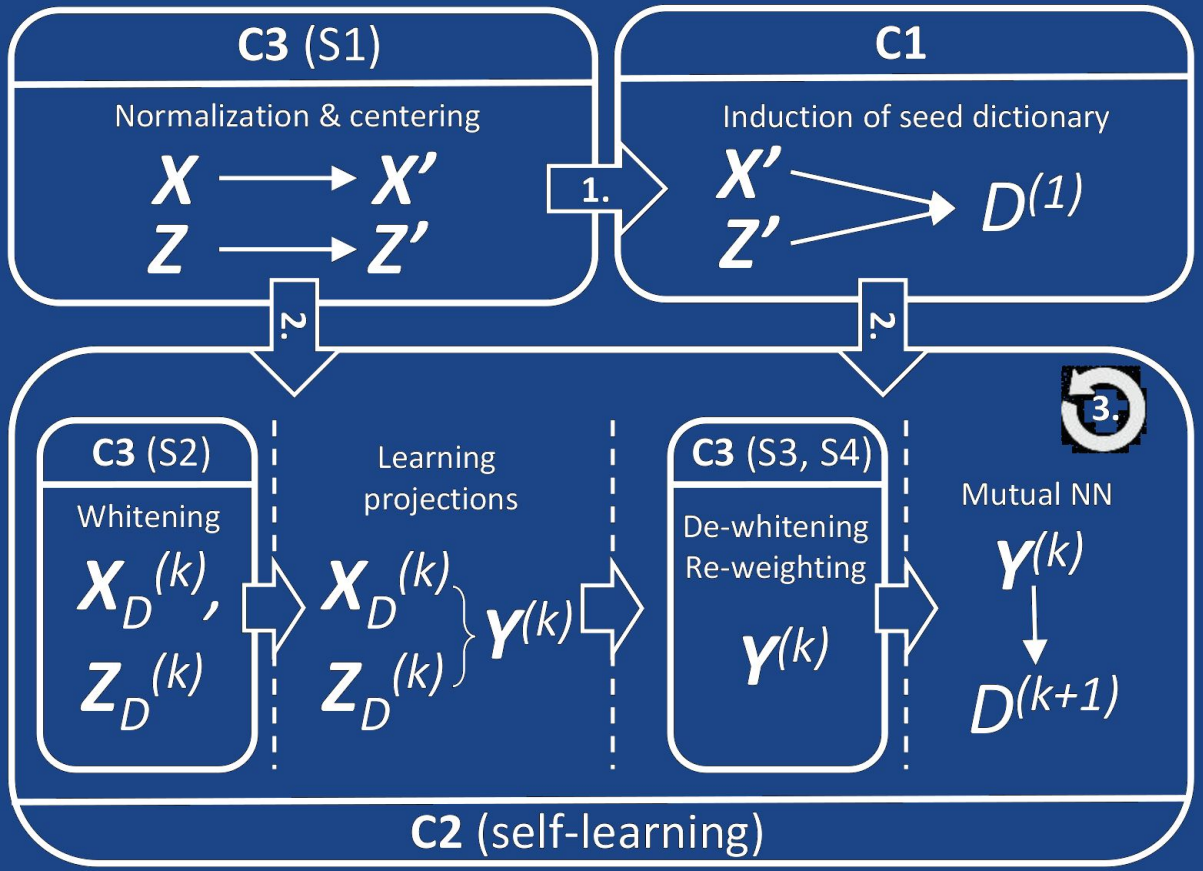
- SGD induces a strong inductive bias, provably learns a network close to the random initialization and with a *small generalization error (Li and Liang, 2018); also in the context of matrix factorization (Gunasekar et al., 2017), which can be used for cross-lingual learning (Zou et al., 2013). Drop-out and noise injection is extremely important for stable training of GANs (Chintala, 2016).

Reminder: Why SGD works so well

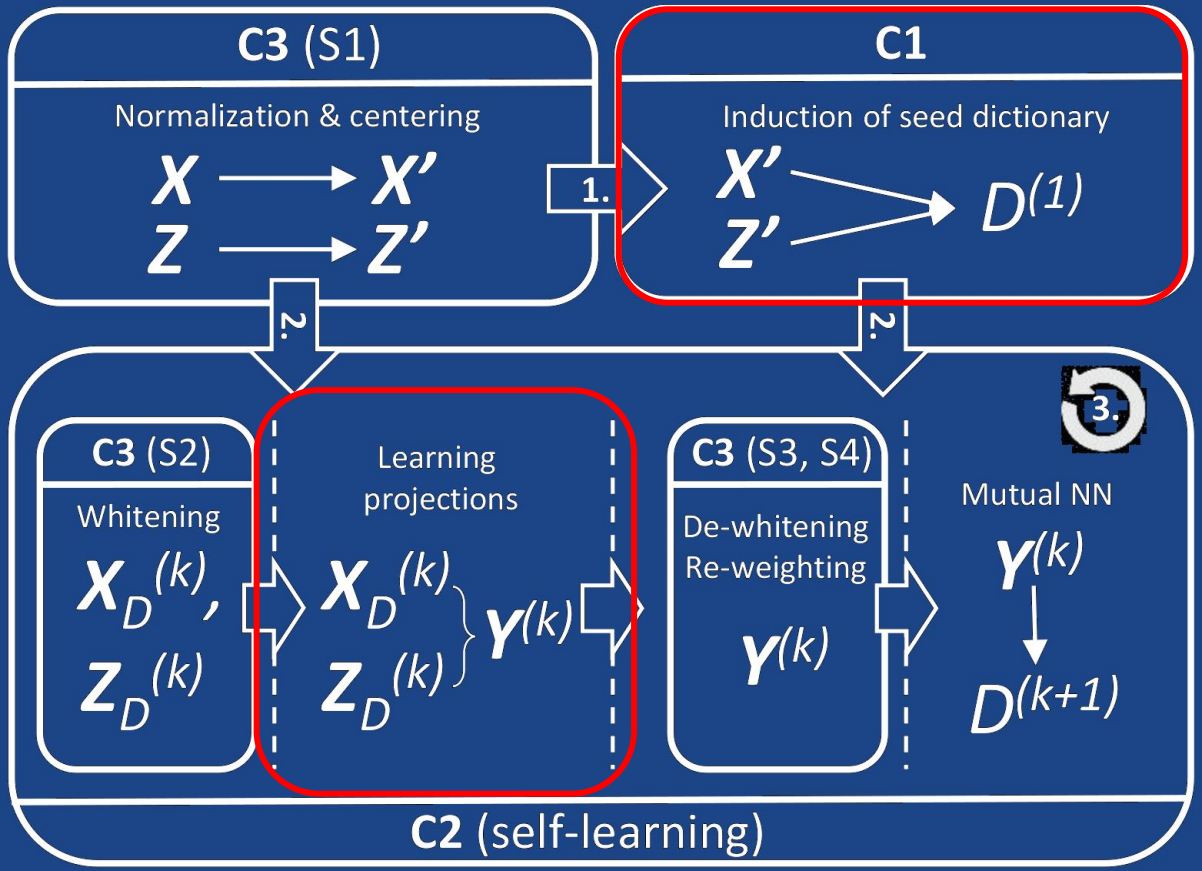
- SGD induces a strong inductive bias, provably learns a network close to the random initialization and with a *small generalization error (Li and Liang, 2018); also in the context of matrix factorization (Gunasekar et al., 2017), which can be used for cross-lingual learning (Zou et al., 2013). Drop-out and noise injection is extremely important for stable training of GANs (Chintala, 2016).
- *: SGD is biased toward wide valleys, which tend to generalize better (Chaudhari et al., 2017).

Systematic comparisons of unsupervised methods

A General Framework (with all the tricks…)



A General Framework (with all the tricks…)



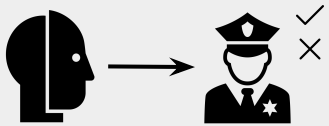
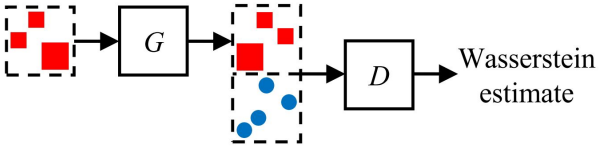
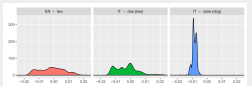
Typical focus: Seed dictionary induction and learning projections.

Common wisdom: Artetxe et al. (2018) state of the art, with incremental improvements in 2019.

