Lexical Semantic Change: Models, Data and Evaluation

LREC 2022 - Tutorial - 20 June 2022

Pierpaolo Basile[1], Annalina Caputo[2], Pierluigi Cassotti[1] and Rossella Varvara[3]
University of Bari[1], Dublin City University[2], Université de Fribourg[3]

Outline

- **Background concepts**
    - PPMI matrix factorization
    - Word2Vec Skip-gram with Negative Sampling (SGNS)
    - BERT-based models
- **Lexical Semantic Change Models**
    - Alignment models
        - Post-alignment models
            - Orthogonal Procrustes
        - Jointly Alignment Models
            - Explicit Alignment Models
                - Dynamic Word Embedding (DWE)
                - Dynamic Bernoulli Embedding (DBE)
            - Implicit Alignment Models
                - Temporal Referencing (TR)
                - Temporal Random Indexing (TRI)
                - Temporal Word Embedding with a Compass (TWEC)
    - Contextualized Models
        - TempoBERT
        - Temporal Attention
        - Deep Mistake
        - Gloss Reader
    - Other Models
        - Local Neighborhood measure
        - Word Sense Induction
        - Grammatical Features

PPMI Factorization

Mutual Information

SYMMETRIC
NON NEGATIVE

$$I(X,Y) = \sum_{x} \sum_{y} P(x,y) \log_2 \frac{P(x,y)}{P(x)P(y)}$$

Mutual information (MI) of two random variables is a measure of the mutual dependence between the two variables.

It quantifies the "amount of information" obtained about one random variable by observing the other random variable

Pointwise Mutual Information

$$\mathrm{PMI}(w,c) = \log_2 \frac{P(w,c)}{P(w)P(c)}$$

PMI(w,c) = 0	w and c are statistically independent
PMI(w,c)>0	w and c co-occur more frequently than would be expected under an independence assumption
PMI(w,c)<0	w and c co-occur less frequently than would be expected

Positive Pointwise Mutual Information

	computer	data	result	pie	sugar	count(w)
cherry	2	8	9	442	25	486
strawberry	0	0	1	60	19	80
digital	1670	1683	85	5	4	3447
information	3325	3982	378	5	13	7703
count(context)	4997	5673	473	512	61	11716

$$\text{PPMI}(w,c) = \max\left(\log_2 \frac{P(w,c)}{P(w)P(c)}, 0\right)$$

Positive Pointwise Mutual Information

	computer	data	result	pie	sugar	count(w)
cherry	2	8	9	442	25	486
strawberry	0	0	1	60	19	80
digital	1670	1683	85	5	4	3447
information	3325	3982	378	5	13	7703
count(context)	4997	5673	473	512	61	11716

$$p_{ij} = \frac{f_{ij}}{\sum_{i=1}^{W}\sum_{j=1}^{C} f_{ij}}, \quad p_{i*} = \frac{\sum_{j=1}^{C} f_{ij}}{\sum_{i=1}^{W}\sum_{j=1}^{C} f_{ij}}, \quad p_{*j} = \frac{\sum_{i=1}^{W} f_{ij}}{\sum_{i=1}^{W}\sum_{j=1}^{C} f_{ij}}$$

$$\text{PPMI}_{ij} = \max(\log_2 \frac{p_{ij}}{p_{i*}p_{*j}}, 0)$$

Jurafsky, Dan. *Speech & language processing*. Pearson Education India, 2000.

Positive Pointwise Mutual Information

$PPMI_{digital,data}$

	computer	data	result	pie	sugar	count(w)
cherry	2	8	9	442	25	486
strawberry	0	0	1	60	19	80
digital	1670	$\boxed{1683}$	85	5	4	3447
information	3325	3982	378	5	13	7703
count(context)	4997	5673	473	512	61	11716

$$p_{ij} = \frac{\boxed{f_{ij}}}{\sum_{i=1}^{W}\sum_{j=1}^{C} f_{ij}}, \quad p_{i*} = \frac{\sum_{j=1}^{C} f_{ij}}{\sum_{i=1}^{W}\sum_{j=1}^{C} f_{ij}}, \quad p_{*j} = \frac{\sum_{i=1}^{W} f_{ij}}{\sum_{i=1}^{W}\sum_{j=1}^{C} f_{ij}}$$

$$\text{PPMI}_{ij} = \max(\log_2 \frac{p_{ij}}{p_{i*}p_{*j}}, 0)$$

Positive Pointwise Mutual Information

$\text{PPMI}_{\text{digital,data}}$

	computer	data	result	pie	sugar	count(w)
cherry	2	8	9	442	25	486
strawberry	0	0	1	60	19	80
digital	1670	1683	85	5	4	3447
information	3325	3982	378	5	13	7703
count(context)	4997	5673	473	512	61	11716

