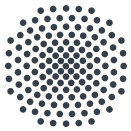

University of Stuttgart
Germany

Bachelorthesis
Optimierung von Clustern von Wortverwendungsgraphen

November 30, 2021

Benjamin Tunc

Institut für Maschinelle Sprachverarbeitung, Universität Stuttgart

Motivation

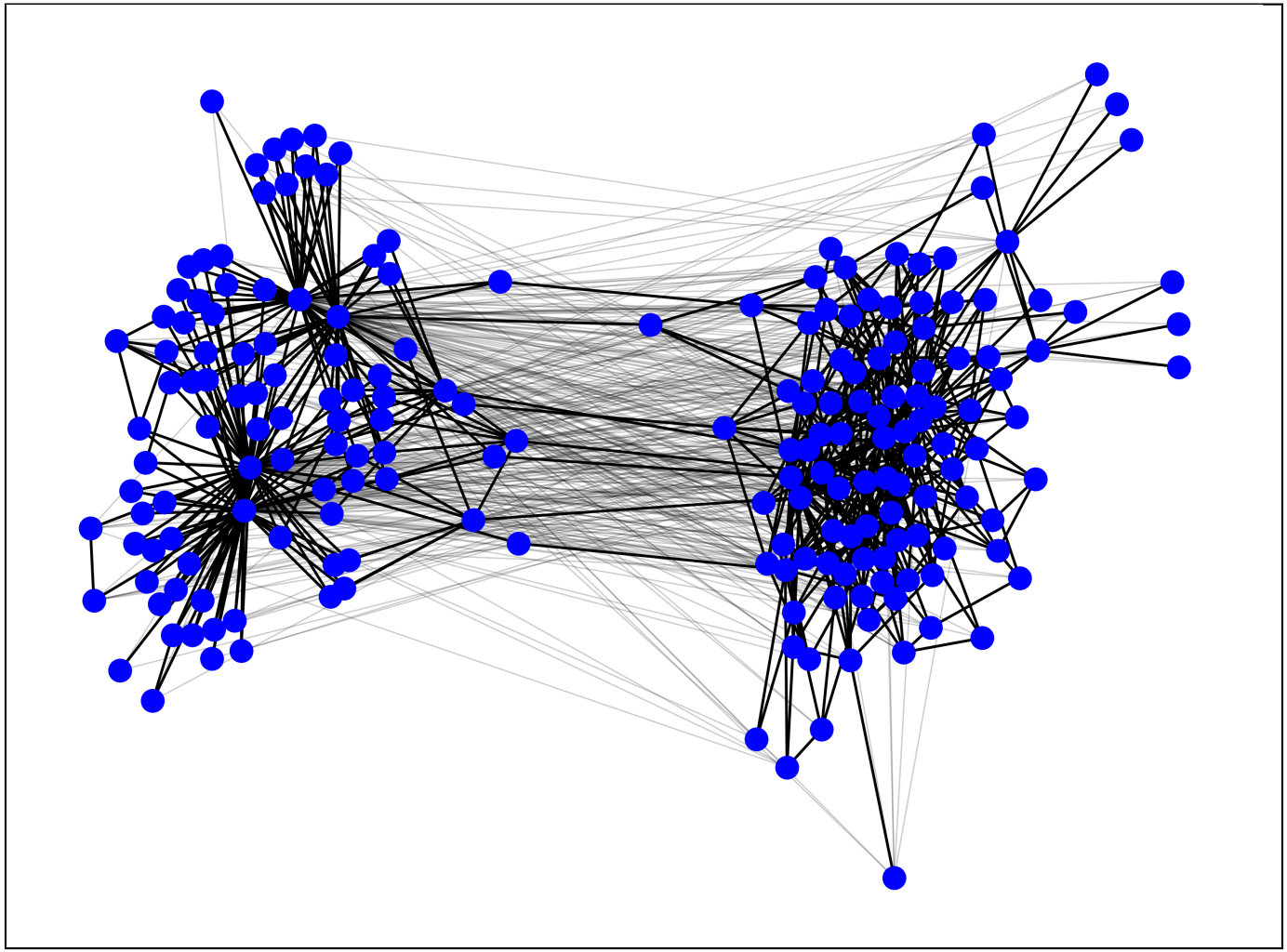

Figure 1: Wortverwendungsgraph (WUG) von *zersetzen* aus DWUG DE[1]

