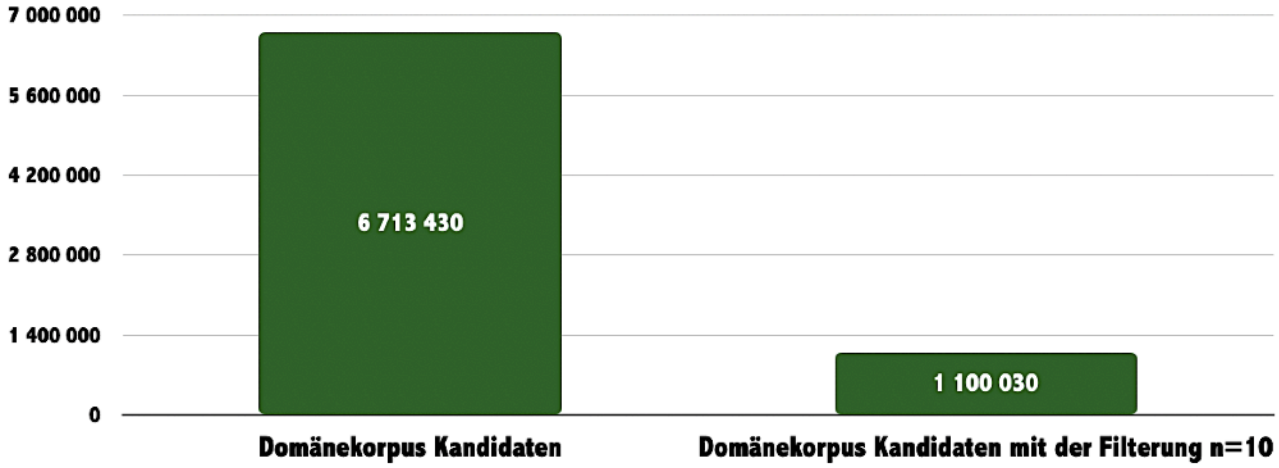
Computerlinguistische Techniken
Programmierprojekt
Bericht

Daryna Ivanova, 804197

Die Aufgabe des Projektes war eine statistische Methode zu implementieren, die die Extraktion der computerlinguistischer Terminologie aus den gegebenen wissenschaftlichen Arbeiten ermöglicht. Das Projekt besteht aus drei Hauptteilen: Kandidatenauswahl, Terminologieextraktion und Evaluierung.

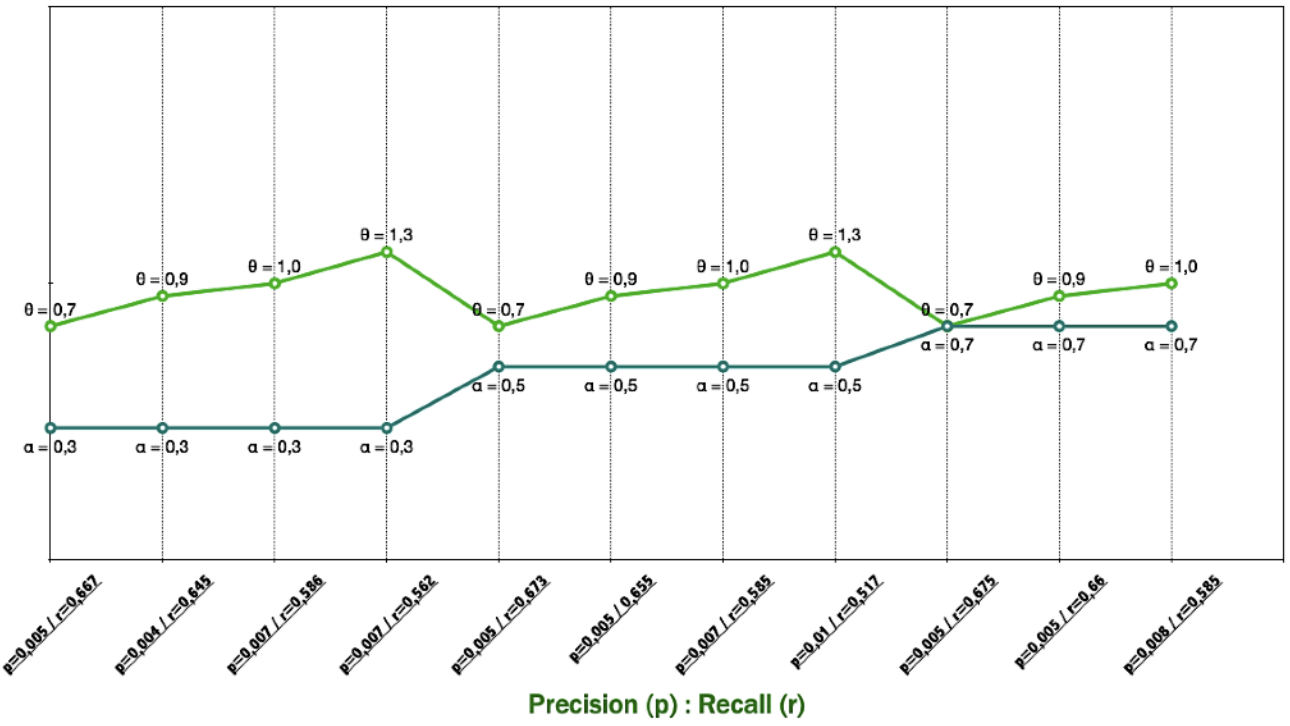
Die Gesamtanzahl der Texten im Domänenkorpus beträgt fast elf Tausend. Nachdem die Stoppwörter, Interpunktion und Zahlen aus dem Domänenkorpus entfernt wurden und alle Tokens in Kleinschreibung umgewandelt wurden, habe ich mehr als sechs Millionen Kandidaten bekommen. Zum Vergleich, im Referenzkorpus beträgt diese Zahl 861 386 Kandidaten. Ich habe einen zusätzlichen Filter eingereicht, wo man auf Wunsch eine beliebige ganze Zahl n eines Wortvorkommens im Text eingeben kann. Die Tokens, deren Vorkommen kleiner, als n ist, werden beim Kandidatenerstellung ignoriert. Die Gesamtzahl der Kandidaten des Domänenkorpuses nach der Filterung (n = 10) ergab nur eine Million:



Die Ausfilterung wurde nicht auf das Referenzkorpus angewendet, um vorzubeugen, dass die Unigrams, die für die computerlinguistische Terminologie relevant sind, entfernt werden. Auf jeden Fall, bewies sich die Ausfilterung mit n=10 als ineffizient. Der Grund dafür ist, dass ein Wortvorkommen innerhalb einer txt-Datei berechnet wird und es führt dazu, dass auch die relevante Wörter ausgesondert werden.

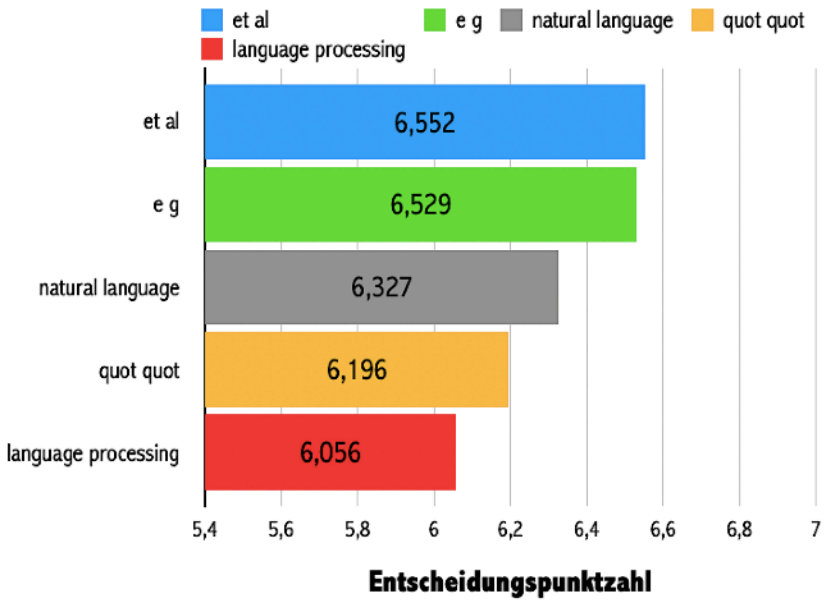
Ein weiteres und ein fixes Ausfilterungsverfahren ist Part-of-speech-Tagging (POS-Tagging). Die Idee ist, dass die Fachbegriffe nur aus der Kombination bestimmter Wortarten bestehen können müssen. Zum Beispiel, Adjektiv (JJ) + Nomen (NN): ‘activity creation’ ist akzeptabel, aber Verb (VB) + Verb (VB): ‘activate create’ ist nicht akzeptabel und daher wird aus der Kandidatenliste entfernt.

Das erste Schaubild zeigt die Precision/ Recall Scores für jede Alpha-Theta-Kombination vor der POS-Tagging Ausfilterung. Die Ergebnisse sind sehr geringwertig.