$$p_{ij} = \frac{f_{ij}}{\boxed{\sum_{i=1}^{W}\sum_{j=1}^{C} f_{ij}}}, \quad p_{i*} = \frac{\sum_{j=1}^{C} f_{ij}}{\boxed{\sum_{i=1}^{W}\sum_{j=1}^{C} f_{ij}}}, \quad p_{*j} = \frac{\sum_{i=1}^{W} f_{ij}}{\boxed{\sum_{i=1}^{W}\sum_{j=1}^{C} f_{ij}}}$$

$$\text{PPMI}_{ij} = \max(\log_2 \frac{p_{ij}}{p_{i*}p_{*j}}, 0)$$

Positive Pointwise Mutual Information

	computer	data	result	pie	sugar	count(w)
cherry	2	8	9	442	25	486
strawberry	0	0	1	60	19	80
digital	1670	1683	85	5	4	3447
information	3325	3982	378	5	13	7703
count(context)	4997	5673	473	512	61	11716

$\text{PPMI}_{\text{digital,data}}$

$$p_{ij} = \frac{f_{ij}}{\sum_{i=1}^{W}\sum_{j=1}^{C} f_{ij}}, \quad p_{i*} = \frac{\boxed{\sum_{j=1}^{C} f_{ij}}}{\sum_{i=1}^{W}\sum_{j=1}^{C} f_{ij}}, \quad p_{*j} = \frac{\sum_{i=1}^{W} f_{ij}}{\sum_{i=1}^{W}\sum_{j=1}^{C} f_{ij}}$$

$$\text{PPMI}_{ij} = \max(\log_2 \frac{p_{ij}}{p_{i*}p_{*j}}, 0)$$

Positive Pointwise Mutual Information

PPMI$_{\text{digital,data}}$

	computer	data	result	pie	sugar	count(w)
cherry	2	8	9	442	25	486
strawberry	0	0	1	60	19	80
digital	1670	1683	85	5	4	3447
information	3325	3982	378	5	13	7703
count(context)	4997	5673	473	512	61	11716

$$p_{ij} = \frac{f_{ij}}{\sum_{i=1}^{W}\sum_{j=1}^{C} f_{ij}}, \quad p_{i*} = \frac{\sum_{j=1}^{C} f_{ij}}{\sum_{i=1}^{W}\sum_{j=1}^{C} f_{ij}}, \quad p_{*j} = \frac{\boxed{\sum_{i=1}^{W} f_{ij}}}{\sum_{i=1}^{W}\sum_{j=1}^{C} f_{ij}}$$

$$\text{PPMI}_{ij} = \max(\log_2 \frac{p_{ij}}{p_{i*}p_{*j}}, 0)$$

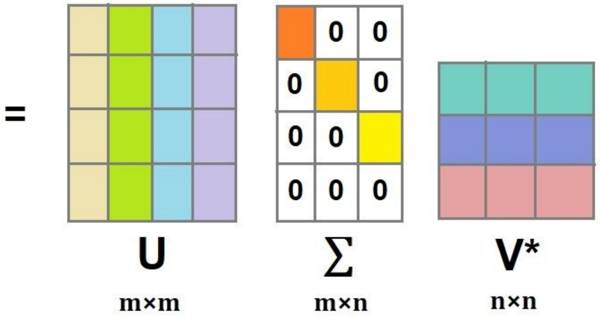
Positive Pointwise Mutual Information

	computer	data	result	pie	sugar
cherry	0	0	0	4.38	3.30
strawberry	0	0	0	4.10	5.51
digital	0.18	0.01	0	0	0
information	0.02	0.09	0.28	0	0

