Matching Algorithmus Ziel: Start Ort aus VL -> ID aus GOV TODO? Muss Zerlege Eintrag in seine es immer Bestandteile. Komma sein? Gleiche jeden Bestandteil mit Namen im GOV ab. Alle Bestandteile Ja im GOV? min. 1 Bestandteil nicht im GOV gefunden Mindestens ein Bestandteil im GOV? Ankersuche alle Bestandteile nicht im GOV Bricht ab, sobald ein wurde. Sucht mit steigenden Kosten bis max_cost erreicht ist Suche für jeden Bestandteil mit Hilfe Kölner Phonetik nach Kandidaten, die gleich codiert werden. Anker für einen Bestandteil gefunden? Nein Suche für jeden Bestandteil zuerst in KREIS oder höher, dann in STADT und zuletzt im ganzen GOV nach Kandidaten. Suche für nicht gematchte Bestandteile in der Menge aller Orte, die von den bisher gematchten TODO, Geo-Einsatz aktuell nicht Bestandteilen erreicht werden können, nach implementiert Kandidaten. Bilde eine Liste aller Kombinationen aus allen Kandidaten für jeden Bestandteil. Ermittle für jede Kombination von Kandidatennamen die Liste aller Kombinationen aus allen IDs pro Kandidat. Betrachte eine Kombination aus IDs für Entweder in einer direkten Beziehung A Kandidatennamen als "possible match", wenn -> B oder aber über einen gemeinsamen die Beziehung im GOV existiert. Parent A -> C -> B Lösung für Kombinationen mit Gib für jede valide Kombination von IDs einen TODO, Treffer aus mit GOV IDs, textual IDs, type IDs aktuell nicht validen IDs, aber Beziehung nicht implementiert und type names pro Bestandteil. im GOV abgebildet Ende