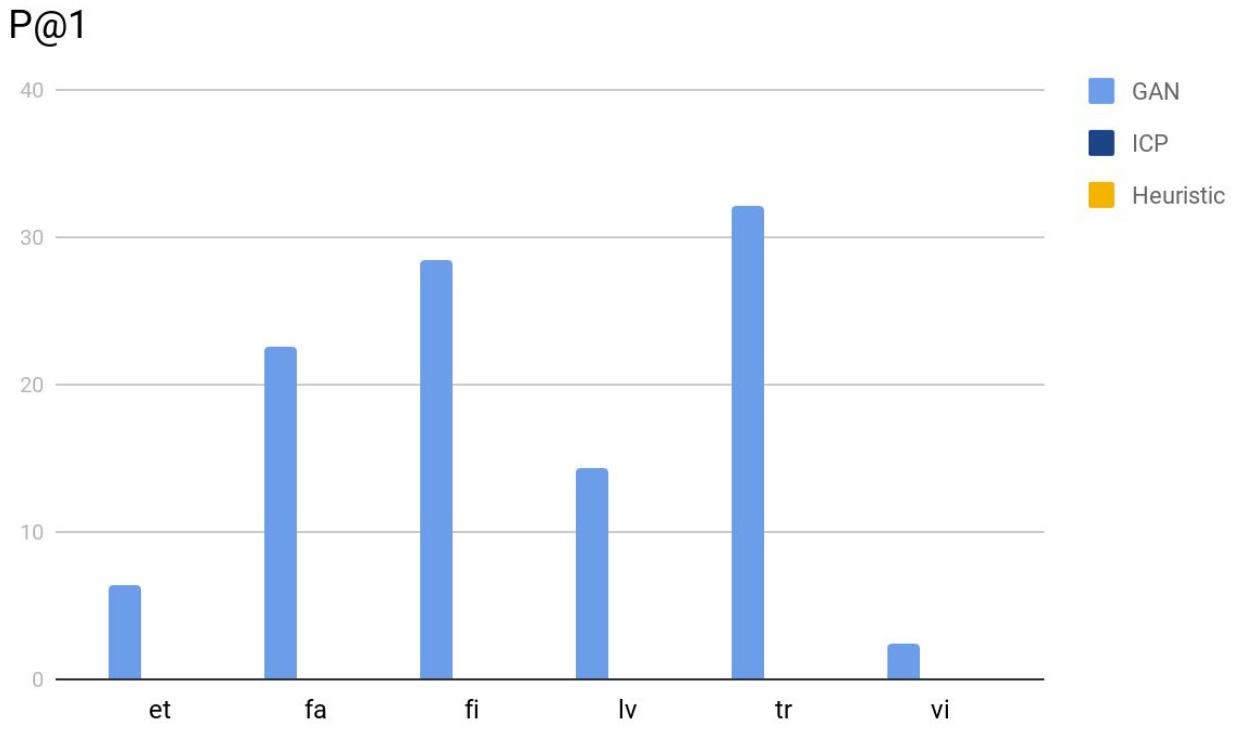
What we do: Fix learning projections and compare approaches to C1.

Comparing GAN, ICP and Heuristic

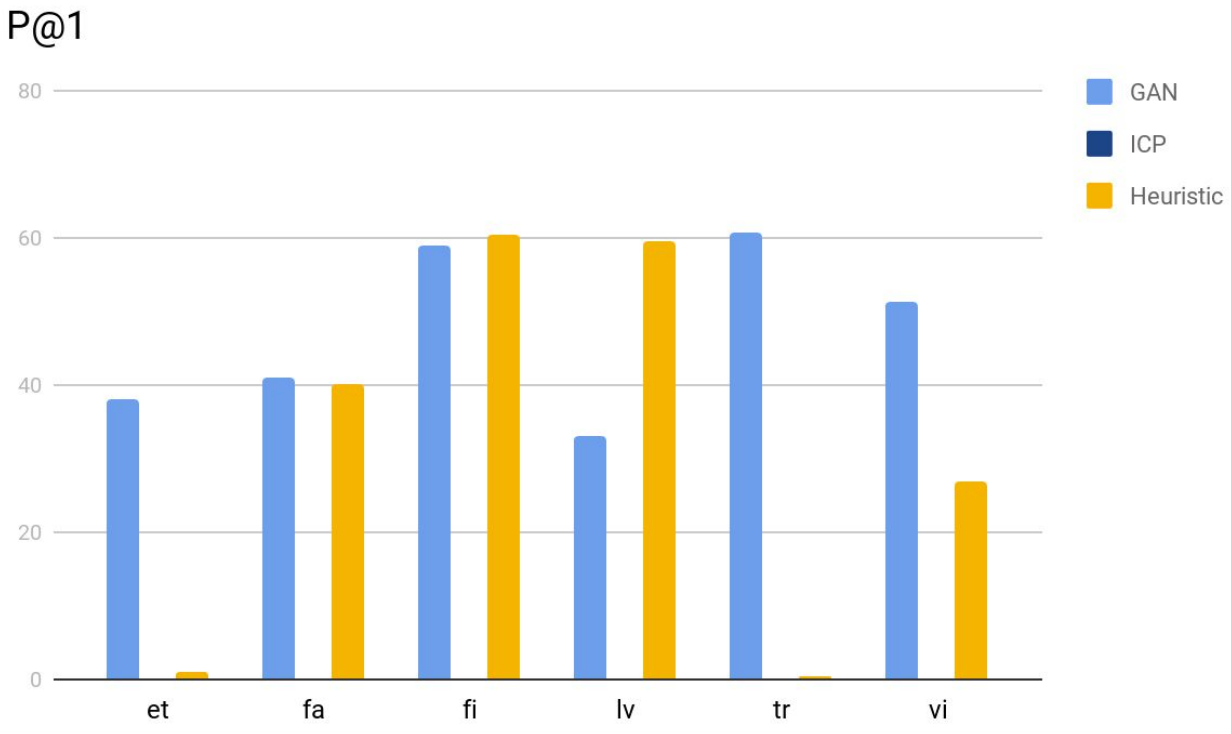
Overview

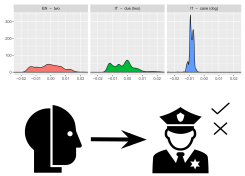
Authors	Seed dictionary induction
Barone (2016)	GAN 
Zhang et al. (2017a)	
Conneau et al. (2018)	
Zhang et al. (2017b)	Wasserstein GAN / Optimal transport 
Xu et al. (2018)	
Alvarez-Melis and Jaakkola (2018)	
Artetxe et al. (2018)	Heuristic 
Hoshen and Wolf (2018)	Point Cloud Matching 

GAN, ICP, and GWA (with no refinement)

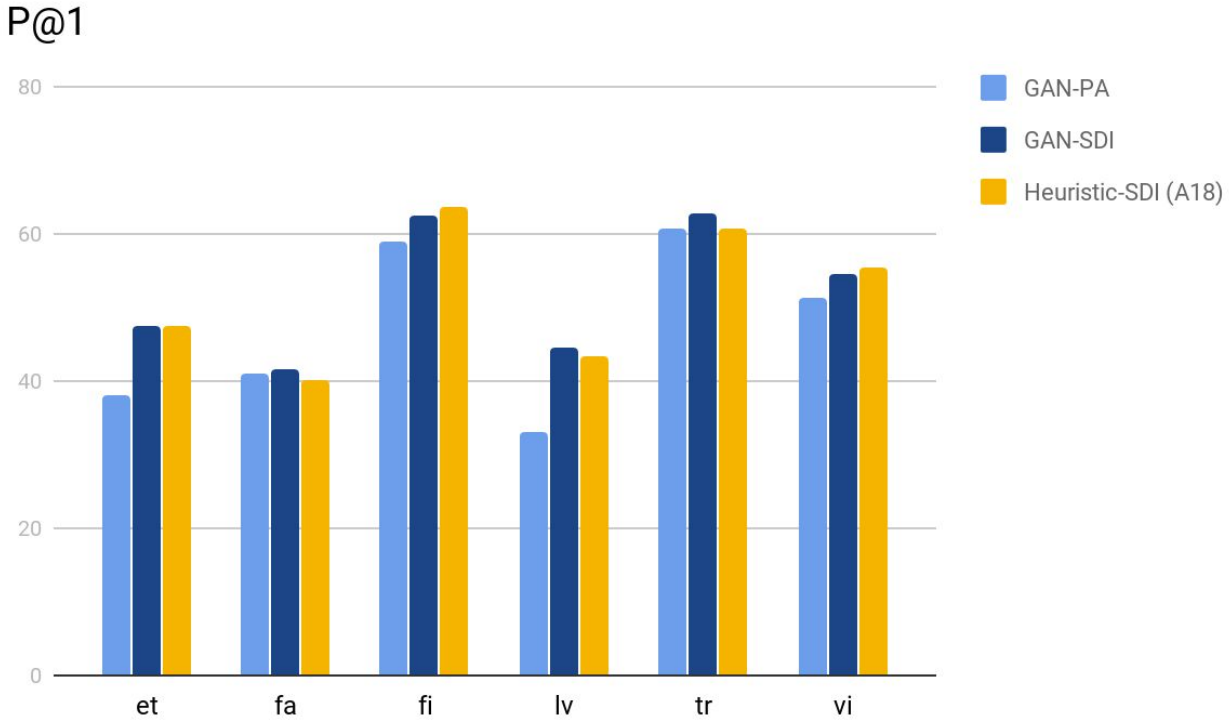


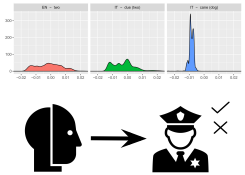
GAN, ICP, and GWA **with Procrustes Analysis**





GAN and GWA **with SDI**



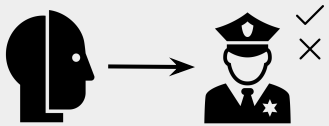
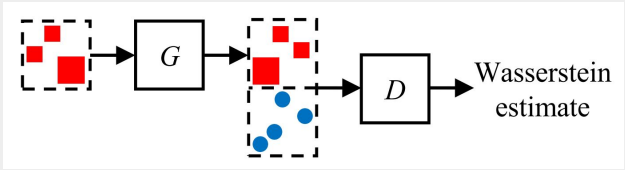
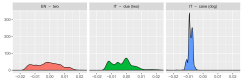


State of the art

Common wisdom	Unsupervised is better than supervised (unless you have several thousands of good seeds)?
	VecMap (second order initialization and stochastic dictionary induction) is better than MUSE (GANs and Procrustes Analysis)?
Our message	Stochastic dictionary induction improves GAN-based seed induction. How about other flavors of GANs?