PPMI Factorization

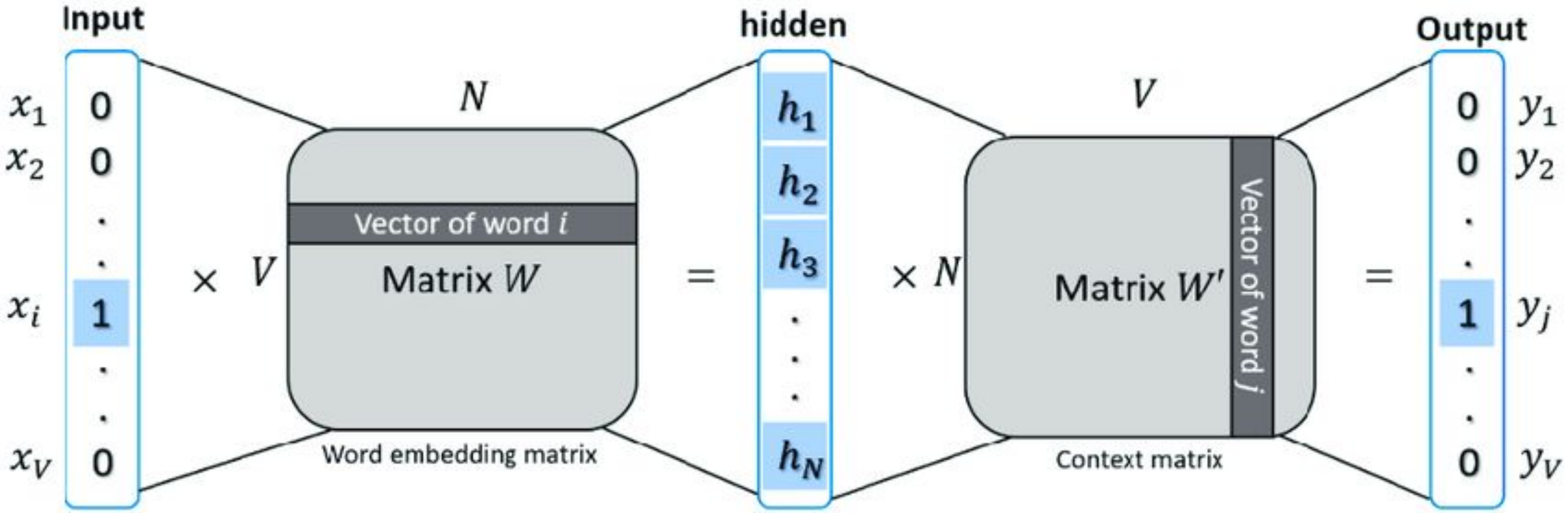
	computer	data	result	pie	sugar
cherry	0	0	0	4.38	3.30
strawberry	0	0	0	4.10	5.51
digital	0.18	0.01	0	0	0
information	0.02	0.09	0.28	0	0

m×n



Word2Vec Skip-gram with Negative sampling

Word2Vec Skip-gram with Negative Sampling (SGNS)



Mikolov, Tomás ‘Efficient Estimation of Word Representations in Vector Space’. *1st International Conference on Learning Representations, ICLR 2013, Scottsdale, Arizona, USA, May 2-4, 2013, Workshop Track Proceedings*.

Negative Sampling

Vanilla Skip-Gram

W_output (old) − Learning R. × grad_W_output = W_output (new)

W_output (old)

-0.560	0.340	0.160
-0.910	-0.440	1.560
-1.210	-0.130	-1.320
1.670	-0.150	-1.030
1.720	-1.460	0.730
0.000	1.390	-0.120
-0.060	1.520	-0.790
0.800	1.850	-1.670
-1.370	1.320	-0.480
0.670	1.990	-1.850
-1.520	-1.740	-1.860

(11X3)

−

Learning R.

0.05

×

grad_W_output

0.064	0.071	-0.014
0.098	0.015	0.063
0.069	0.089	0.045
0.014	0.085	0.079
-0.021	0.067	0.071
-0.098	-0.088	0.091
-0.072	-0.078	-0.089
0.046	-0.079	-0.053
-0.049	-0.087	0.025
-0.060	0.092	0.042
0.074	0.050	0.070

(11X3)

=

W_output (new)

-0.563	**0.336**	**0.161**
-0.915	**-0.441**	**1.557**
-1.213	**-0.134**	**-1.322**
1.669	**-0.154**	**-1.034**
1.721	**-1.463**	**0.726**
0.005	**1.394**	**-0.125**
-0.056	**1.524**	**-0.786**
0.798	**1.854**	**-1.667**
-1.368	**1.324**	**-0.481**
0.673	**1.985**	**-1.852**
-1.524	**-1.743**	**-1.864**

(11X3)

aegis4048.github.io

Negative Sampling

W_output (old) − Learning R. × grad_W_output = W_output (new)

W_output (old)

-0.560	0.340	0.160
-0.910	-0.440	1.560
-1.210	-0.130	-1.320
1.670	-0.150	-1.030
1.720	-1.460	0.730
0.000	1.390	-0.120
-0.060	1.520	-0.790
0.800	1.850	-1.670
-1.370	1.320	-0.480
0.670	1.990	-1.850
-1.520	-1.740	-1.860

(11X3)

−

Learning R.

0.05

×

grad_W_output

Not computed!

Positive sample, w_o	0.031	0.030	0.041
Negative sample, k=1	-0.090	0.031	-0.065
Negative sample, k=2	0.056	0.098	-0.061
Negative sample, k=3	0.069	0.084	-0.044

(11X3)

=

W_output (new)

-0.560	0.340	0.160
-0.910	-0.440	1.560
-1.210	-0.130	-1.320
1.670	-0.150	-1.030
1.720	-1.460	0.730
0.000	1.390	-0.120
-0.060	1.520	-0.790
0.798	**1.849**	**-1.672**
-1.366	**1.318**	**-0.477**
0.667	**1.985**	**-1.847**
-1.523	**-1.744**	**-1.858**