[1] https://www.ims.uni-stuttgart.de/data/wugs

Motivation (2)

SemEval-2020 Task 1:
Unsupervised Lexical Semantic Change Detection
Schlechtweg et al. (2020)

Datensätze

- DWUG Datensätze (EN V1.0.0, DE V1.1.0, SV 1.0.0) [2]
- 30-50 Wörter mit je 200 Verwendungen pro Datensatz
- für ihre semantische Nähe auf der DURel relatedness scale annotiert

[2] https://www.ims.uni-stuttgart.de/data/wugs

Datensätze (2)

4: Identical
3: Closely Related
2: Distantly Related
1: Unrelated

Table 1: DURel relatedness scale.

Sense descriptions

- bei 24 deutschen Wörtern
- Beschreibung des Wortsinns
- Annotatoren wählen aus einem Set von verschiedenen Wortbedeutungen ein passendes aus

Correlation Clustering

- Methode zur Aufteilung der Knoten eines gewichteten Graphen $\mathbf{G} = (\mathbf{U}, \mathbf{E}, \mathbf{W})$ in eine optimale Anzahl von Clustern
- Kantengewicht $W(e)$ der Kanten $e = (u, v) \in E$ sind binär $W(e) \in \{-1, 1\}$
- Minimierung der Summe der positiven Kantengewichte zwischen verschiedenen Clustern und die Summe der negativen Kantengewichte innerhalb von Clustern

DWUG Correlation Clustering

- Kantengewichte sind nicht-binär
- die Gewichte $W(e)$ werden nach $W'(e) = W(e) - 2.5$ verschoben

$$L(C) = \sum_{e \in \phi_{E,C}} W'(e) + \sum_{e \in \psi_{E,C}} |W'(e)| \tag{1}$$

Simulated Annealing

```
state ← initial
while attempts < maxA and i < maxI do
    temp ← T(i)
    i ← i + 1
    if temp = 0 then
        break
    else
        delta_e ← L(nghbr) − L(state)
        prob ← exp(delta_e/temp)
        random ← random(0, 1)
        if delta_e > 0 or random < prob then
            state ← nghbr
            attempts ← 0
        else
            attempts ← attempts + 1
        end if
    end if
end while
```

Figure 2: Pseudocode umgewandelt von mlrose Simulated Annealing [3]

[3] `https://github.com/gkhayes/mlrose`.

Parameter

- **Maximale Clusteranzahl s**:
  - keine feste Clusteranzahl k
  - iterieren durch $0 \leq k \leq s$
  - $s \in \{5, 7, 10, 15, 20\}$