Comparing flavors of GAN

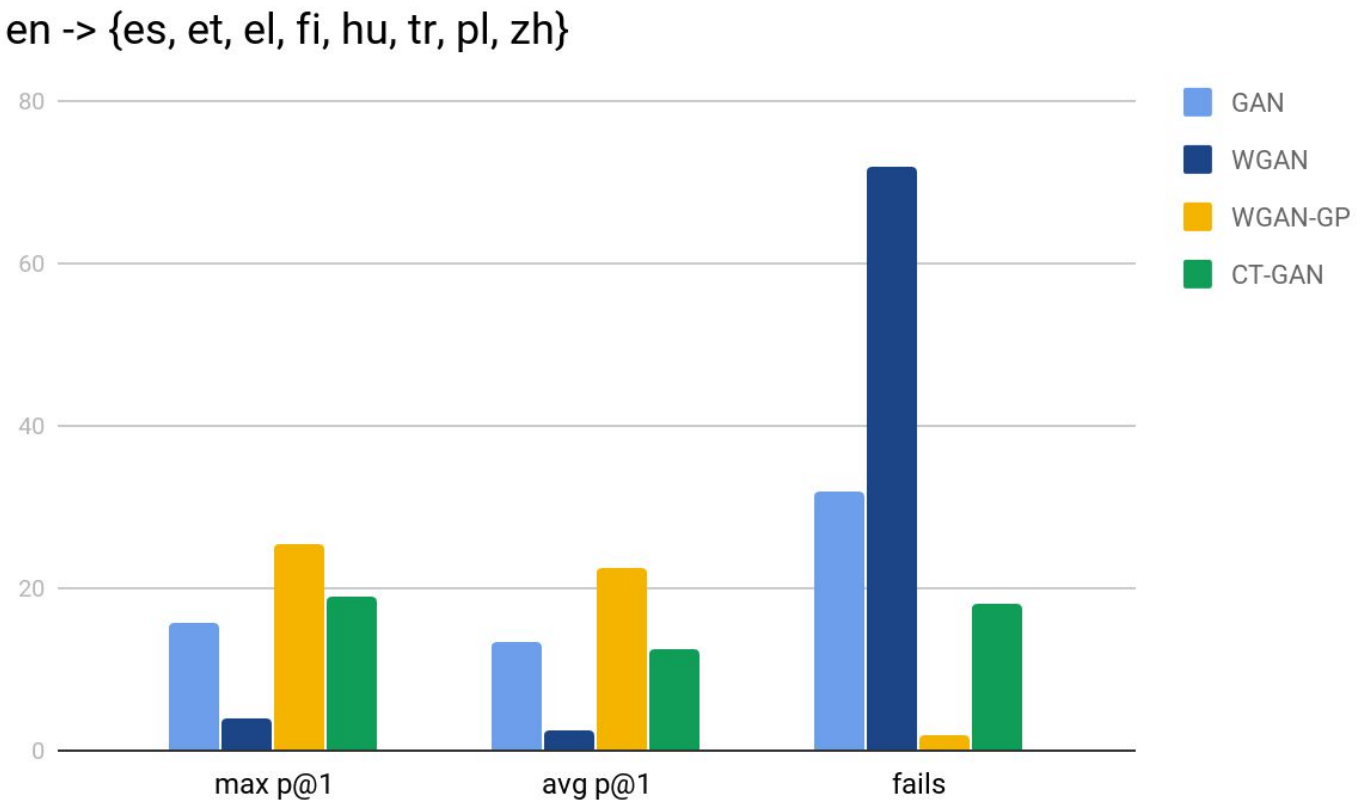
Overview

Authors	Seed dictionary induction
Barone (2016)	GAN 
Zhang et al. (2017a)	
Conneau et al. (2018)	
Zhang et al. (2017b)	Wasserstein GAN / Optimal transport 
Xu et al. (2018)	
Alvarez-Melis and Jaakkola (2018)	
Artetxe et al. (2018)	Heuristic 
Hoshen and Wolf (2018)	Point Cloud Matching 

Flavors of GANs

- *Ivan and Sebastian introduced vanilla GANs and WGANs.*
- **WGANs**: Motivated by mode collapse and hubs. Replaces discriminator by real-valued function. Uses gradient clipping.
- **WGAN-GPs** (Gulrajani et al., 2017): Provide gradient regularization (gradient penalty; GP) for more stable training (as an alternative to gradient clipping): norm of the discriminator gradients regularized to be 1 almost everywhere.
- **CT-GANs** (Wei et al., 2018): Use data augmentation (consistency regularization; CT) instead of regularization. Perturb each data point twice, bounding the difference between the discriminator responses to the two datapoints.

GAN, WGAN, WGAN-GP and CT-GAN (w/o refinement)



State of the art

Common wisdom	Unsupervised is better than supervised (unless you have several thousands of good seeds)?
	VecMap (second order initialization and stochastic dictionary induction) is better than MUSE (GANs and Procrustes Analysis)?
Our message	Stochastic dictionary induction improves GAN-based seed induction. WGAN-GP works best as GAN-based seed induction.

Systematic comparisons of unsupervised vs. supervised

Unsupervised vs. supervised

[Conneau et al.; ICLR 2018]: *"Without using any character information, our model even outperforms existing supervised methods on cross-lingual tasks for some language pairs"*

[Artetxe et al.; ACL 2018]: *"Our method succeeds in all tested scenarios and obtains the best published results in standard datasets, even surpassing previous supervised systems"*

[Hoshen and Wolf; EMNLP 2018]: *"...our method achieves better performance than recent state-of-the-art deep adversarial approaches and is competitive with the supervised baseline"*

[Xu et al.; EMNLP 2018]: *"Our evaluation (...) shows stronger or competitive performance of the proposed method compared to other state-of-the-art supervised and unsupervised methods..."*

[Chen and Cardie; EMNLP 2018]: *"In addition, our model even beats supervised approaches trained with cross-lingual resources."*

Unsupervised vs. supervised

- How come **unsupervised** is reportedly better than **supervised**?

Unsupervised vs. supervised

- How come **unsupervised** is reportedly better than **supervised**?
- *Argument 1:* Supervision is poor quality.

Unsupervised vs. supervised

- How come **unsupervised** is reportedly better than **supervised**?
- *Argument 1:* Supervision is poor quality. *Counter-argument:* We evaluate on the same data. *Possible counter-counter-argument:* Maybe the train splits are particularly poor?

Unsupervised vs. supervised

- How come **unsupervised** is reportedly better than **supervised**?
- *Argument 1:* Supervision is poor quality. *Counter-argument:* We evaluate on the same data. *Possible counter-counter-argument:* Maybe the train splits are particularly poor? *Argument 2:* Supervision is too limited.

Unsupervised vs. supervised

- How come **unsupervised** is reportedly better than **supervised**?
- *Argument 1:* Supervision is poor quality. *Counter-argument:* We evaluate on the same data. *Possible counter-counter-argument:* Maybe the train splits are particularly poor? *Argument 2:* Supervision is too limited.
- Note the standard motivation - that resources are lacking - often is **not** true: To see this, try to think of a language for which we can induce good embeddings, but for which we cannot collect 200 translations into a major language (say from PanLex or ASJP).

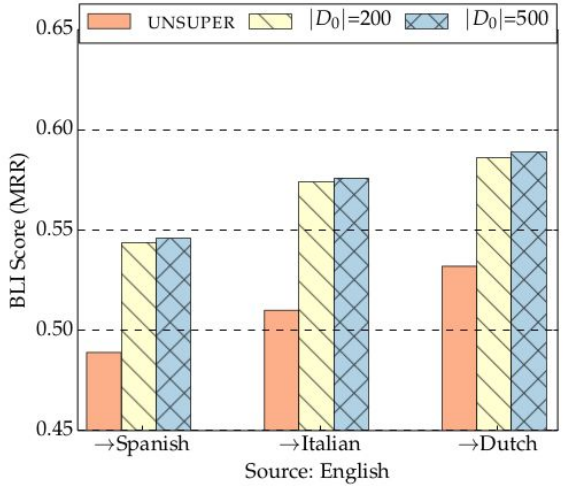
Unsupervised vs. supervised

{Bulgarian, Catalan, Esperanto, Estonian, Basque, Finnish, Hebrew, Hungarian, Indonesian, Georgian, Korean, Lithuanian, Bokmål, Thai, Turkish}

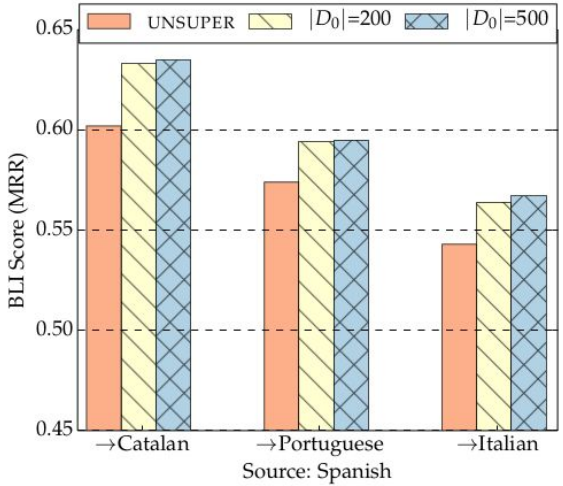
x

{Bulgarian, Catalan, Esperanto, Estonian, Basque, Finnish, Hebrew, Hungarian, Indonesian, Georgian, Korean, Lithuanian, Bokmål, Thai, Turkish}