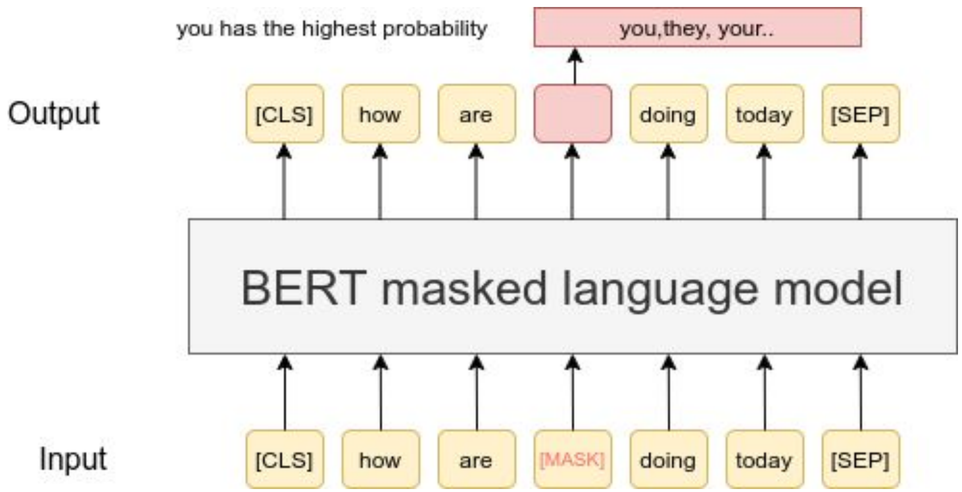
(11X3)

aegis4048.github.io

aegis4048.github.io

BERT-based models

BERT-based models

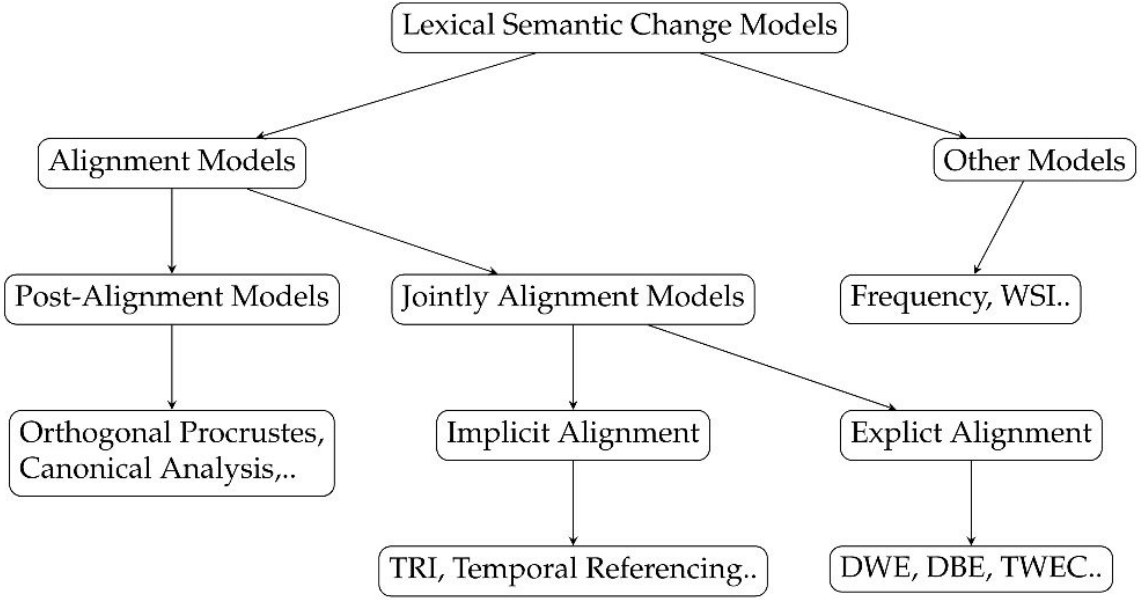


Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. [BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding](). In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

type vs token embeddings

- type-based (static embeddings): Word2Vec, FastText, GloVe, ...
- token-based (contextualized embeddings): BERT, eLMo, roBERTa, ...

Lexical Semantic Change Models



P. Cassotti, P. Basile, M. de Gemmis, and G. Semeraro, “Analyzing Gaussian distribution of semantic shifts in Lexical Semantic Change Models,” IJCoL Ital. J. Comput. Linguist., vol. 6, no. 6–2, pp. 23–36, 2020.

Alignment Models

Alignment approach

Post-alignment

- Post-alignment models first train static word embeddings for each time slice and then align them

Jointly alignment

- Jointly Alignment models train word embeddings and jointly align vectors across all time slices

- Jointly Alignment models can be distinguished in Explicit alignment models and Implicit alignment models.

Alignment approach

Post-alignment

- Post-alignment models first train static word embeddings for each time slice and then align them

Jointly alignment

- Jointly Alignment models train word embeddings and jointly align vectors across all time slices

- Jointly Alignment models can be distinguished in Explicit alignment models and Implicit alignment models.

