Woche 07 Praktische Anwendungsbeispiele: K-Nearest Neighbours

Skript

Erarbeitet von
Anna Stein

Lernziele 1
Inhalt 1
Einstieg 1
TIMBL 2
Suchvorschläge mit KNN 3
Take-Home Message 4
Quellen 4
Weiterführendes Material 5
Disclaimer 5

Lernziele

- Konkrete Anwendungsbeispiele für KNN benennen

Inhalt

Einstieg

Jetzt, da wir wissen, wie der K-Nearest-Neighbours-Algorithmus, oder auch KNN, funktioniert, können wir uns zwei konkrete Anwendungsbeispiele angucken.

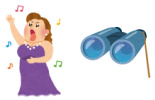
TIMBL

Als Erstes werfen wir einen Blick in die Linguistik. Dort werden verschiedene Implementationen von KNN für die Forschung genutzt, die sich hauptsächlich in ihrem Namen und in der Programmiersprache unterscheiden. Darunter ist auch der *„Tilburg memory-based learner“*, TIMBL.

Quellen [1, 2]

Die Linguistin Sabine Arndt-Lappe hat TIMBL dazu verwendet, die Betonung von zusammengesetzten Nomen im Englischen zu untersuchen.

Einblendung Nomen mit Übersetzungen



"Òpera-glasses"
Opernfernglas

"Steel-brídge"
Stahlbrücke

Dort gibt es nämlich Nomen wie zum Beispiel *„Òpera-glasses“*, wo die Betonung auf den linken Teil des Nomens fällt (auch „left stress“ genannt), aber auch Nomen wie *„Steel-brìdge“*, wo die Betonung auf den rechten Teil fällt (das nennt man „right stress“). Arndt-Lappe hat sich die Frage gestellt, wie viele Informationen gebraucht werden, um die Betonung eines zusammengesetzten Wortes zu bestimmen.

Dazu hat sie verschiedene KNN-Modelle trainiert, die alle unterschiedliche Informationen zur Verfügung hatten, um die Nomen in *„right stress“* oder *„left stress“* zu kategorisieren.

Die Art der Information wurde in abstrakte und in nicht-abstrakte Informationen geteilt. Zu abstrakten Eigenschaften gehören Information wie, ob das Nomen ein Subjekt oder ein Objekt ist, und die nicht-abstrakten Eigenschaften sind Informationen, die direkt zum Nomen gehören, wie orthographische Repräsentation der beiden Bestandteile des Nomens. Die orthographische Repräsentation ist einfach nur, wie der Teil geschrieben wird.

Vorherige Forschung zum Thema Betonung zusammengesetzter Nomen im Englischen ließ darauf schließen, dass die Modelle mit den abstrakteren Informationen besser sein sollten. Diese Experimente, die nicht KNN genutzt haben, konnten die Betonung von Nomen nämlich mit abstrakten Informationen gut klassifizieren. Allerdings wurden diese Erwartungen von den Ergebnissen dieser Studie nicht erfüllt.

Ein Modell, das die orthographische Repräsentation des linken und des rechten Teils des Nomens zur Verfügung hatte, war das beste Modell. Auch nach Hinzufügen abstrakter

Informationen wurde dieses Modell nicht besser. Im Endeffekt wurden also gar keine abstrakten Informationen gebraucht. Daraus kann man interessante Schlüsse ziehen, wie Menschen Sprache verarbeiten, zum Beispiel welche Informationen wir für welche Prozesse der Sprachverarbeitung wirklich benötigen.

Suchvorschläge mit KNN

Für das zweite Beispiel gucken wir uns einen kommerziellen Nutzen von KNN an, wo ein Suchsystem implementiert werden sollte.

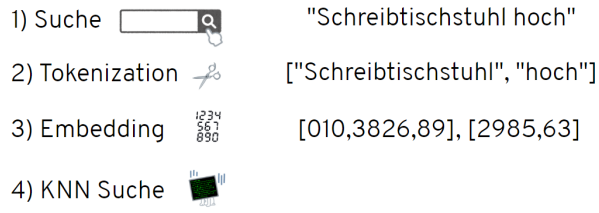
Quellen [3, 4]

Bisher hatten die Standorte des Pharmazeutika-Unternehmens Novartis AG separate, interne Systeme zum Kauf von Laborprodukten. Dadurch mussten die Produkte immer einzeln von den Mitarbeiter*innen in der Internetseite der Anbieter*innen verglichen werden, um den besten Preis zu finden. Dies führte zu schlechterer Organisation und Verzögerungen. Das Ziel für ein neues System war es also, einen Katalog mit vielen verschiedenen Angeboten zu haben, damit dieser von den Mitarbeiter*innen durchsucht werden kann.

