Topic Drift

Melih Demirel

1953139

Introduction

Het doel van dit project is het ontwikkelen van een programma dat in staat is om onderwerpen te identificeren op basis van publicatietitels. We zullen dit programma vervolgens gebruiken om de evolutie van onderwerpen binnen het vakgebied van Data Mining en/of Database Systemen in de loop der tijd te onderzoeken. Door de titels van publicaties te analyseren, streven we ernaar inzicht te krijgen in de opkomende trends, verschuivingen en ontwikkelingen binnen deze belangrijke domeinen van de informatica. In dit rapport presenteren we de methodologie, resultaten en conclusies van onze analyse.

Preprocessing

Voordat we begonnen met het implementeren van methoden om onderwerpen te detecteren, hebben we eerst onze data voorbewerkt. We beschikken ove reen lijst met titels die veel woorden of symbolen bevatten die niet relevant zijn voor onze implementatie in de volgende fasen. Het voorbewerken van onze gegevens vergemakkelijkt het later detecteren van overeenkomsten. De voorbewerking omvat het verwijderen van leestekens, het verkleinen van letters en het toepassen van een stemmer of lemmatizer. We hebben beide methodes geprobeerd, en elk heeft zijn voor- en nadelen.

Stemming:

Voordelen:

- Sneller: stemming eenvoudigweg de woorden inkort tot hun stam.
- Minder complexiteit: minder rekenintensief vergeleken met lemmatisering

Nadelen:

- Minder accuraat: stemming kort woorden in zonder rekening te houden met de context of betekenis, dit leidt tot minder nauwkeurige resultaten

Lemmatisering:

Voordelen:

- Nauwkeuriger: houdt rekening met context en betekenis van woorden, levert dus meer accurate resultaten.
- Betere interpretatie: behoudt basisvorm van woorden -> gemakkelijker te interpreteren

Nadelen:

- Langzamer: complexere verwerking -> computationally intersiever dan stemming.

Aangezien het voor onze analyse van belang is om rekening the houden met de context en betekenis van woorden om nauwkeurigere resultaten te verkrijgen, hebben we besloten om lemmatisering te gebruiken. Deze van library NLTK. Het nadeel van deze keuze is dat het meer tijd vergt en computatief intensiever is. Echter, de nauwkeurigheid en interpretatievoordelen wogen zwaarder voor ons specifieke doel.

Distance Metric

We hebben ervoor gekozen om TF-IDF (Term Frequentie-Inverse Document Frequentie) als onze afstandsmaat te definiëren. Deze keuze is ingegeven door verschillende overwegingen die een cruciale rol spelen in ons streven naar effectieve tekstrepresentatie en clustering van publicatietitels.

Ten eerste biedt TF-IDF een gestandaardiseerde manier om de relevantie van woorden in documenten te beoordelen. Door rekening te houden met zowel de frequentie van woorden in een specifiek document als de mate waarin deze woorden uniek zijn voor dat document in vergelijking met het hele corpus, krijgen we een metriek die de nadruk legt op de onderscheidende kenmerken van elk document.

Een alternatieve benadering die we hebben onderzocht, is het gebruik van CountVectorizer. Echter, na een initiële test met een klein dataset waren we niet tevreden met de resultaten. CountVectorizer houdt alleen rekening met de frequentie van woorden, zonder rekening te houden met hun relevantie in het bredere corpus. Dit kan leiden tot een minder nauwkeurige representatie van de documenten, vooral in het geval van teksten met variabele lengtes.

Kortom, de keuze voor TF-IDF als onze afstandsmaat is gebaseerd op de wens om zowel de frequentie als de relevantie van woorden in onze documenten te waarderen, met het oog op nauwkeurige tekstrepresentatie en effectieve clustering.

Cluster

In onze zoektocht naar een geschikt clusteringalgoritme hebben we verschillende opties geëvalueerd, waaronder K-means, MiniBatchKMeans, Agglomerative, Birch, en DBSCAN van Sckikit-learn. Na het experimenteren bleken K-means en MiniBatchKMeans bijzonder effectief, vooral gezien de omvang van onze dataset. K-means werd verkozen boven Birch vanwege de snellere uitvoering en schaalbaarheid, met name bij hoger-dimensionale data. De uiteindelijke keuze tussen K-means en MiniBatchKMeans werd gemaakt op basis van documentatie, waaruit bleek dat MiniBatchKMeans sneller convergeert, met slechts een kleine concessie aan de resultaatkwaliteit. In de praktijk hebben we waargenomen dat MiniBatchKMeans niet alleen sneller was, maar ook iets betere clusteringresultaten opleverde, waardoor het de meest geschikte keuze werd voor onze opdracht. Agglomerative en DBSCAN werden minder geschikt bevonden vanwege hun aanzienlijk langere uitvoeringstijden.