!! Post-alignment and Explicit alignment rely on the assumption that only few words change their meaning

Orthogonal Procrustes (OP)

Embedding matrix
time t

$$R = \arg\min_{Q^TQ=I} \|QW^t - W^{t+1}\|_F$$

Rotation matrix

Embedding matrix
time t+1

William L. Hamilton, Jure Leskovec, and Dan Jurafsky. 2016. [Diachronic Word Embeddings Reveal Statistical Laws of Semantic Change](). In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1489–1501, Berlin, Germany. Association for Computational Linguistics.

Jointly Alignment - Alignment constraint

Explicit alignment

- The objective function of explicit alignment models involves constraints on word vectors

- Typically those constraints require that the distance of two-word vectors in two consecutive periods is the smallest possible

Implicit alignment

- In the implicit alignment, the alignment is automatically performed by sharing the same word context vectors across all the time spans

Dynamic Word Embedding (DWE)

PMI factorization

Regularization term

$$\min_{U(t)} \frac{1}{2} \boxed{\|Y(t) - U(t)U(t)^T\|_F^2} + \frac{\lambda}{2} \boxed{\|U(t)\|_F^2} +$$

$$\frac{\tau}{2} \left(\boxed{\|U(t-1) - U(t)\|_F^2} + \boxed{\|U(t) - U(t+1)\|_F^2} \right)$$

Explicit temporal constraints

Yao, Zijun, et al. "Dynamic word embeddings for evolving semantic discovery." *Proceedings of the eleventh acm international conference on web search and data mining*. 2018.

Dynamic Bernoulli Embedding (DBE)

$$\mathcal{L}(\boldsymbol{\rho}, \boldsymbol{\alpha}) = \mathcal{L}_{\text{pos}} + \mathcal{L}_{\text{neg}} + \mathcal{L}_{\text{prior}}$$

$$\mathcal{L}_{\text{pos}} = \sum_{i=1}^{N} \sum_{v=1}^{V} x_{iv} \log \sigma(\boxed{\eta_{iv}})$$

$$\mathcal{L}_{\text{neg}} = \sum_{i=1}^{N} \sum_{v=1}^{V} (1 - x_{iv}) \log(1 - \sigma(\boxed{\eta_{iv}}))$$

$$\eta_{iv} = \rho_v^{(t_i)\top} \left(\sum_{j \in c_j} \sum_{v'} \alpha_{v'} x_{jv'} \right)$$

embedding vector $\rho_v \in \mathbb{R}^K$

context vector $\alpha_v \in \mathbb{R}^K$

$$\mathcal{L}_{\text{prior}} = \log p(\boldsymbol{\alpha}) + \log p(\boldsymbol{\rho})$$

$$\log p(\boldsymbol{\alpha}) = -\frac{\lambda_0}{2} \sum_v \boxed{||\alpha_v||^2}$$

$$\log p(\boldsymbol{\rho}) = -\frac{\lambda_0}{2} \sum_v \boxed{||\rho_v^{(0)}||^2} - \frac{\lambda}{2} \sum_{v,t} \boxed{||\rho_v^{(t)} - \rho_v^{(t-1)}||^2}$$

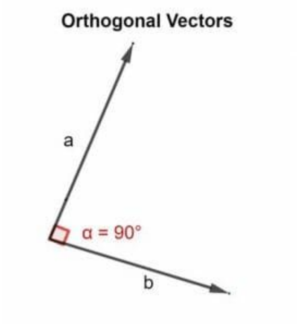
Regularization term

Explicit temporal constraints

Rudolph, Maja, and David Blei. "Dynamic embeddings for language evolution." *Proceedings of the 2018 World Wide Web Conference*. 2018.

Temporal Random Indexing (TRI)

- Produce aligned word embeddings in a single step.



- Count-based method.

- TRI is based on Random Indexing: near-orthogonality random index vectors shared across all time slices so that word spaces are comparable.