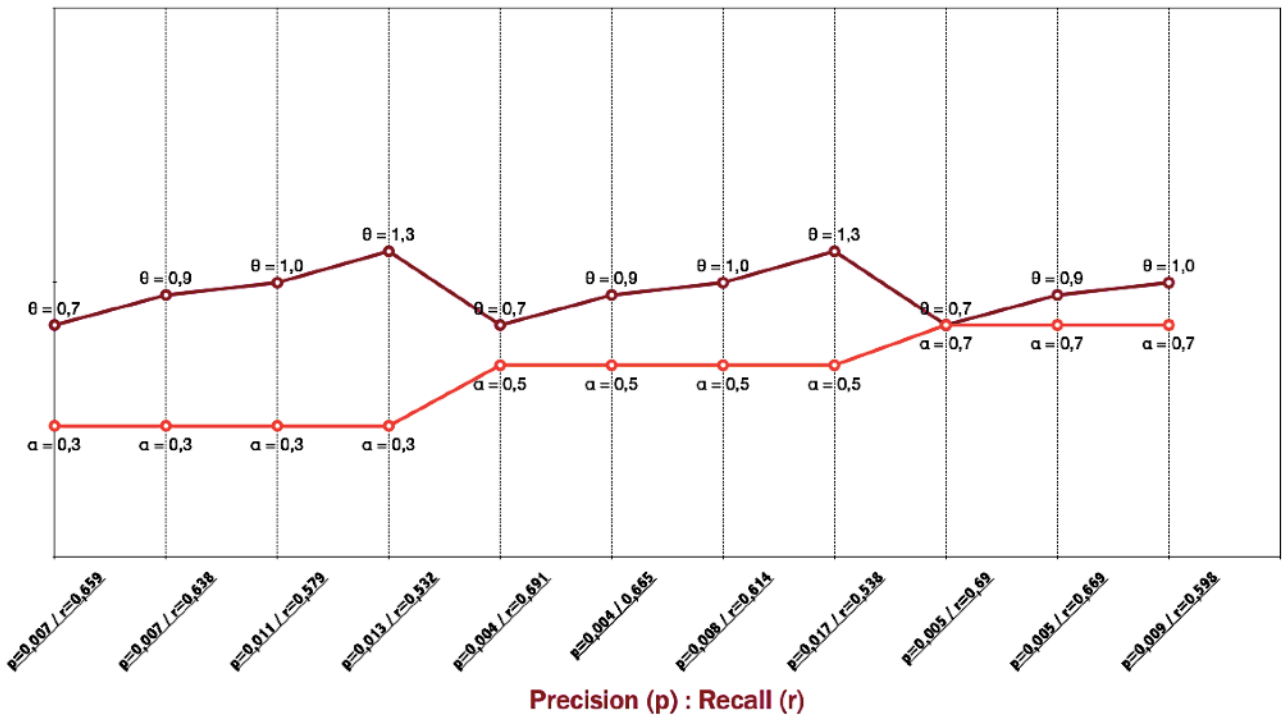


Bei der besten Kombination von alpha=0.5 und theta=1.3 sind 703 583 Begriffe in die Terminologie gelandet. Die Entscheidungspunktzahl der Begriffen liegt bei Die ersten fünf Begriffen, die höchste Entscheidungspunktzahl haben, sind im Bild unten demonstriert. Nach der Untersuchung der Endbegriffen habe ich festgestellt, dass viele Stoppwörter im Korpus beibehaltet wurden. Deswegen habe ich noch eine Erweiterung der Stoppwörterliste gemacht, die im Format einer txt-Datei ‘stopwords.txt’ gespeichert ist. Die Datei enthält außer semantisch leeren Einheiten auch einige Adjektive, Verben und Substantive, die sehr unwahrscheinlich für das Erstellen der Fachausdrücken verwendet werden können.

Die häufigst auftretende Begriffe für α=0.5, θ=1.3



Die zweite Abbildung (siehe unten) stellt die Precision/ Recall Scores nach der POS-Tagging. Anhand der Diagrammen ist es ersichtlich, dass Precision/ Recall Scores bevor und nach der POS-Tagging sich nicht auffallend unterscheiden lassen.



Generell lässt sich eine steigende Tendenz der Precision und eine fallende Tendenz des Recalls mit der Erhöhung einer Theta-Parameter erkennen. So, für alpha=0.95, theta=1.5 beträgt die Precision 66% und Recall weniger als 1% (0.0001).

Die endgültige Variante des Programms wählt zwei Millionen Kandidaten aus. Die Werte unterscheiden sich kaum von den vorherigen Ergebnissen. Die Abweichungen treten nur bei der Anzahl der resultierenden Begriffen auf, nähmlich jetzt sind sie auf bis zu einer Million beschränkt. Früher war diese Zahl circa zweimal Größer.

Die schlechte Leistung kann unterschiedliche Ursachen haben. Die erste ist die mangelnde Qualität von Domain Korpus. Eine andere mögliche Ursache ist, dass viele Begriffe im Goldstandard ein Kurzstrich zwischen einander enthalten und als ein Wort geschätzt sind. Mein Programm kann solche Begriffe nicht identifizieren. Ein weiteres Problem sind die Mehrwortbenennungen, die ganz oft im Korpus auftreten und in Terminologie gelandet sind.

Die Ergebnisse könnten mit den zahlreichen Methoden verbessert werden. Die erste bezieht sich auf Stemming und Lemmatisierug. So wie mein Versuch, die POS-Tags Kombinationen durchzusetzen, könnten die Lemmatisierung und Stemming wahrscheinlich effizienter helfen, die Daten zu normalisieren. Für die Anerkennung der Wörter, die einen Kurzstrich erhalten, könnte Chunk Parsing behilflich sein. Allein die oben genannten Annahmen werden vermutlich nicht für die Verbesserung der Precision und Recall Werten ausreichen. Sie können z.B. mit der Maximun-Likelihood-Methode oder mit dem maschinellen Lernen zusammengestellt werden. Dann, denke ich, werden die Ergebnisse überzeugender.