Projektbericht– Computerlinguistische Techniken

Eszter Bukovszky, 824911

31.03.2025

Einleitung

Im Rahmen dieses Projekts habe ich ein System zur automatischen Terminologieextraktion aus einem Domänenkorpus der ACL-Publikationen und einem Referenzkorpus (Reuters-Korpus) entwickelt. Ziel war es, durch geeignete Metriken domänenspezifische Begriffe identifizieren zu können und diese mit einem Goldstandard zu evaluieren. Das manuelle Erstellen domänenspezifischer Terminologien ist aufwändig, weshalb automatische Verfahren eine bedeutende Erleichterung darstellen. Ziel war es daher, ein robustes, skalierbares System zu implementieren, das rein auf statistischer Analyse basiert und sich effizient auf große Datenmengen anwenden lässt.

Methodenüberblick

Die Extraktion basiert auf zwei zentralen Metriken:

Domain Relevance (DR): misst die Wahrscheinlichkeit eines Terms im Domänenkorpus im Vergleich zum Referenzkorpus. Je häufiger ein Begriff im Domänenkorpus im Vergleich zum allgemeinen Sprachgebrauch vorkommt, desto wahrscheinlicher handelt es sich um einen Fachbegriff.

Domain Consensus (DC): quantifiziert die Konsistenz eines Terms innerhalb des Domänenkorpus über verschiedene Dokumente hinweg. Die Berechnung basiert auf der Entropie der Verteilung des Begriffs über alle Dokumente. Begriffe, die in vielen Dokumenten gleichmäßig verteilt sind, erhalten einen höheren DC-Wert.

Die finale Entscheidung erfolgt über eine gewichtete Kombination dieser beiden Werte:

$$f(t) = \alpha \cdot DR(t) + (1 - \alpha) \cdot DC(t)$$

Ein Term wird ausgewählt, wenn $f(t) \geq \theta$. In der Praxis wurden verschiedene Kombinationen von α (z. B. 0.0, 0.3, 0.5, 0.7, 1.0) und θ (z. B. 0.4 bis 1.3) getestet.

Um die Qualität der extrahierten Bigrams zu erhöhen, wurde vor der Erstellung der Bigrams eine Filterung eingeführt. Satzzeichen wurden explizit entfernt, ebenso wie reine nicht-alphanumerische Tokens. Nach der Bigram-Erstellung wurden dann alle Kombinationen entfernt, die ein Stoppwort enthalten. Diese Vorgehensweise verhinderte, dass potenziell relevante Begriffe aufgrund einer zu frühen

Stoppwortentfernung verloren gingen, gleichzeitig aber auch triviale Phrasen wie "of the" oder "and a" berücksichtigt wurden.

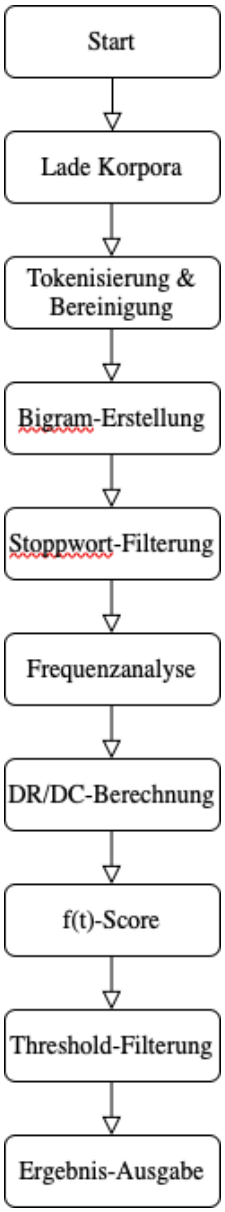
Pseudocode der Hauptkomponenten:

```
function extract_terms(acl, reuters):

dom_freqs = count_bigrams(acl, clean=True)

 ref_freqs = count_bigrams(reuters, clean=True)

dr = compute_domain_relevance(dom_freqs, ref_freqs)

dc = compute_domain_consensus(dom_freqs, acl_docs)

f_scores = combine_scores(dr, dc, alpha)

return filter_by_threshold(f_scores, theta)
```

Programmablauf

1. Laden des Domänen- und Referenzkorpus
2. Tokenisierung → Satzzeichen entfernen → Bigram-Erstellung
3. Entfernen von Bigrams mit Stoppwörtern
4. Frequenzberechnung für DR und Dokumentverteilung für DC
5. Score-Berechnung $f(t)$ mit gewähltem α
6. Filtern aller Begriffe mit $f(t) \geq \theta$
7. Ausgabe in TSV-Dateien und Vergleich mit Goldstandard

Flowchart



Jeder Schritt ist modular implementiert, wodurch das System flexibel erweitert oder angepasst werden kann.

Datensätze und Statistiken

- ACL-Dokumente: 59.501 Abstracts
- Reuters-Dokumente: 10.788 Texte
- Kandidatenbigrams im Domänenkorpus (nach Filterung): ≈ 920.000
- Kandidatenbigrams im Referenzkorpus (nach Filterung): ≈ 230.000

Hyperparameter-Kombinationen und Evaluation

Zur Evaluation wurden gezielt zehn Kombinationen von α und θ getestet, darunter auch Extremwerte wie α = 0.0 (nur DC) und α = 1.0 (nur DR). Die Ergebnisse zeigten deutlich: während ein niedriger Schwellenwert θ (z. B. 0.4) viele Begriffe durchlässt und somit einen hohen Recall erzeugt, steigt die Precision nur dann spürbar an, wenn θ ≥ 1.0 gewählt wird.