$$sv_i = \sum_{d \in C} \sum_{-m<i<+m} c_i$$

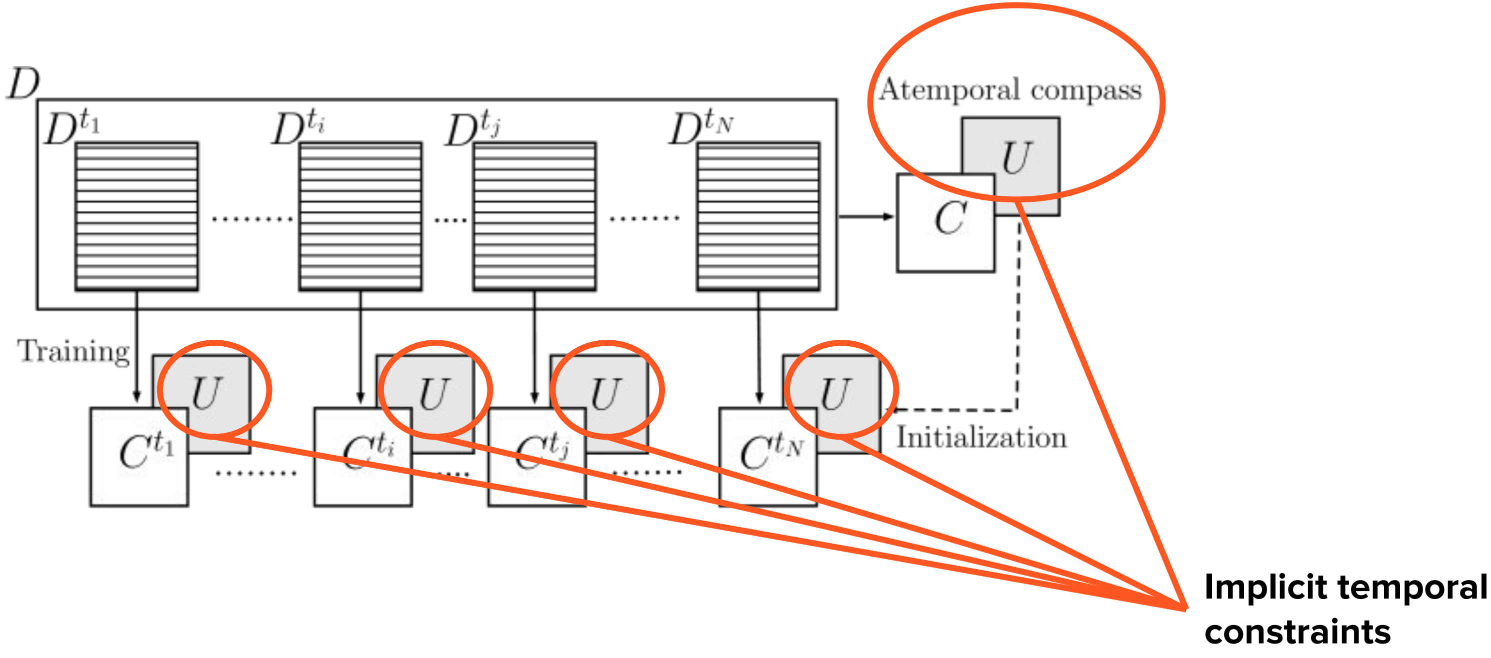
Caputo, Annalina, Pierpaolo Basile, and Giovanni Semeraro. "Temporal random indexing: A system for analysing word meaning over time." *IJCoL. Italian Journal of Computational Linguistics* 1.1-1 (2015): 61-74.

Temporal Referencing (TR)

- Replace a subset of words in the dictionary (target words) with time-specific tokens
- Temporal Referencing is not performed when the word is considered a context word
- Since TR is a generic framework, it can be applied to both low-dimensional embeddings learned with SGNS and high-dimensional sparse PPMI vectors

Haim Dubossarsky, Simon Hengchen, Nina Tahmasebi, and Dominik Schlechtweg. 2019. [Time-Out: Temporal Referencing for Robust Modeling of Lexical Semantic Change](). In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 457–470, Florence, Italy. Association for Computational Linguistics.

Temporal Word Embedding with a Compass (TWEC)

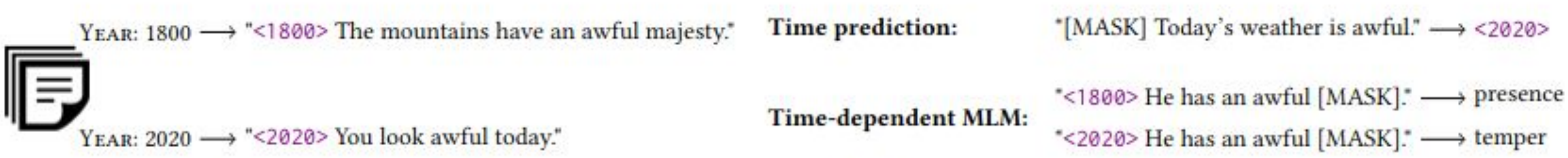


Di Carlo, Valerio, Federico Bianchi, and Matteo Palmonari. "Training temporal word embeddings with a compass." *Proceedings of the AAAI conference on artificial intelligence*. Vol. 33. No. 01. 2019.

Contextualized Models

TempoBERT

- Use time as additional context
- Exploit time masking



(a) TempoBERT is trained on temporal corpora, where each sequence is prepended with temporal context information.

(b) TempoBERT can be used for inference in two modes: (1) time prediction; (2) time-dependent mask filling.