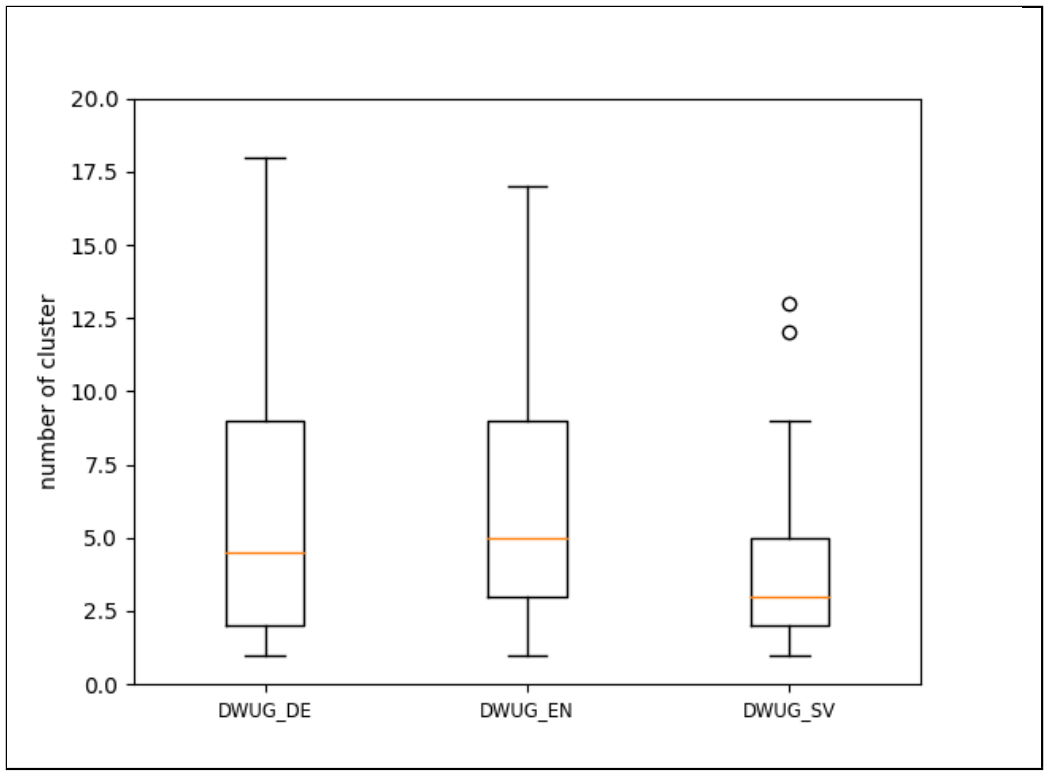

Figure 3: Anzahl von Clustern pro Datensatz.

Parameter (2)

- **MaxA und MaxI**:
  - $maxA$ = maximale Anzahl der Versuche, in einen Nachbarzustand zu wechseln
  - $maxI$ = maximale Anzahl der Iterationen, die der Algorithmus durchläuft
  - $maxA/maxI \in \{100/10000, 100/20000, 500/10000, 1000/10000, 1000/20000, 5000/10000, 5000/20000\}$
- **Wiederholungen**
- **Initialisierung**:
  - unabhängig: ein Clustering mit einer zufälligen Initialisierung und ein Clustering, welches mit Connected Compontents initalisiert wird (Schlechtweg et al., 2021)
  - abhängig:ein Clustering mit einer zufälligen Initialisierung und ein Clustering initialisiert mit der bisdato besten Lösung

Parameter (3)

- **Stoppkriterien**:
  - Feste Anzahl von Wiederholungen ($r = 5$, $r = 10$)
  - Vergleich mit der letzten Wiederholung ($r = l1$)
  - Vergleich mit den letzten drei Wiederholungen ($r = l3$)