Other methods

We hebben ook geëxperimenteerd met andere methoden zoals LDA (Latent Dirichlet Allocation), Linkage Matrix, F-clustering en Non-Negative Matrix Factorization. Echter, bij deze methoden hebben we niet het gewenste niveau van succes bereikt, en om efficiëntie en effectiviteit te waarborgen, hebben we besloten ze niet verder te verkennen. Onze focus verschoof naar clusteringalgoritmen zoals K-means en MiniBatchKMeans, die beter aansloten bij onze specifieke dataset en clusteringbehoeften. Deze beslissing werd genomen op basis van empirische resultaten en praktische

overwegingen, waarbij we prioriteit gaven aan algoritmen die betrouwbaar presteerden en snel convergeerden binnen de context van ons onderzoek.

Implementation

Vanwege tijdgebrek en het feit dat ik alleen aan deze taak werkte, kon ik niet voldoende tijd besteden aan het implementeren van de visualisatie van de clusters. Als ik meer tijd had gehad, ben ik er vrij zeker van dat ik dit succesvol had kunnen implementeren. Het programma draait momenteel op het bestand **"data_mining_publications.txt",** en het is eenvoudig aan te passen voor andere invoerbestanden. Bovendien heb ik parameters zoals time-interval, overlap en n-grams opgenomen, omdat een onderwerp niet enkel uit 1 woord bestaat maar ook meerdere, zoals "data mining" of "graph neural network". Deze parameters zijn flexibel en kunnen binnen het programma worden aangepast. Running time was een paar seconden. Alle testen zijn gedaan met Jupyter Notebook. Binnen deze file kunt u alles zien wat ik heb geprobeerd.

Results

Nu bespreken we de resultaten van onze clusters. De output is te vinden onder sectie Output.

1988-2001:

In de eerste periode blijkt een overheersende focus op rules, associations, en algorithms, wat wijst op een nadruk op rule-based systems en algoritmische benaderingen in de vroege jaren. Tegelijkertijd zien we de opkomst van het belang van "data mining" als een cruciaal onderwerp gedurende deze periode. Er is ook aandacht voor knowledge discovery, databases, en rough sets, wat wijst op een gelijktijdige interesse in methodologieën voor knowledge discovery.

1998-2011:

In deze periode verschuift de focus naar meer geavanceerde onderwerpen zoals machine learning, social networks, en Bayesian networks, wat wijst op een diversificatie van onderwerpen met een grotere nadruk op machine learning en netwerkgerelateerde thema's. Er is ook voortdurende interesse in clusteringmethodologieën en een diepere verkenning van patronen binnen data mining.

2008-2021:

Deze fase handhaaft de focus op "data" en "mining," wat wijst op een consistente nadruk op data mining door de jaren heen. Er is een uitbreiding naar geavanceerde machine learning technieken zoals deep learning, multitask learning, en active learning. Bovendien zien we een opkomst van complexe netwerkgerelateerde onderwerpen zoals social network analysis, anomaly detection, en neural networks, wat duidt op een verschuiving naar complexere netwerkgerelateerde onderwerpen. Ten slotte zien we de opkomst van graph neural networks, graph-based classification, en feature selection, wat wijst op een groeiende interesse in graph-related data mining methodologies.

2018-2031:

In de meest recente periode blijft de focus centraal liggen op "data," wat wijst op de voortdurende relevantie van data mining themes. Er is verdere uitbreiding naar geavanceerde machine learning

paradigma's zoals deep learning, representation learning, en reinforcement learning. Deze bevindingen bieden waardevolle inzichten in het veranderende landschap van data mining topics in de loop van de tijd, waarbij elke cluster verschillende thema's en methodologieën vastlegt.

Conclusion

De resultaten ogen veelbelovend. We konden daadwerkelijk onderwerpen afleiden uit de clustering. Indien er meer tijd beschikbaar was geweest voor het detecteren van betere clusters, zou ik extra methoden hebben verkend om de data te bestuderen en niet alleen stopwoorden, maar ook andere woorden te verwijderen die niet gerelateerd zijn aan onze bestudeerde gegevens. Misschien zou ik ook verschillende benaderingen voor het detecteren van onderwerpen proberen, naast het gebruik van specifieke clusteringmethoden.