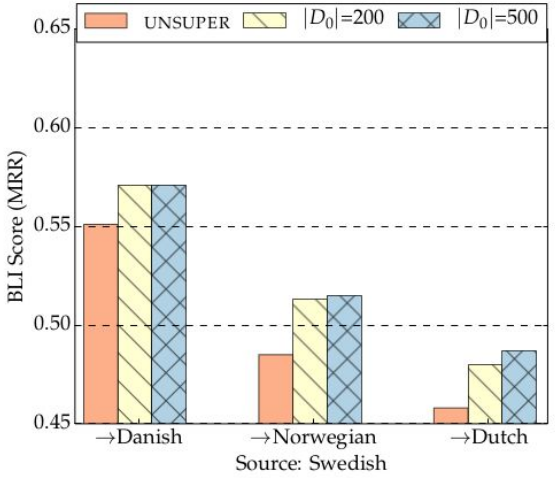
Unsupervised vs. supervised: similar languages



(a) English→ L_2



(b) Spanish→ L_2

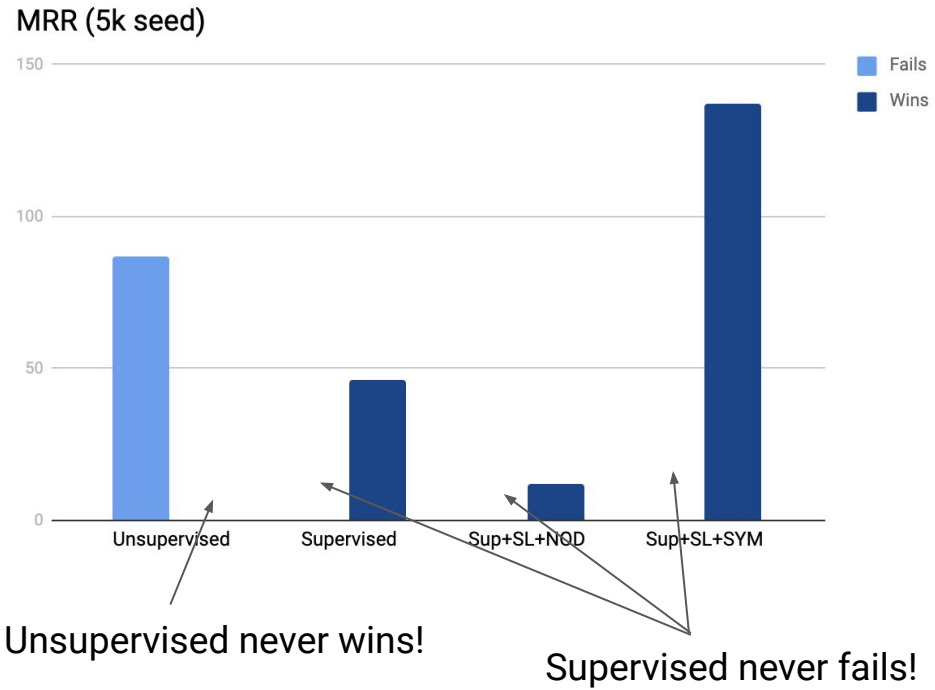


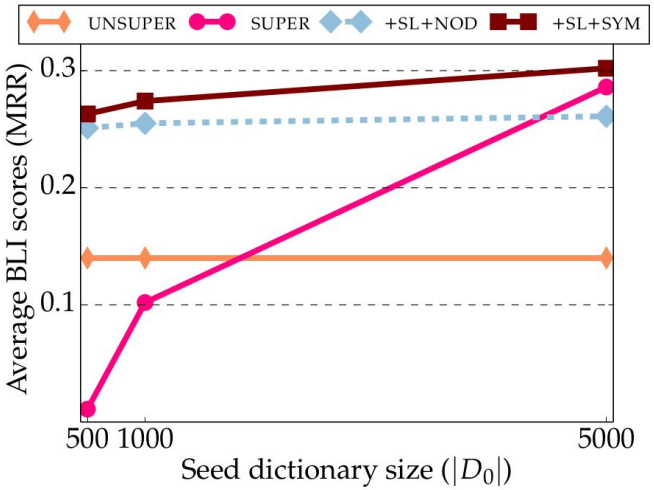
(c) Swedish→ L_2

While fully unsupervised CLWEs really show impressive performance for similar language pairs, they are still worse than weakly supervised methods…

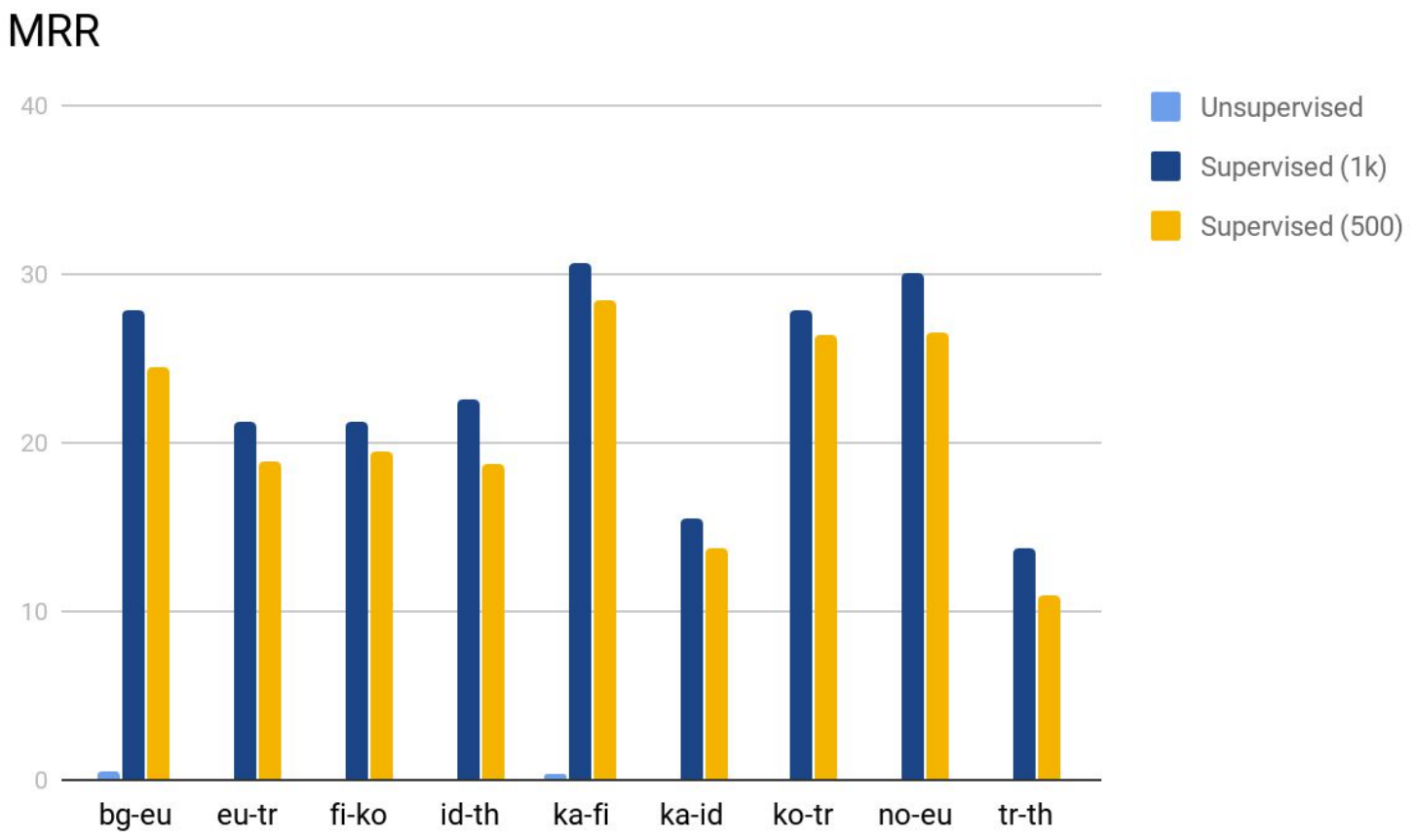
- Furthermore, we don't really need them for these scenarios...