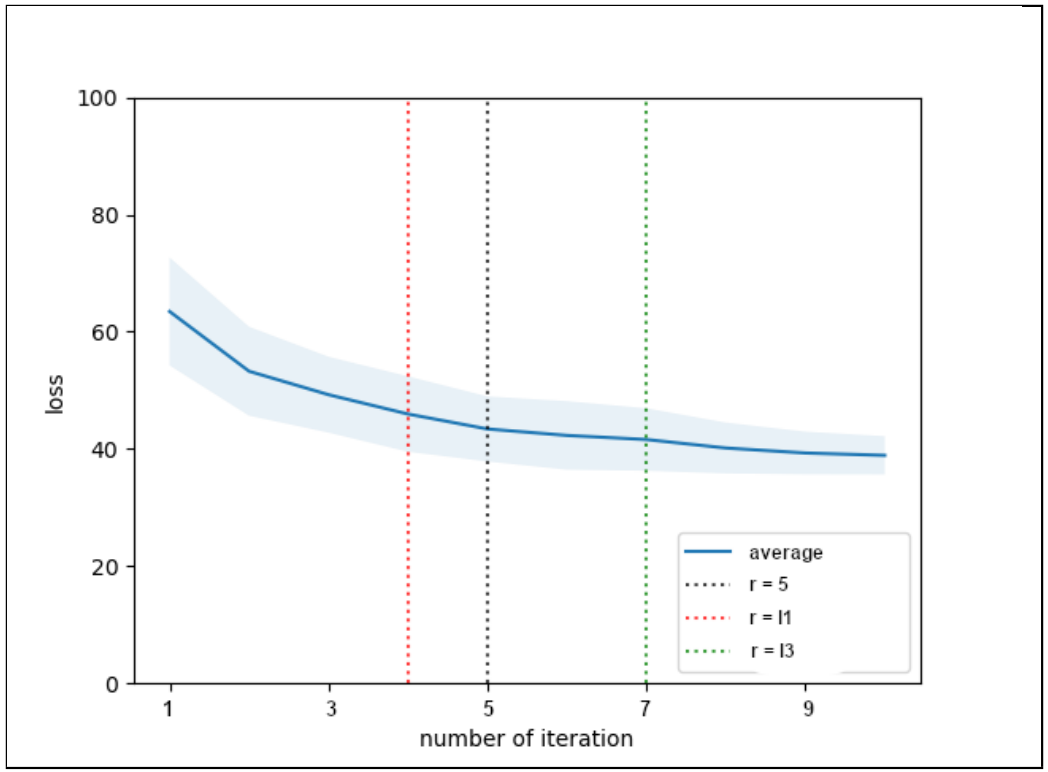

Figure 4: Verlauf eines Modells beim Wort *Knotenpunkt*

Evaluierungsmetrik

- **Loss**
- **Laufzeit**
- **Adjusted Rand Index (*ARI*)**
  - Adjusted Rand Index zwischen einer Clusterlösung und den sense descriptions
  - zwischen -1.0 und 1.0
- **Robustheit**
  - Adjusted Rand Index zwischen Clusterlösungen des gleichen Modells
  - zwischen -1.0 und 1.0

Experimente

- **Modellstruktur:** s, $maxA$, $maxI$, $Initialisierung$, r
- 10 Durchläufe pro Modell
- Durchschnitt der Medianwerte eines Modells
- 320 Modelle insgesamt
- Zusatzmodell: $s = 20$, $maxA = 2000$, $maxI = 50000$, unabhängig initialisiert und $r = 5$

Resultate - Welches Modell findet den niedrigsten Loss?

s	MaxA	MaxI	Init	r	Loss	Laufzeit	ARI	Robust
10	500	10k	a.	10/5/l3	**8.0**	30/15/**17**	.72/.70/.71	.97/.96/.96
15	500	10k	a.	10	**8.0**	36	.71	.98
20	500	10k	a.	10	**8.0**	43	**.73**	.98
10	500	20k	a.	10/l3	**8.0**	51/27	.72/.72	.98/.97
15	500	20k	a.	10	**8.0**	51	**.73**	.98
20	500	20k	a.	10	**8.0**	62	**.73**	**.99**
10	1000	10k	a.	10/l3	**8.0**	31/**17**	**.73**/.72	.98/.97
15	1000	20k	a.	10	**8.0**	64	**.73**	.98
15	5000	20k	a.	10	**8.0**	32	.72	.98

Table 2: Übersicht über alle Modelle mit dem niedrigsten Median-Loss.

Resultate - Welches Modell findet den niedrigsten Loss?

Muster der Topmodelle:

$s \in \{10, 15, 20\}$, $maxA \in \{500, 1000, 5000\}$,
$maxI \in \{10000, 20000\}$ und abhängig initialisiert

Resultate - Sind die Wiederholungen sinnvoll?

- $S = 20$, $maxA = 5000$, $maxI = 50000$, $r = 1$
- Loss von 13.0 und Laufzeit von 29s
- $S = 20$, $maxA = 500$, $maxI = 10000$, abhängig Initialisiert, $r = I3$
- Loss von 9.55 und Laufzeit von 22s

Resultate - Welches Modell ist besonders effizient?

- Nur Modelle mit einem Loss-Median von 10.0 (16%)
- Vergleich der Laufzeiten
- $s = 10$, $maxA = 500$, $maxI = 10000$, abhängige Initialisierung und $r = I1$ mit Loss von 9.35 und Laufzeit von 7s

Resultate - Stoppkriterien im Vergleich

r	ARI	Loss	Robust	Laufzeit
10	**.72**	**9.5**	**.97**	64.6
5	.69	11.6	.95	32.5
l1	.68	12.0	.94	**14.7**
l3	.70	10.5	.95	35.1

Table 3: Metrik für Stoppkriterien (Deutscher Datensatz).

Resultate - niedrigerer Loss = höhere Robustheit?