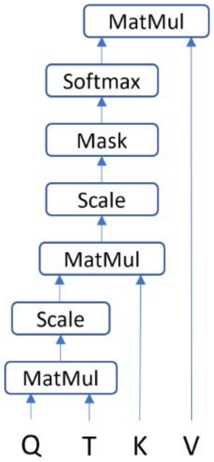
Rosin, Guy D., Ido Guy, and Kira Radinsky. "Time masking for temporal language models." *Proceedings of the Fifteenth ACM International Conference on Web Search and Data Mining*. 2022.

Temporal Attention

- Extends self-attention to include time dimension

$$TemporalAttention(Q, K, V, T) = \text{softmax}\left(\frac{Q \boxed{\frac{T^{\intercal} T}{\|T\|}} K^{\intercal}}{\sqrt{d_k}}\right) V$$

Time-specific weight matrix



Rosin, Guy D., and Kira Radinsky. "Temporal Attention for Language Models." *arXiv preprint arXiv:2202.02093* (2022).

XLM-RoBERTa

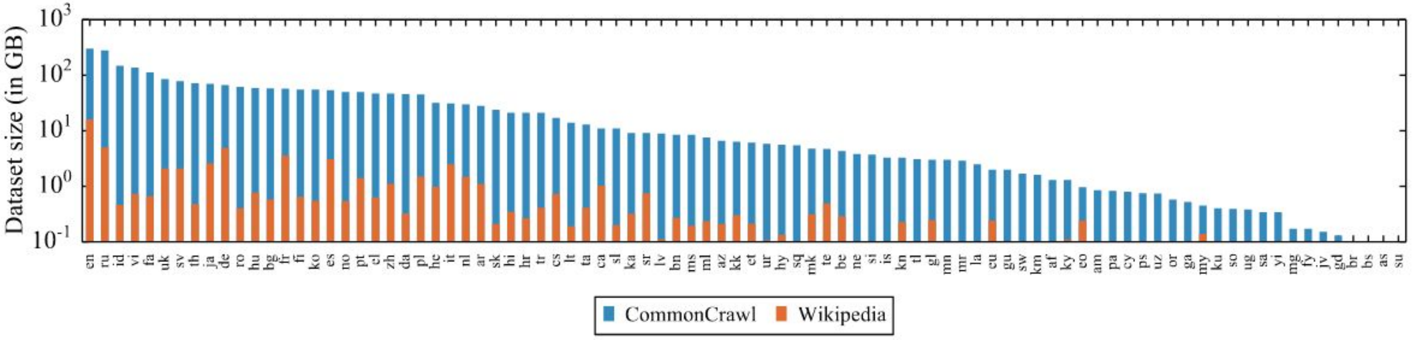


Figure 1: Amount of data in GiB (log-scale) for the 88 languages that appear in both the Wiki-100 corpus used for mBERT and XLM-100, and the CC-100 used for XLM-R. CC-100 increases the amount of data by several orders of magnitude, in particular for low-resource languages.

Sebastian Ruder, Anders Søgaard, and Ivan Vulić. 2019. [Unsupervised Cross-Lingual Representation Learning](). In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics: Tutorial Abstracts*, pages 31–38, Florence, Italy. Association for Computational Linguistics.

Gloss Reader

- Rely on XLM-RoBERTa and trained on a English Word Sense Disambiguation (WSD) dataset (SemCor)
- Zero-shot ability on other languages such as Russian

The $\boxed{\text{bank}}$ can guarantee deposits will eventually cover future tuition costs because it invests in adjustable-rate mortgage securities. → **Context Encoder**

bank^1	Gloss:	$\boxed{\text{a financial institution that accepts deposits and channels the money into lending activities}}$
bank^2	Gloss:	sloping land (especially the slope beside a body of water)

→ **Gloss Encoder**

Rachinskiy, Maxim, and Nikolay Arefyev. "Zeroshot Crosslingual Transfer of a Gloss Language Model for Semantic Change Detection." *Computational linguistics and intellectual technologies: Papers from the annual conference Dialogue*. 2021.

Deep Mistake

- Pretrained XLM-R fintuned on MCL-WiC task
- Not depends on fixed sense inventories

Lang	Target	Context-1	Context-2	Label
EN	Beat	We beat the competition	Agassi beat Becker in the tennis championship.	True
DA	Tro	Jeg tror p˚a det, min mor fortalte.	Maria troede ikke sine egne øjne.	True
ET	Ruum	Uhel hetkel olin v̄ aljaspool aega ja ruumi.	Umberringi oli l̄ oputu t̄ uhi ruum.	True
FR	Causticité	Sa causticité lui a fait bien des ennemis.	La causticité des acides.	False
KO	틀림	틀림이 있는지 없는지 세어 보시오.	그 아이 하는 짓에 틀림이 있다면 모두 이 어미 죄이지요.	False
ZH	發	建築師希望發大火燒掉城市的三分之一。	如果南美洲氣壓偏低，則印度可能發乾旱	True
FA	صرف	صرف غذا نیم ساعت طول کشید	معلم صرف افعال ماضی عربی را آموزش داد	False

Arefyev, Nikolay, et al. "DeepMistake: Which Senses are Hard to Distinguish for a WordinContext Model." *Computational linguistics and intellectual technologies: Papers from the annual conference Dialogue*. 2021.