Unsupervised vs. supervised: all languages





Unsupervised vs. supervised: distant languages

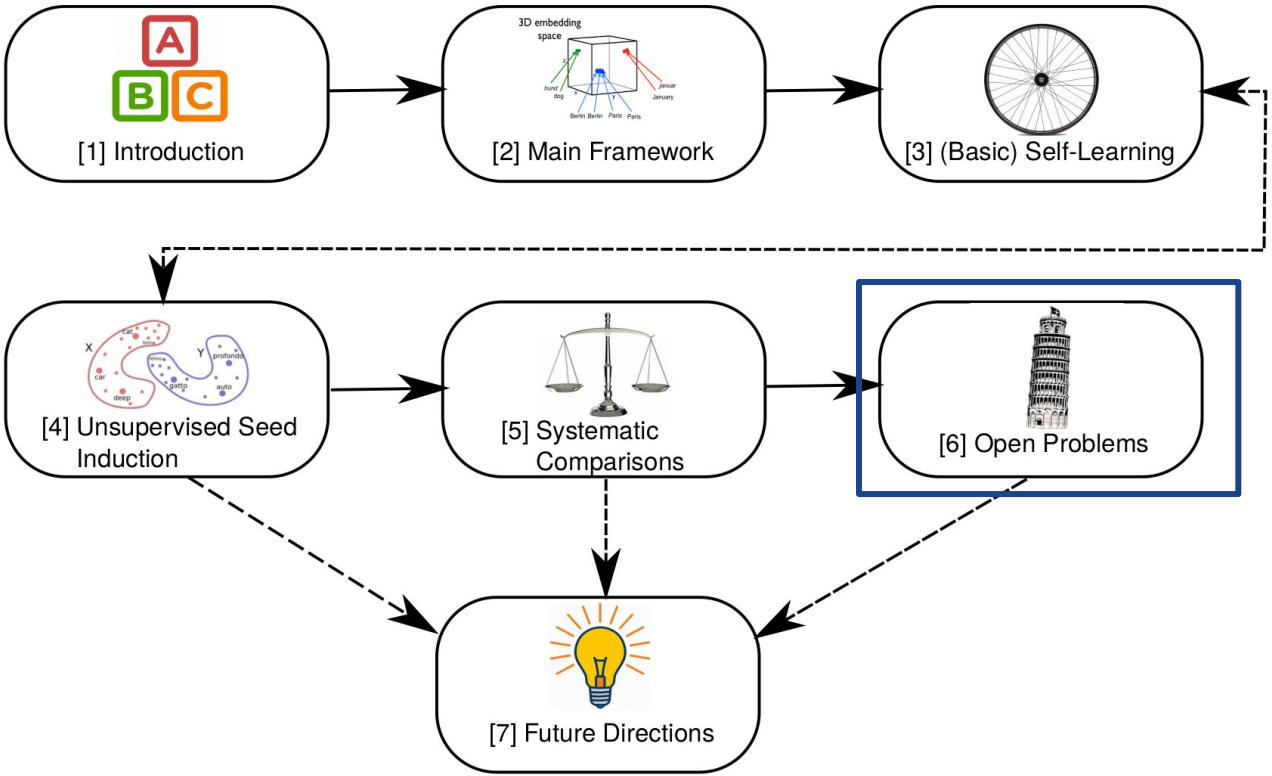


State of the art

Common wisdom	Unsupervised is better than supervised (unless you have several thousands of good seeds)?
	VecMap (second order initialization and stochastic dictionary induction) is better than MUSE (GANs and Procrustes Analysis)?
Our message	Stochastic dictionary induction improves GAN-based seed induction. WGAN-GP works best as GAN-based seed induction.
	Supervision doesn't seem to hurt.

6. Open Problems

6. Systematic Comparisons



Open problems

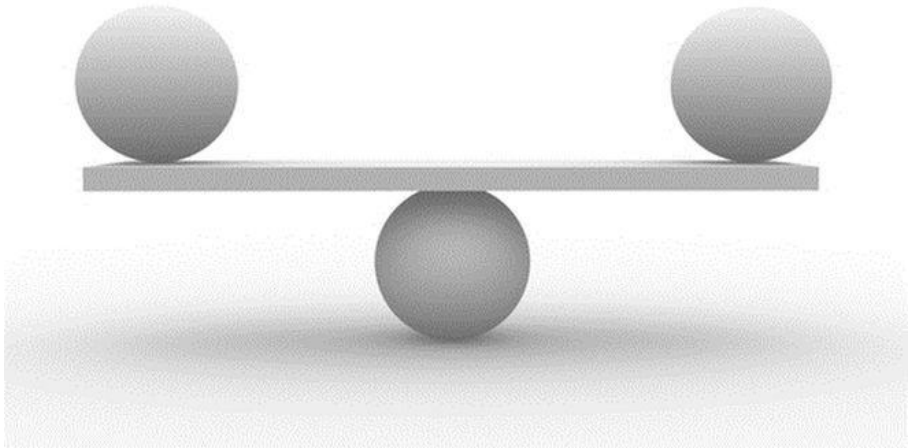
A. Robustness and Instability*
B. Morphology
C. Isomorphism
D. Evaluation

***We argue:** Maybe Instability is not a problem.

(Robustness and) Instability

Robustness and instability

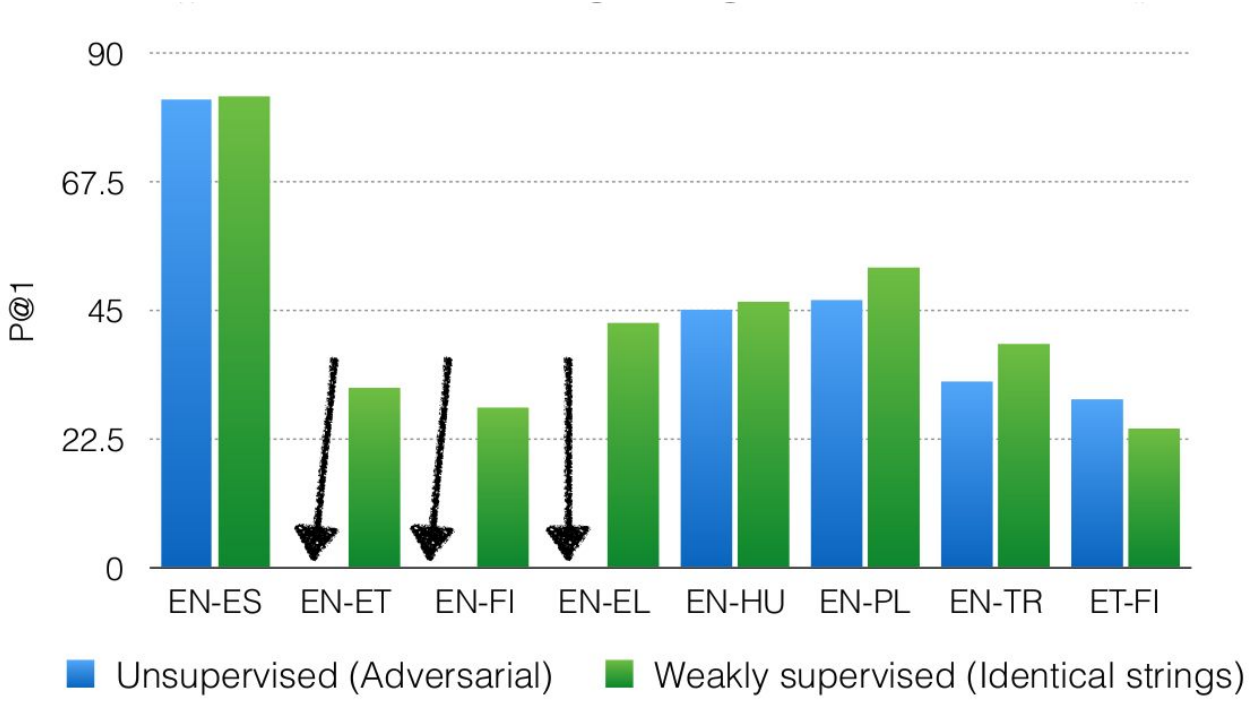
- Robustness is (lack of) sensitivity to language, domain, algorithm, etc.
- Instability is (lack of) sensitivity to random seed.



Robustness and instability

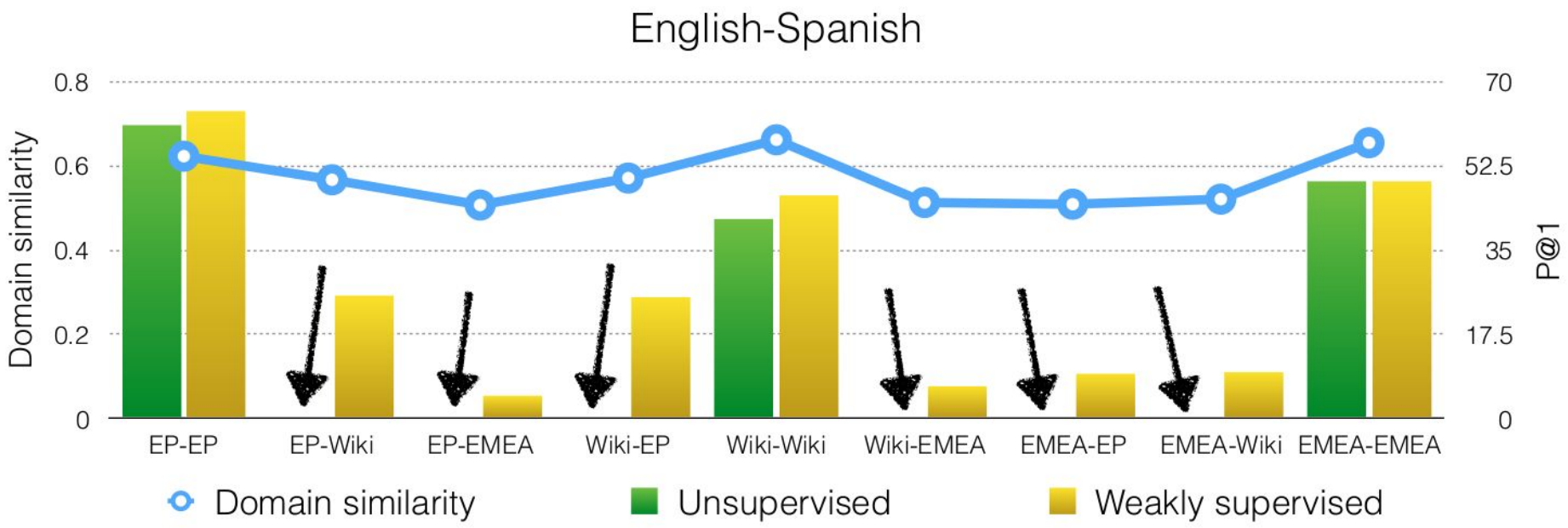
- MUSE was shown to lack robustness and stability in Søgaard et al. (2018), Hartmann et al. (2018), and Artetxe et al. (2018).
- ICP requires 600 random restarts to work reasonably.
- More robust and stable methods have been presented, but we've just seen that even VecMap is not stable on hard language pairs.
- Let's briefly review the ways in which MUSE lacks robustness and stability…