- Spearman's rank correlation coefficient von -0.73
- alle Modelle aus Tabelle 2 haben eine hohe Robustheit zwischen 0.96 and 0.99

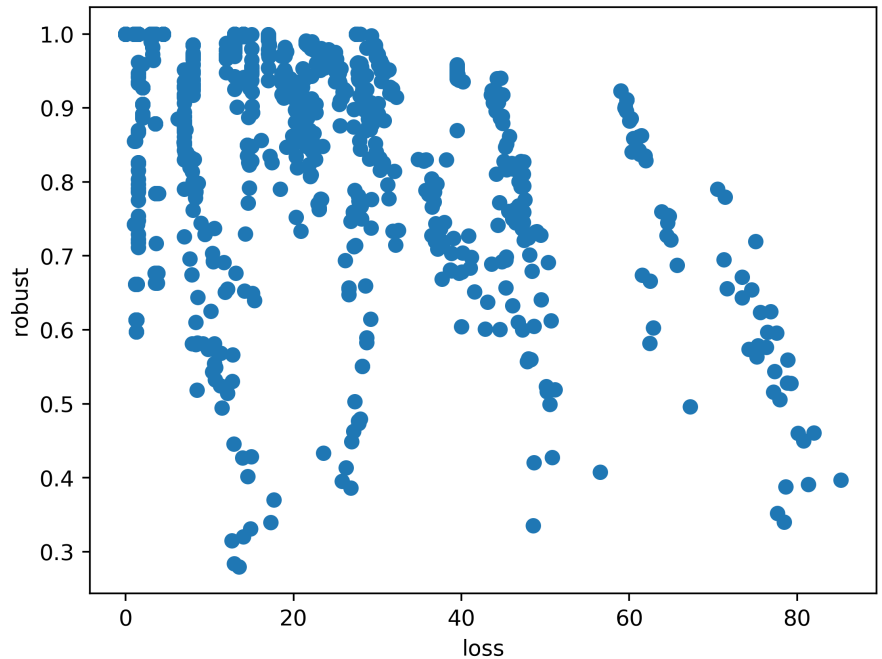

Figure 5: Loss und Robustheit aller Clusterlösungen.

Resultate - Welches Modell erzielt den höchsten ARI?

- Höchster ARI (0.73) wurde von 16 Modellen erzielt
- alle Modelle aus der Tabelle 2 haben einen annähernd guten *ARI*-Wert
- 5 Modelle haben auch den niedrigsten Median-Loss
- Die anderen Modelle haben vergleichsweise gute Loss-Werte

Resultate - Gibt es einen Zusammenhang?

- Spearman Correlation von -0.45
- Bei 19 von 24 deutschen Wörtern mit sense descriptions ist die Clusterlösung mit dem höchsten *ARI* auch die Lösung mit dem geringsten Loss

Resultate - unabhängige Initialisierung oder abhängige Initialisierung?

Init.	Loss	Laufzeit	ARI	Robust
unabhängig	12.7	28	.62	.92
abhängig	**10.6**	**27**	**.70**	**.96**

Table 4: Vergleich von unabhängiger Initialisierung und abhängiger Initialisierung.

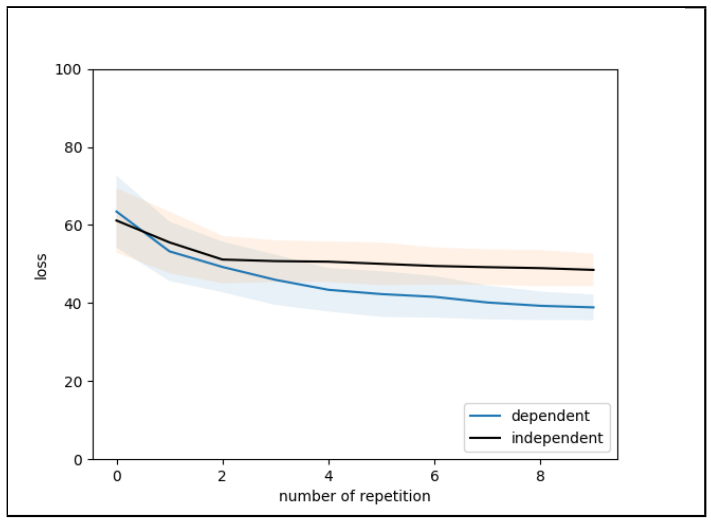

Figure 6: Vergleich von unabhängiger Initialisierung und abhängiger Initialisierung beim Wort *Knotenpunkt*.