Beispielhafte Ergebnisse:

α	θ	Precision	Recall
0.0	1.0	0.0154	0.8403
0.3	1.3	0.0156	0.8396
0.5	1.1	0.0155	0.8403
0.7	1.2	0.0226	0.8254
1.0	0.9	0.0022	0.8635

Diese Ergebnisse belegen, dass insbesondere der θ-Wert entscheidend für das Precision-Recall-Verhältnis ist. Die Wahl von α beeinflusst eher die Gewichtung zwischen DR und DC, wobei höhere α-Werte tendenziell zu präziseren, aber weniger vollständigen Ergebnissen führen.

Quantitative und qualitative Bewertung

Die Ergebnisse wurden sowohl quantitativ anhand von Precision und Recall als auch qualitativ anhand einer manuellen Sichtung der extrahierten Begriffe bewertet. Die quantitativen Werte wurden durch den Vergleich mit der Datei gold_terminology.txt berechnet. Hierbei zeigte sich, dass sich Precision je nach α- und θ-Kombination im Bereich von 0.0022 bis 0.0226 bewegte, während Recall-Werte konstant hoch zwischen 0.825 und 0.916 lagen.

Diese Zahlen deuten darauf hin, dass das System viele relevante Begriffe erkennen konnte (hoher Recall), jedoch auch viele nicht-goldene Begriffe auswählte (niedrige Precision).

Qualitative Bewertung nach Punktzahl

Die Begriffe wurden zusätzlich qualitativ analysiert, indem sie anhand ihrer Entscheidungspunktzahl f(t) geordnet wurden. Drei Kategorien wurden dabei unterschieden:

- Obere Begriffe:
  Begriffe wie *“language model”*, *“word embeddings”*, *“machine translation”*, *“named entity”* oder *“statistical machine”* erzielten Punktzahlen über 9.0 (z. B. in

terms_alpha0.00_theta1.00.txt) und sind eindeutig domänenspezifisch und relevant.

- Mittlere Begriffe:
  Begriffe wie *"topic modeling"*, *"corpus annotation"*, *"dependency tree"* oder *"semantic similarity"* lagen typischerweise im Bereich von 5.5 bis 7.0 (je nach Konfiguration) und waren sinnvoll, wenn auch teilweise weniger konsistent über die Dokumente verteilt.
- Niedrige Begriffe (nahe θ):
  Begriffe wie *"important result"*, *"available data"*, *"different methods"* oder *„case study"* hatten Punktzahlen knapp über dem Schwellenwert θ und zeigten oft generische oder schwach domänenspezifische Inhalte.

Begriffe mit $f(t) < \theta$ wurden ausgeschlossen. Auffällig war, dass manche durchaus sinnvolle Begriffe dennoch unterhalb des Schwellenwerts lagen, was teils auf geringe Verteilung oder moderate DR-Werte zurückzuführen ist. Ebenso wurden durch die Stoppwort-Filterung erfolgreich viele triviale Ausdrücke wie „in a", „of the" oder „) ," entfernt, die in früheren Iterationen noch enthalten waren.

Vergleich mit und ohne Vorverarbeitung

Durch die Integration der Stoppwortentfernung und Satzzeichenbereinigung konnte eine qualitative Verbesserung der Ergebnisse erzielt werden. Begriffe wie „) ." oder „in a" wurden erfolgreich aus der Ergebnisliste ausgeschlossen. Gleichzeitig blieb die Recall-Rate stabil hoch. Ohne diese Vorverarbeitung hätten zahlreiche triviale oder grammatikalisch bedingte Bigramme die Precision deutlich verschlechtert.

Probleme und Lösungen bei der Implementierung

Die größte technische Herausforderung bestand in der effizienten Berechnung der DC-Metrik, weil diese die Entropie der Verteilung über zehntausende Dokumente umfasst. Ursprünglich wurde die Berechnung nur auf die häufigsten Begriffe beschränkt, um Ressourcen zu sparen. Durch die spätere Integration der Vorverarbeitung konnte die Anzahl der Kandidaten drastisch reduziert werden, sodass die komplette Analyse möglich wurde.

Ein weiteres Problem war die Erkennung und Entfernung von Satzzeichenkombinationen in Bigrammen, wie ") .", die trotz Tokenisierung teilweise verblieben. Die Einführung eines Filters für reine Satzzeichen und nicht-alphanumerische Tokens war notwendig, um diese zu eliminieren.

Was ich durch das Projekt gelernt habe

Ich habe gelernt, wie wichtig eine durchdachte Vorverarbeitung für NLP-Aufgaben ist, insbesondere bei statistischen Verfahren. Außerdem konnte ich mein Verständnis für

Metriken wie Entropie, DR und DC vertiefen und lernen, wie man diese systematisch miteinander kombiniert. Die Evaluation gegen einen Goldstandard hat mir zudem gezeigt, wie komplex es ist, Precision und Recall im Gleichgewicht zu halten und wie stark Hyperparameter wie α und θ das Ergebnis beeinflussen.

Was ich zukünftig anders machen würde

Bei einer Wiederholung würde ich früher eine POS-basierte Filterung implementieren (z. B. nur NOUN+NOUN-Kombinationen), um von vornherein generische Phrasen auszuschließen. Auch eine Lemmatisierung der Token könnte helfen, Dubletten wie "model training" und "models training" zusammenzufassen. Zudem würde ich die Evaluation um semantische Ähnlichkeitsmetriken erweitern, da der Goldstandard nicht alle korrekten Begriffe abdeckt.