Sensitive to languages [Søgaard et al., ACL-18]



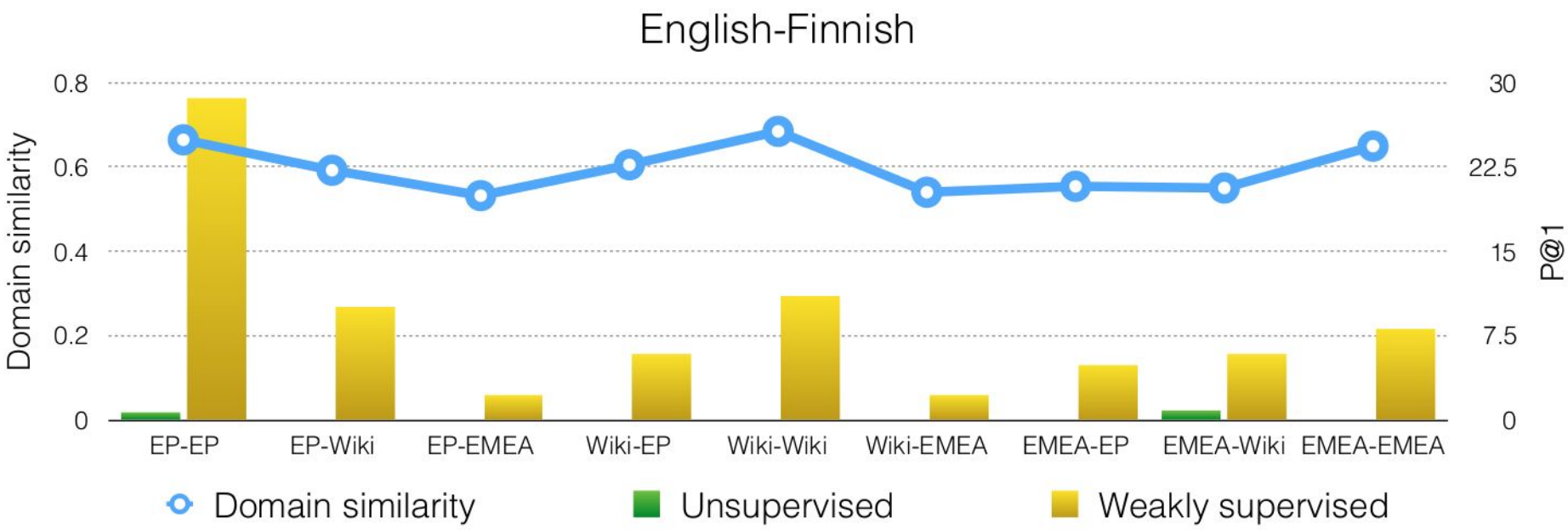
Adversarial training fails for more distant language pairs

Sensitive to domains [Søgaard et al., ACL-18]



Even the most robust unsupervised model breaks down when the domains are dissimilar, even for similar language pairs

Sensitive to both [Søgaard et al., ACL-18]



Domain differences may exacerbate difficulties of generalising across dissimilar languages

Sensitive to algorithm [Hartmann et al., EMNLP-18]

	Hyperwords-SGNS	Hyperwords-SVD	CBOW	GloVe	FastText
UNSUPERVISED					
Hyperwords-SGNS	**0.997**				
Hyperwords-SVD	0.000	**0.992**			
CBOW	0.000	0.000	**0.997**		
GloVe	0.000	0.000	0.000	**0.997**	
FastText	0.000	0.000	0.000	0.000	**0.997**
SUPERVISED					
Hyperwords-SVD	0.967				
CBOW	0.990	0.989			
GloVe	0.985	0.992	0.999		
FastText	0.994	0.994	0.999	0.997	

Sensitive to random seed [Søgaard et al.; Artetxe et al.]

- Søgaard et al. (2018): We note, though, that varying our random seed, performance for Estonian, Finnish, and Greek is sometimes (approximately 1 out of 10 runs) on par with Turkish. Detecting main causes and remedies for the inherent instability of adversarial training is one the most important avenues for future research.

- Artetxe et al. (2018): Given the instability of these methods, we perform 10 runs for each, and report the best and average accuracies, the number of successful runs (those with >5% accuracy)...

Is instability a problem?

- Instability is not a new phenomenon in NLP.
- EM-HMM training is highly unstable (Goldberg et al., 2008); MERT training is highly unstable (Moore & Quirk, 2008); Gibbs sampling for dependency parsing is highly unstable (Naseem and Barzilay, 2011), etc.
- Common solution: Use multiple random (or non-random) restarts and unsupervised selection criterion.
- High level view: This form of hill-climbing is not different from tricks in other learning algorithms (drop-out, exploration, etc.).

Morphology

Morphology

- Two basic assumptions: isomorphism and **1:1 correspondence**.
- Morphologically rich languages have a high average number of morphemes per word. This often breaks 1:1 correspondence.

Morphology

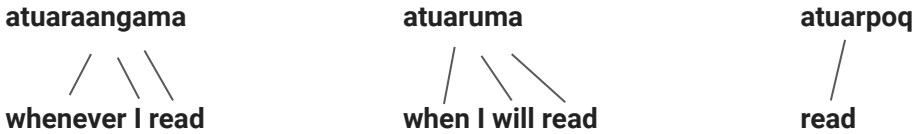
- Two basic assumptions: isomorphism and **1:1 correspondence**.
- Morphologically rich languages have a high average number of morphemes per word. This often breaks 1:1 correspondence.
- Instead it leads to 1:m correspondences and less support.

Morphology

- Two basic assumptions: isomorphism and **1:1 correspondence**.
- Morphologically rich languages have a high average number of morphemes per word. This often breaks 1:1 correspondence.
- Instead it leads to 1:m correspondences and less support.
- Lemmatization?

Morphology

- Two basic assumptions: isomorphism and **1:1 correspondence**.
- Morphologically rich languages have a high average number of morphemes per word. This often breaks 1:1 correspondence.
- Instead it leads to 1:m correspondences and less support.
- Lemmatization? Often just leads to m:n.
- Segmentation?



Morphology

- Morphology has been shown to lead to poor performance (Adams et al., 2017; Søgaard et al., 2018).
- … but so far, no flag's been planted.

Morphology

- Morphology has been shown to lead to poor performance (Adams et al., 2017; Søgaard et al., 2018).
- … but so far, no flag's been planted.
- Except maybe: Zhang et al. (2019) use subword information for cross-lingual document classification and show competitive performance on BDI for related languages.

Open questions

Does cross-lingual embeddings - and/or BDI - even make sense in the context of different morphologies?

Isomorphism

Isomorphism

- Philosophy class: :) Why would embedding spaces be isomorphic in the first place?
- Anna Wierzbicka's *Semantic primes*: I am also positing certain innate and universal rules of syntax – not in the sense of some intuitively unverifiable formal syntax à la Chomsky, but in the sense of intuitively verifiable patterns determining possible combinations of primitive concepts
- Universal structure of lexical semantics (Youn et al., 2016): Indeed, our results are consistent with the hypothesis that cultural and environmental factors have little statistically significant effect on the semantic network of the subset of basic concepts studied here. To a large extent, the semantic network appears to be a human universal: For instance, SEA/OCEAN and SALT are more closely related to each other than either is to SUN, and this pattern is true for both coastal and inland languages.

Measuring isomorphism

- More specifically, monolingual embedding spaces should be approximately isomorphic, i.e. same number of vertices, connected the same way
- Does not strictly hold even for related languages
- Can characterise similarity based on structure of nearest neighbour graphs.
- Eigenvector similarity:

Nearest neighbour (NN) graphs of ten most frequent nouns in English and their German translations

Laplacian eigenvalues

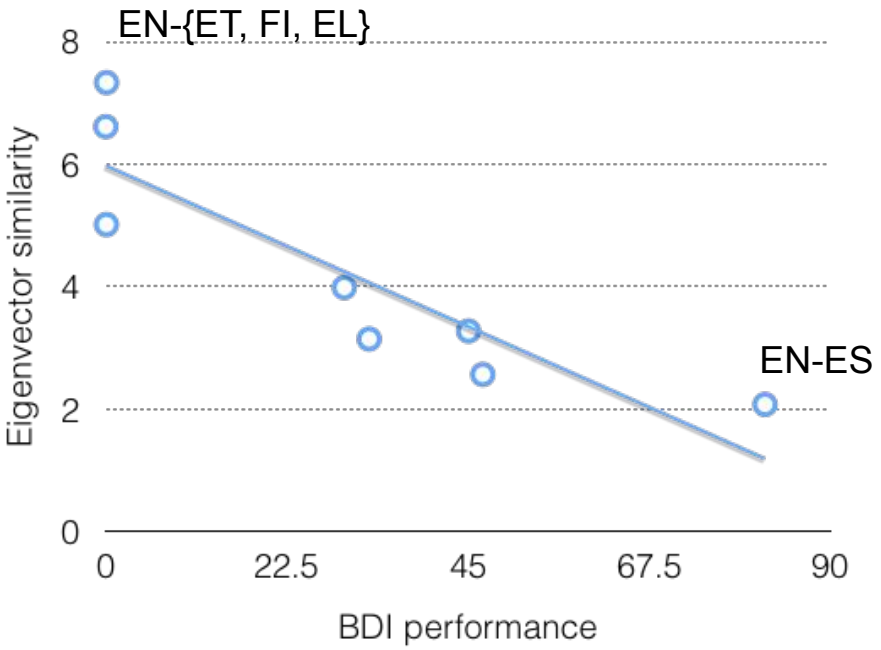
smallest k so that sum of largest k eigenvalues is > 90% of sum of all eigenvalues

$$\Delta = \sum_{i=1}^{k} (\lambda_{1_i} - \lambda_{2_i})^2 \text{ where } k = \min_j \{ \frac{\sum_{i=1}^{k} \lambda_{j_i}}{\sum_{i=1}^{n} \lambda_{j_i}} > 0.9 \}$$

Other measures of isomorphism: [Patra et al., ACL-19]

Measuring isomorphism

- Eigenvector similarity correlates strongly with bilingual dictionary induction performance ($\rho \sim 0.89$).