Resultate - Liegt es an Connected Components?

- Unterschied von 0.5 im Loss-Median $\rightarrow$ kein nennenswerter Unterschied

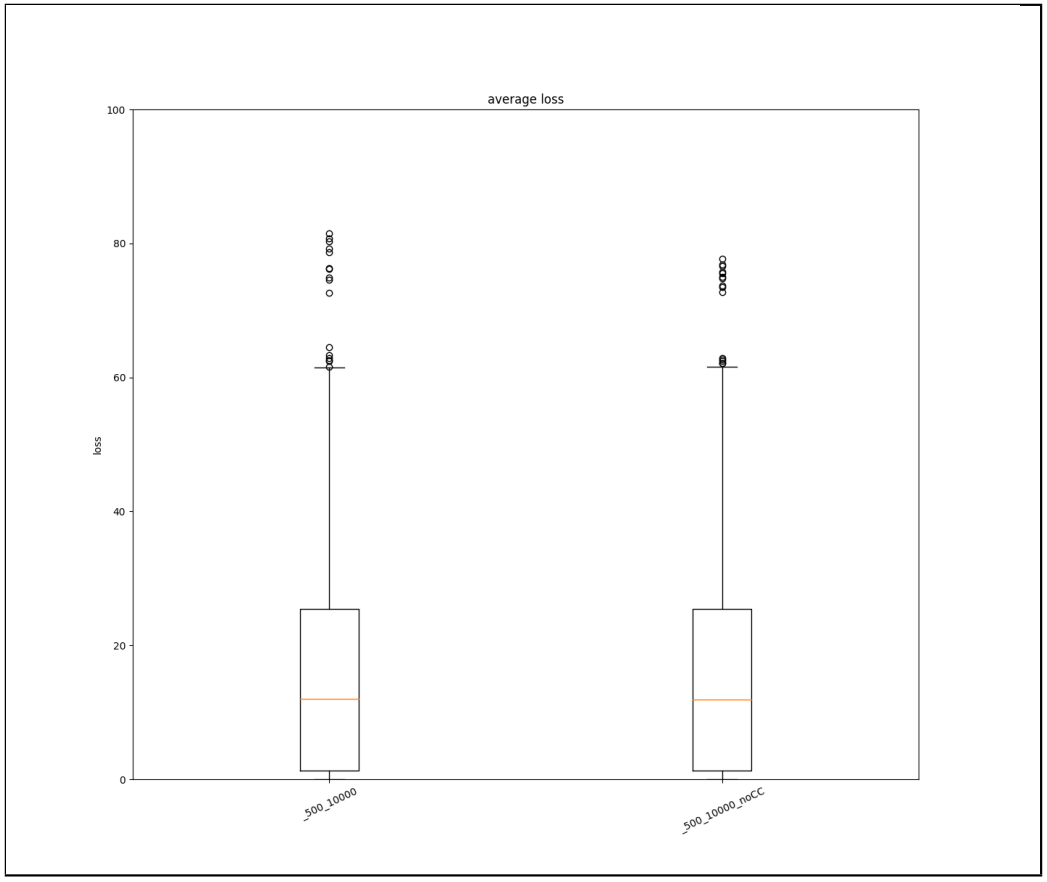

Figure 7: Vergleich von unabhängiger Initialisierungen mit und ohne Conected Components.

Resultate - Wieso? (2)

- Median-Loss ohne zufällige Initialisierung: 24.0
- Median-Loss mit zufällige Initialisierung: 9.8