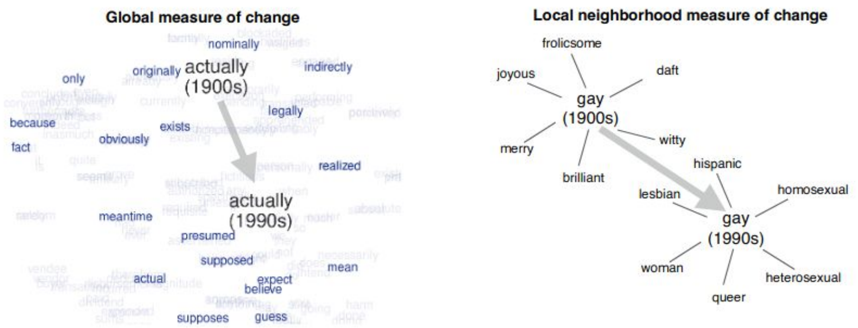
Other models

Local Neighborhood measure

- Global measure

$$d^G(w_i^{(t)}, w_i^{(t+1)}) = \text{cos-dist}(\mathbf{w}_i^{(t)}, \mathbf{w}_i^{(t+1)})$$

- Local Neighborhood measure



$$s^{(t)}(j) = \text{cos-sim}(\mathbf{w}_i^{(t)}, \mathbf{w}_j^{(t)})$$

$$\forall w_j \in \boxed{\mathcal{N}_k(w_i^{(t)})} \cup \boxed{\mathcal{N}_k(w_i^{(t+1)})}$$

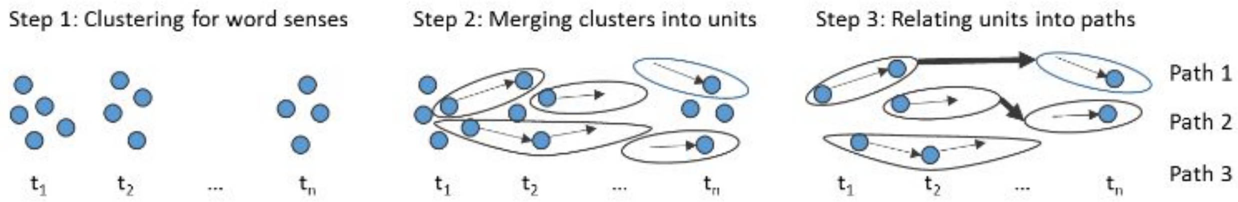
$$d^L(w_i^{(t)}, w_i^{(t+1)}) = \text{cos-dist}(\mathbf{s}_i^{(t)}, \mathbf{s}_i^{(t+1)})$$

***k* nearest-neighbors**

William L. Hamilton, Jure Leskovec, and Dan Jurafsky. 2016. Cultural Shift or Linguistic Drift? Comparing Two Computational Measures of Semantic Change. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 2116–2121, Austin, Texas. Association for Computational Linguistics.

Word Sense Induction

- Curvature clustering
- *lin* measure (based on the WordNet synset similarity)



Nina Tahmasebi and Thomas Risse. 2017. Finding Individual Word Sense Changes and their Delay in Appearance. In *Proceedings of the International Conference Recent Advances in Natural Language Processing, RANLP 2017*, pages 741–749, Varna, Bulgaria.

Grammatical Features

- Grammatical features such as PoS tags, dependency labels, number, case, tense
- Grammatical features are language-dependent
- Interpretable results

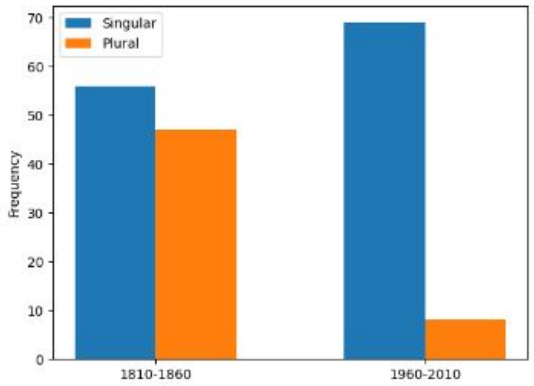


Figure 1: Changes in the number category distribution for the English noun '*lass*' over time, calculated on the English corpora of the SemEval 2020 shared task 1 (Schlechtweg et al., 2020). '*Lass*' is annotated as semantically changed in the SemEval dataset.

Andrey Kutuzov, Lidia Pivovarova, and Mario Giulianelli. 2021. Grammatical Profiling for Semantic Change Detection. In *Proceedings of the 25th Conference on Computational Natural Language Learning*, pages 423–434, Online. Association for Computational Linguistics.