Søgaard et al. (2018)

Isomorphism

- Non-linear methods have been proposed by Nakashole (2018) and Zhang et al. (2019).

Isomorphism

- Non-linear methods have been proposed by Nakashole (2018) and Zhang et al. (2019).
- Nakashole (2018) combine several, independent linear maps to align two vector spaces.
- Zhang et al. (2019) use an iterative normalization (alternating projection) technique that enables alignment of non-isomorphic vector spaces.

Open questions

Does cross-lingual embeddings - and/or BDI - even make sense in the context of different morphologies?
Is non-isomorphism a problem we should deal with during alignment? Or a problem we should deal with when inducing monolingual embeddings?

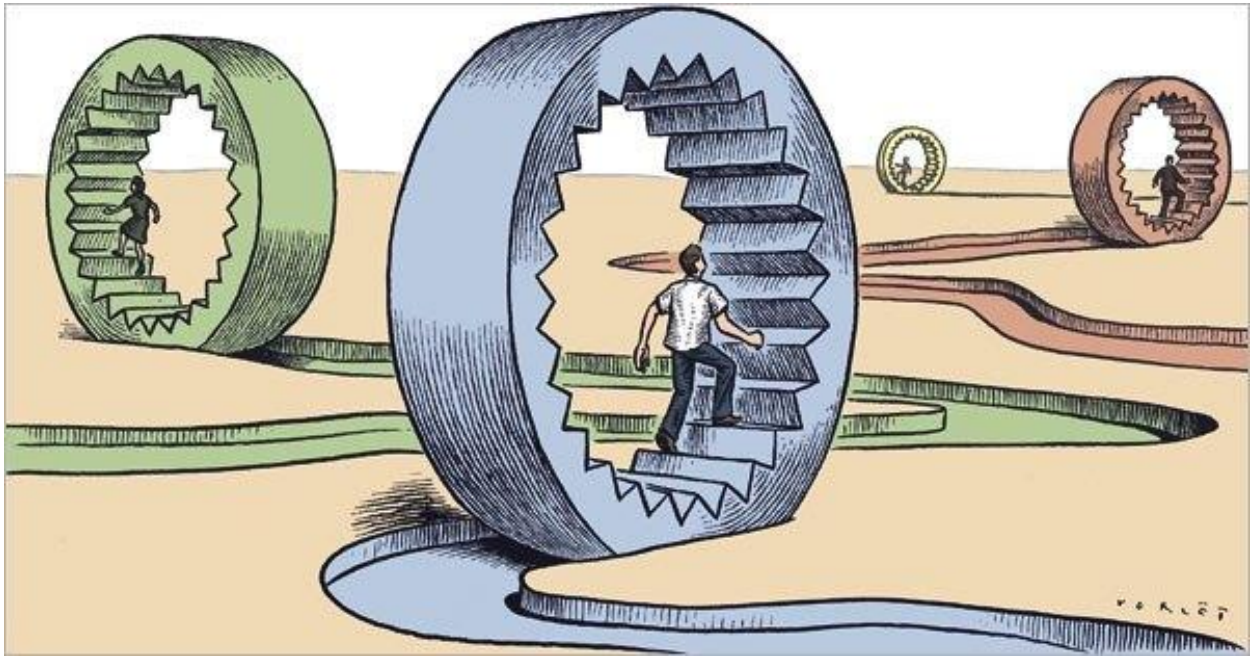
Evaluation

Open questions

Does cross-lingual embeddings - and/or BDI - even make sense in the context of different morphologies?
Is non-isomorphism a problem we should deal with in alignment? Or a problem we should deal with when inducing monolingual embeddings?
If the purpose of cross-lingual word embeddings is unsupervised MT, bilingual dictionary induction, and cross-lingual transfer, how should we best evaluate them?

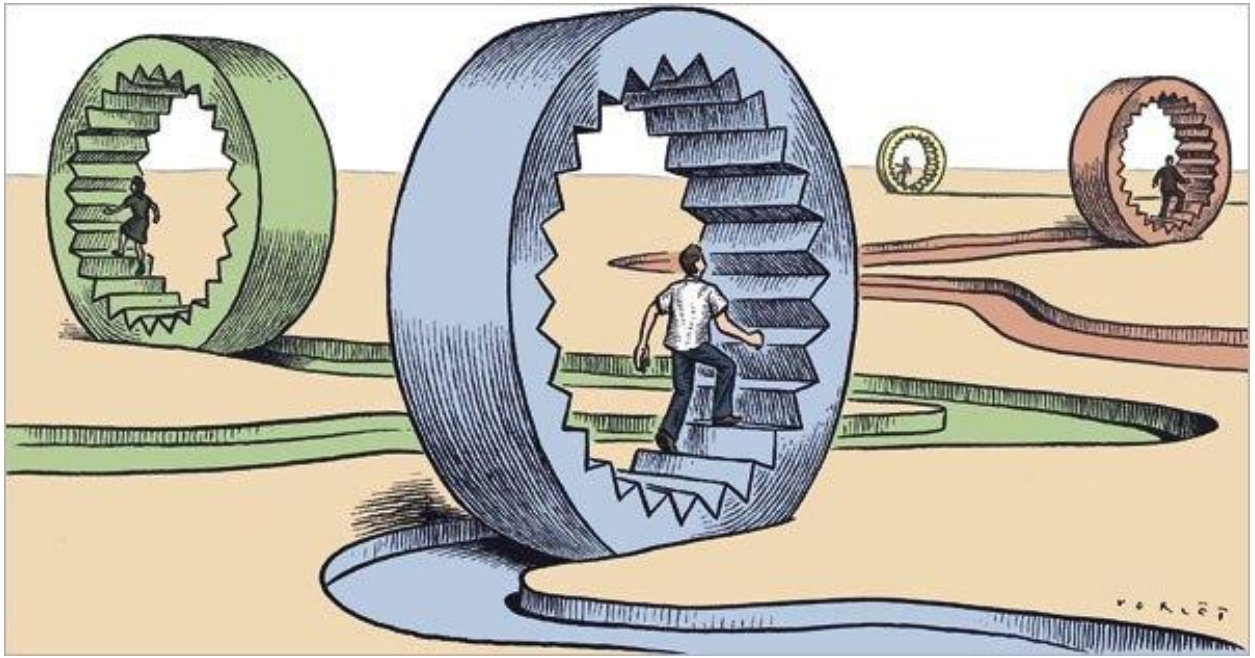
Standard approach

- Bilingual dictionary induction - using the MUSE dictionaries.



Standard approach

- Bilingual dictionary induction - using the MUSE dictionaries.



We argue:

Bad choice!

Bilingual dictionaries

MUSE	45
Wiktionary	400+
Panlex	9000+
ASJP	9000+

The problem with MUSE

- Dictionaries differ. MUSE is the *de facto* standard, but suffers from several weaknesses:
- Coverage gaps: Manual inspection shows many predictions by state of the art methods that are true, but forms/senses are absent from MUSE.
- Proper nouns (25% on average).
- Errors (1%).

Alternatives to Bilingual Lexicon Induction

Downstream tasks	
Document classification	Klementiev et al. (2003), etc.
Syntactic analysis	Gouws and Søgaard (2015), etc.
Semantic analysis	Glavas et al. (2019), etc.
Machine translation	Conneau et al. (2017), etc.
Word alignment	Levy et al. (2017), etc.

BLI correlation with downstream tasks

- Without RCSLS, BLI performance correlates almost perfectly with downstream performance for XNLI and CLIR, weakly for CLDC
- Why is RCSLS different?
- RCSLS relaxes orthogonality constraint on projection matrix

→ For non-orthogonal projections, downstream evaluation is particularly important

[Glavaš et al., ACL-19]

Models	XNLI	CLDC	CLIR
All models	0.269	**0.390**	0.764
All w/o RCSLS	**0.951**	0.266	**0.910**

Correlations of model-level results between BLI and each of the downstream tasks.

The problem (or beauty) of downstream evaluation

- Results are likely to be all over the map (Elming et al., 2013).

  ⇒ Statistical testing over tasks is unlikely to lead to significance.

  ⇒ Raises the bar for publication.

Metrics for BDI

- P@1
- P@10
- MRR
- Spearman's

$$MRR = \frac{1}{N} \sum_{i=1}^{N} \frac{1}{rank_i}$$

Other methodological weaknesses

- Unclear what task to evaluate on.
- Often unclear what metric to use to measure performance.
- Community-wide overfitting to a small set of Indo-European languages.

  BUT: More and more people are working on more distant and low-resource language pairs.