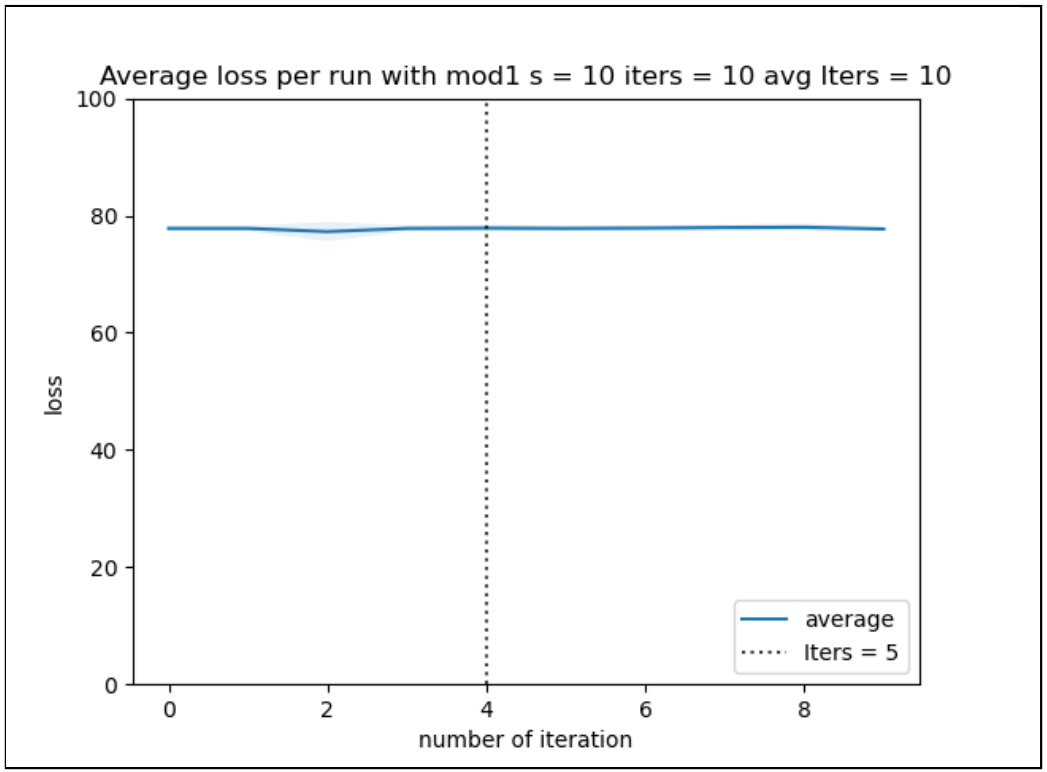

Figure 8: Modell beim Wort *Knotenpunkt* ohne die zufällige Initialisierung.

Resultate - Welche Parameterkombination für Simulated Annealing ist zu empfehlen?

MaxA	MaxI	Loss	Laufzeit	ARI	Robust
100	10000	12.9	**14**	.63	.89
100	20000	12.8	15	.63	.89
500	10000	**11.0**	25	.67	**.96**
500	20000	11.1	42	.67	**.96**
1000	10000	11.2	25	.67	**.96**
1000	20000	11.3	46	**.68**	**.96**
5000	10000	11.1	26	**.68**	**.96**
5000	20000	11.4	46	**.68**	**.96**

Table 5: Durchschnittliche Resultate von verschiedenen *maxI* und *maxA* Kombinationen.

Resultate - Wie sieht es bei anderen Datensätzen aus?

- **Schwedisch**:
  - Der niedrigste Loss (2.5) wurde mit 9 unabhängigen Modellen und 29 abhängigen Modellen erzielt
  - Stoppkriterien $r = l/3$ und $r = 10$ sind am häufigsten

MaxA	**MaxI**	**Loss**	**Laufzeit**	**Robust**
100	10000	2.9	**9**	.93
100	20000	3.0	11	.89
500	10000	2.8	15	.93
500	20000	2.8	29	**.98**
1000	10000	2.8	15	**.98**
1000	20000	2.8	27	**.98**
5000	10000	**2.7**	23	**.98**
5000	20000	2.8	27	.97

Table 6: Durchschnittliche Resultate von verschiedenen *maxI* und *maxA* Kombinationen (Schwedischer Datensatz).

Resultate - Wie sieht es bei anderen Datensätzen aus? (2)

r	Loss	Robust	Laufzeit
10	**2.7**	**.98**	35.5
5	2.8	.96	17.7
l1	3.0	.96	**7.5**
l3	2.8	.96	16.5

Table 7: Metrik für Stoppkriterien (Schwedischer Datensatz).