Als Erstes brauchte man dazu also einen Katalog mit durchsuchbaren Einträgen. Das waren in diesem Fall die Laborprodukte von verschiedenen Händler*innen. Aus den Produktnamen wurden als erstes *„word embeddings“* gemacht. Das sind numerische Repräsentationen von Wörtern, die unter anderem bestimmte Labels darstellen. Beispielsweise wäre in dem Vektor, der das Produkt „Perfekter Bürostuhl“ repräsentiert, das Label „Möbelstück“ enthalten. Wenn ihr wissen wollt, wie das genau funktioniert, guckt euch gerne den Link dazu in den weiterführenden Materialien an.

Wenn also jemand etwas in das Suchfeld eingibt, passiert Folgendes:

Einblendung Suchbeispiel



Als Erstes wird der Text in einzelne Wörter aufgeteilt, das nennt sich *„tokenization“*. Dann werden die einzelnen Wörter auch in *word embeddings* umgewandelt. Für die *embeddings* von der Suche werden dann mithilfe von KNN die ähnlichsten Produkte herausgesucht.

Anders als bei den Anwendungsbeispielen, die wir bisher gesehen haben, werden hier also nicht Klassifizierungen ausgegeben, sondern einfach nur die *nearest neighbours*. Es wird sozusagen der letzte Schritt der Klassifizierung weggelassen.

Einblendungen Keyword Ergebnisse, KNN Ergebnisse

Keyword Ergebnisse

Kategorie	Name
"Möbelstück"	"P U Leder Schreibtischstuhl"
"Möbelstück"	"Pulse Schreibtischstuhl"
"Möbelstück"	"Mid Back Schreibtischstuhl"
"Möbelstück"	"High Back Schreibtischstuhl"
"Möbelstück"	"Serta Schreibtischstuhl"

KNN Ergebnisse

Kategorie	Name
"Möbelstück"	"High back Schreibtischstuhl"
"Möbelstück"	"Pulse Schreibtischstuhl"
"Möbelstück"	"Vita Bürostuhl"
"Möbelstück"	"Beginnings Schreibtisch"
"Möbelstück"	"Palladia Schreibtisch"

Die Firma wollte ihre Suche aber noch weiter verbessern und hat daher zusätzlich zu KNN noch eine „*Keyword* Suche“ implementiert. Bei einer *Keyword* Suche werden einfach die Produkte ausgegeben, die das eingegebene Wort enthalten. In diesen beiden Tabellen kann man die Ergebnisse von KNN und der *Keyword* Suche für den Suchbegriff „Schreibtischstuhl“ sehen. In der oberen Tabelle sind nur Ergebnisse enthalten, wo das Wort auch tatsächlich enthalten ist. In der Unteren, wo die Ergebnisse von KNN zu sehen sind, sind auch andere Dinge wie Schreibtische enthalten. Wie hier zu sehen ist, bekommt man unterschiedliche Suchergebnisse von den beiden Suchansätzen heraus.

Für die kombinierte Suche werden zuerst die passenden Produkte nach *Keyword* Suche herausgesucht, und dann die Produkte, die laut KNN am passendsten sind. Diese Auswahl wird von einem anderen Algorithmus nach Relevanz sortiert und dann in dieser Reihenfolge präsentiert.

Take-Home Message

Wir kennen jetzt also zwei Beispiele, wie man den KNN-Algorithmus anwenden kann, sowohl in der Wissenschaft, als auch für Vorschläge in einem Suchsystem.

Quellen

Quelle [1] TiMBL 6.4 (c) CLS/ILK/CLiPS 1998 - 2022, Centre for Language Studies, Radboud University Nijmegen, Induction of Linguistic Knowledge Research Group, Tilburg University and Centre for Dutch Language and Speech, University of Antwerp

Quelle [2] Arndt-Lappe, S. (2011). Towards an exemplar-based model of stress in English noun-noun compounds. *1. Journal of linguistics*, 47(3), 549-585.

Quelle [3] https://aws.amazon.com/de/blogs/industries/how-novartis-brought-smart-into-smart-procurement-with-aws-machine-learning/

Quelle [4] https://aws.amazon.com/de/blogs/industries/novartis-ag-uses-amazon-elasticsearch-k-nearest-neighbor-knn-and-amazon-sagemaker-to-power-search-and-recommendation/

Weiterführendes Material

Spotify Music Classifier mit KNN.
Code. https://github.com/sameehaafr/spotify_music_sorte
Medium Artikel. https://towardsdatascience.com/ml-step-by-step-using-knn-algorithm-to-classify-spotify-songs-into-playlists-8c7892428371

Open Source software.
https://itsfoss.com/what-is-foss/

Netflix Vorschläge zum Selbermachen.
https://blog.jaysinha.me/train-your-first-knn-model-for-collaborative-filtering/

Disclaimer

Transkript zu dem Video „Woche 07 Praktische Anwendungsbeispiele: K-Nearest Neighbours“, Anna Stein.