Other methodological weaknesses

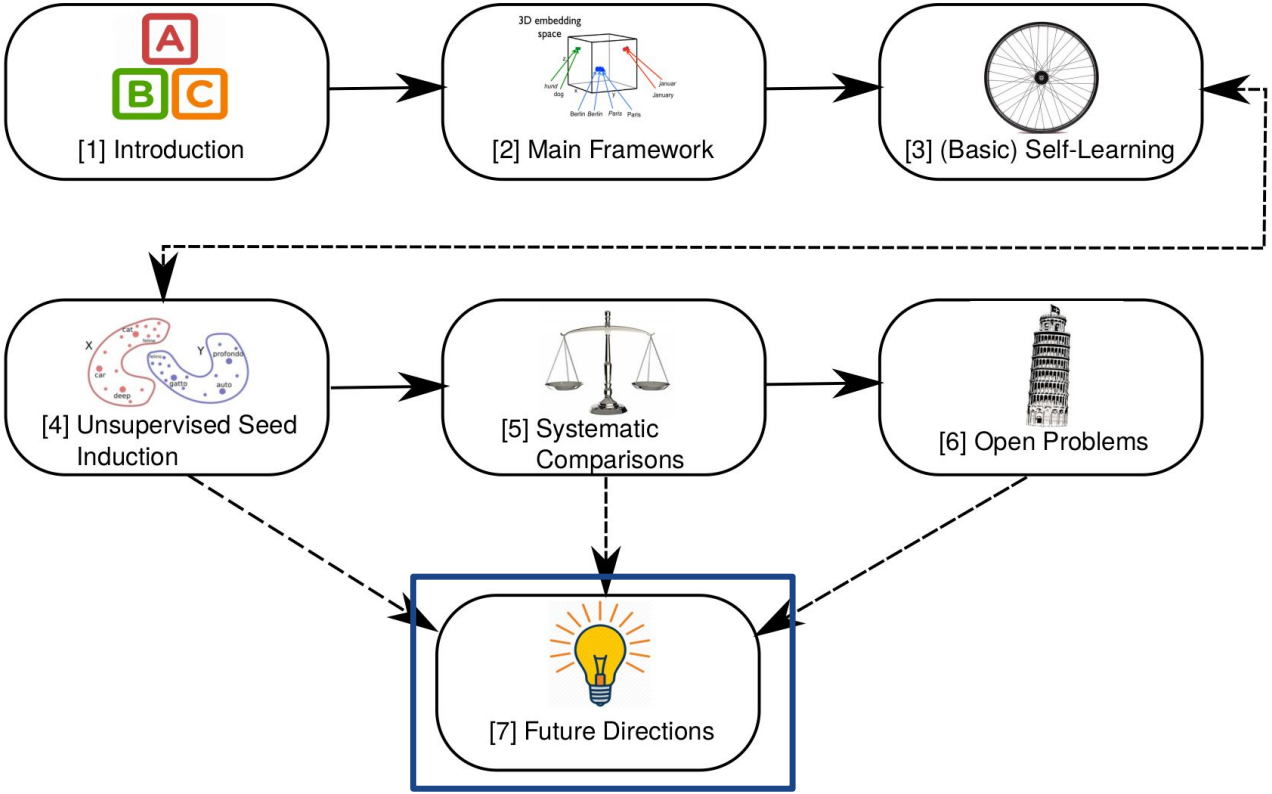
- Unclear what task to evaluate on.
- Often unclear what metric to use to measure performance.
- Community-wide overfitting to a small set of Indo-European languages.
- BUT: More and more people are working on more distant and low-resource language pairs.
- No agreement how to set hyper-parameters in experiments; in practice, making it an advantage to relegate parameters to hyper-parameters.

Other methodological weaknesses

- Unclear what task to evaluate on.
- Often unclear what metric to use to measure performance.
- Community-wide overfitting to a small set of Indo-European languages.
- BUT: More and more people are working on more distant and low-resource language pairs.
- No agreement how to set hyper-parameters in experiments; in practice, making it an advantage to relegate parameters to hyper-parameters.
- Not clear how to best think about and quantify instability.

Conclusions

7. Conclusions and Future Directions



Conclusions: Uncertainty prevails

- Comparisons have **not** been systematic so far.
- It seemed that VecMap was superior, even to supervised approaches; instead, while stochastic dictionary induction (and symmetry) is very useful with bad seeds, (W)GAN(-GP) is a competitive initialization strategy
- Supervision does not hurt and is better if you have >1k alignments.
- Such a ranking of methods, of course, assumes a set of languages, a task, a dataset, and a metric.
- Fixing that, our benchmarks are still of poor quality. In sum, we need **better benchmarks**, a way of dealing with morphology, and a philosophy of isomorphism.

Open questions

Does cross-lingual embeddings - and/or BDI - even make sense in the context of different morphologies?
Is non-isomorphism a problem we should deal with in alignment? Or a problem we should deal with when inducing monolingual embeddings?
If the purpose of cross-lingual word embeddings is unsupervised MT, bilingual dictionary induction, and cross-lingual transfer, how should we best evaluate them?
How strong is the correlation between extrinsic and intrinsic evaluation tasks?
Denoising seed lexica? (Beyond dropout and symmetry?)
Cross-lingual contextualized embeddings [Schuster et al., NAACL-19; Aldarmaki and Diab, NAACL-19]

Open questions

Does cross-lingual embeddings - and/or BDI - even make sense in the context of different morphologies?
Is non-isomorphism a problem we should deal with in alignment? Or a problem we should deal with when inducing monolingual embeddings?
If the purpose of cross-lingual word embeddings is unsupervised MT, bilingual dictionary induction, and cross-lingual transfer, how should we best evaluate them?
How strong is the correlation between extrinsic and intrinsic evaluation tasks?
Denoising seed lexica? (Beyond dropout and symmetry?)
Cross-lingual contextualized embeddings, e.g., [Aldarmaki and Diab, NAACL-19]

Are there scenarios in cross-lingual NLP where we can **really really benefit from fully unsupervised** cross-lingual word embeddings?

Useful Links and Resources

Useful Links

- Multilingual fastText vectors in 78 languages:
  https://github.com/Babylonpartners/fastText_multilingual

- More multilingual fastText vectors (44 languages, aligned with RCSLS):
  https://fasttext.cc/docs/en/aligned-vectors.html

- Cross-lingual language modeling pretraining:
  https://github.com/facebookresearch/XLM

- Multilingual BERT:
  https://github.com/google-research/bert/blob/master/multilingual.md

- Unsupervised MT (SMT and NMT):
  https://github.com/facebookresearch/UnsupervisedMT
  https://github.com/artetxem/monoses
  https://github.com/artetxem/undreamt

Useful Links

- VecMap…
  https://github.com/artetxem/vecmap
- ...and its latent variable variant:
  https://github.com/sebastianruder/latent-variable-vecmap
- GAN-based seed induction (MUSE):
  https://github.com/facebookresearch/MUSE
- Gromov-Wasserstein seed induction:
  https://github.com/dmelis/otalign
- Seed induction based on point cloud matching (non-adversarial):
  https://github.com/facebookresearch/NAM

Useful Links

- GANs + Sinkhorn:
  https://github.com/xrc10/unsup-cross-lingual-embedding-transfer
- Unsupervised cross-lingual IR:
  https://github.com/rlitschk/UnsupCLIR
- BLI - classification framework:
  https://github.com/geert-heyman/BLI_classifier
- (Cross-lingual) dialogue state tracking:
  https://github.com/nmrksic/neural-belief-tracker
  https://github.com/salesforce/glad
  https://github.com/wenhuchen/Cross-Lingual-NBT

...And Some Resources (!)

- https://panlex.org/
- https://babelnet.org
- https://asjp.clld.org/ (The ASJP Database)

- 135M parallel sentences in 1,620 language pairs:
  https://github.com/facebookresearch/LASER/tree/master/tasks/WikiMatrix

- Coming soon: JW300 (presented at ACL-19):
  At least 50k-100k parallel sentences for **54,376 language pairs**

- Plus OPUS and other similar repos...

Questions?

Slides: https://tinyurl.com/xlingual

If you found these slides helpful, consider citing the tutorial as:

```
@inproceedings{ruder2019unsupervised,
  title={Unsupervised Cross-Lingual Representation Learning},
  author={Ruder, Sebastian and S{\o}gaard, Anders and Vuli{\'c}, Ivan},
  booktitle={Proceedings of ACL 2019, Tutorial Abstracts},
  pages={31--38},
  year={2019}
}
```

Some Important Aspects **Not Covered Here**

- Other evaluation tasks beyond BLI
  [Upadhyay et al., ACL-16; Heyman et al., NAACL-19; Glavaš et al., ACL-19]
- Different ways to use cross-lingual word embeddings in a variety of applications
  [Guo et al., ACL-15; Mrkšić et al., TACL-17; Vulić et al., EMNLP-17, Upadhyay et al., NAACL-18]
- Quantifying approximate isomorphism and correlating it to CLWE task performance
  [Søgaard et al., ACL-18; Hartmann et al., arXiv-18; Alvarez Melis and Jaakkola, EMNLP-18, Patra et al., ACL-19; Fujinuma et al., ACL-19]
- Different methods to reduce anisomorphism, or to improve model selection, or to refine the initial mapping
  [Doval et al., EMNLP-18, Zhang et al., ACL-19, Patra et al., ACL-19]
- Other probing tests: hyper-parameter variation, different algorithms, word frequency
  [Søgaard et al., ACL-18; Hartmann et al., EMNLP-18; Braune et al., NAACL-18]
- Other similar approaches: Sinkhorn distances, multilingual learning
  [Grave et al., ICLR-18; Wu et al., EMNLP-18; Heyman et al., NAACL-19, Alaux et al., ICLR-19]