Resultate - Wie sieht es bei anderen Datensätzen aus? (3)

- **Englisch**:
  - Niedrigster Loss (14.0) wurde von einem unabhängigen Modell gefunden
  - Niedrigster Loss eines abhängigen Modells: 14.9
  - Durchschnittlicher Loss der Modelle: 16.5

MaxA	**MaxI**	**Loss**	**Laufzeit**	**Robust**
100	10000	17.7	**11**	.82
100	20000	18.0	12	.82
500	10000	16.4	19	**.92**
500	20000	**16.1**	33	.91
1000	10000	16.3	20	**.92**
1000	20000	16.2	36	**.92**
5000	10000	16.9	30	.91
5000	20000	16.3	37	**.92**

Table 8: Durchschnittliche Resultate von verschiedenen *maxI* und *maxA* Kombinationen (Englischer Datensatz).

Resultate - Wie sieht es bei anderen Datensätzen aus? (4)

r	Loss	Robust	Laufzeit
10	**16.2**	**.91**	45.0
5	16.7	.89	22.4
l1	17.0	.89	**9.9**
l3	16.8	.89	21.5

Table 9: Metrik für Stoppkriterien (Englischer Datensatz).

Zusammenfassung

- **Initialisierung**:
  - abhängige Initialisierung erzielt bessere Resultate
  - eine rein abhängige Initialisierung wird wahrscheinlich in einem lokalen Minimum stecken bleiben und sollte von einer zusätzlichen zufälligen Initialisierung ergänzt werden
- **Wiederholungen und Stoppkriterien**:
  - eine mehrmalige Wiederholung des Clustering ist besser als eine einmalige Durchführung mit Brute-Force-Parametern
  - Ermöglicht die Nutzung von Stoppkriterien zur Laufzeitreduzierung ohne großen Qualitätsverlust
- **ARI, Loss, Robustheit**:
  - Verringerung des Loss führt zu einer besseren externen Qualität
  - Minimierung des Loss führt zu hohe Robustheit
- **Datensätze**:
  - gleiche Resultate in Bezug auf Stoppingkriterien
  - gleiche Resultate in Bezug auf Parameterkombinationen
  - gleiche Resultate bei DWUG SV in Bezug auf Initialisierung
  - andere Resultate bei DWUG EN in Bezug auf Initialisierung

References I

Bansal, N., Blum, A., & Chawla, S. (2004). Correlation clustering. *Machine Learning*, *56*(1-3), 89–113. doi: 10.1023/B:MACH.0000033116.57574.95

Hayes, G. (2019). *mlrose: Machine Learning, Randomized Optimization and SEarch package for Python.* `https://github.com/gkhayes/mlrose`. (Accessed: May 22, 2020)

Hubert, L., & Arabie, P. (1985). Comparing partitions. *Journal of Classification*, *2*, 193–218.

McCarthy, D., Apidianaki, M., & Erk, K. (2016). Word sense clustering and clusterability. *Computational Linguistics*, *42*(2), 245-275.

Pincus, M. (1970). A monte carlo method for the approximate solution of certain types of constrained optimization problems. *Operations Research*, *18*(6), 1225–1228. doi: 10.1287/opre.18.6.1225

Schlechtweg, D., McGillivray, B., Hengchen, S., Dubossarsky, H., & Tahmasebi, N. (2020). SemEval-2020 Task 1: Unsupervised Lexical Semantic Change Detection. In *Proceedings of the 14th International Workshop on Semantic Evaluation.* Barcelona, Spain: Association for Computational Linguistics. Retrieved from `https://www.aclweb.org/anthology/2020.semeval-1.1/`

Schlechtweg, D., Schulte im Walde, S., & Eckmann, S. (2018). Diachronic Usage Relatedness (DURel): A framework for the annotation of lexical semantic change. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies* (pp. 169–174). New Orleans, Louisiana. Retrieved from `https://www.aclweb.org/anthology/N18-2027/`

Schlechtweg, D., Tahmasebi, N., Hengchen, S., Dubossarsky, H., & McGillivray, B. (2021). DWUG: A large Resource of Diachronic Word Usage Graphs in Four Languages.. Retrieved from `https://arxiv.org/abs/2104.08540`