Output

Topic Drift:

Time interval: 10 years.

Time overlap: 2 years.

Ngrams set to: (1, 3).

Clustering 515 titles in range: 1968-1981

Cluster 1: 51.65%

- 0.05 database
- 0.03 system
- 0.02 database system
- 0.01 distributed
- 0.01 file
- 0.01 distributed database
- 0.01 relational
- 0.01 relational database
- 0.01 access
- 0.01 query
- 0.01 database machine
- 0.01 information
- 0.01 dbms

Clustering 1881 titles in range: 1978-1991

Cluster 1: 41.89%

0.02 data

0.01 management

0.01 system

0.01 query

Cluster 2: 27.54%

0.08 database

0.04 system

0.03 database system

0.02 distributed

0.02 distributed database

0.02 design

0.01 relational database

0.01 relational

0.01 objectoriented

0.01 database design

0.01 objectoriented database

0.01 language

0.01 deductive

0.01 deductive database

Clustering 3284 titles in range: 1988-2001

Cluster 1: 49.27%

0.01 system

Cluster 2: 20.95%

0.09 database

0.03 database system

0.03 system

0.02 objectoriented

0.02 objectoriented database

0.01 deductive

0.01 relational database

0.01 relational

0.01 deductive database

0.01 distributed

0.01 objectoriented database system

0.01 management

Cluster 3: 14.22%

0.11 data

0.02 data mining

0.02 mining

0.02 data model

0.02 model

0.02 management

0.02 warehouse

0.02 data warehouse

0.01 data management

0.01 system

0.01 warehousing

0.01 data warehousing

Cluster 4: 10.93%

0.12 query

0.04 processing

0.03 optimization

0.03 query processing

0.02 query optimization

0.01 using

0.01 query language

0.01 database

0.01 language

Cluster 5: 4.63%

0.09 transaction

0.08 performance

0.03 performance evaluation

0.03 transaction processing

0.03 evaluation

0.02 processing

0.02 distributed

0.02 concurrency

0.02 control

0.02 model

0.02 transaction management

0.01 transaction model

0.01 management

0.01 performance analysis

0.01 database

0.01 analysis

0.01 performance study

0.01 algorithm

0.01 system

0.01 study

0.01 concurrency control

0.01 high performance

0.01 high

0.01 machine

Clustering 5998 titles in range: 1998-2011

Cluster 1: 53.17%

Cluster 2: 17.21%

0.09 data

0.02 data management

0.02 management

0.01 warehouse

0.01 data warehouse

0.01 mining

Cluster 3: 14.15%

0.10 query

0.04 processing

0.03 query processing

0.02 optimization

0.02 efficient

0.01 query optimization

0.01 xml

0.01 continuous

0.01 topk

Cluster 4: 12.42%

0.11 database

0.03 database system

0.03 system

0.02 relational database

0.02 relational

0.01 search

Cluster 5: 3.05%

0.14 stream

0.12 data stream

0.07 data

0.02 processing

0.02 distributed

0.02 stream processing

0.02 distributed data

0.02 distributed data stream

0.01 query

0.01 continuous

0.01 clustering

0.01 algorithm

0.01 mining

0.01 framework

0.01 shedding

0.01 monitoring

0.01 query data stream

Clustering 9558 titles in range: 2008-2021

Cluster 1: 65.86%

0.01 database

0.01 graph

Cluster 2: 31.12%

0.05 data

0.04 query

0.01 processing

0.01 big

Cluster 3: 2.00%

0.15 topk

0.05 topk query

0.04 query

0.02 uncertain

0.02 topk query processing

0.02 search

0.02 processing

0.02 efficient topk

0.01 efficient

0.01 oracle

0.01 keyword

0.01 database

0.01 graph

0.01 large

0.01 reverse topk

0.01 query processing

0.01 spatial

0.01 similarity

0.01 reverse topk query

0.01 join

0.01 data

0.01 uncertain data

0.01 finding topk

0.01 reverse

Clustering 5386 titles in range: 2018-2031

Cluster 2: 15.76%

0.09 data

0.01 big

0.01 analytics

0.01 data management

0.01 management

0.01 data analytics

0.01 big data

Cluster 3: 9.49%

0.07 system

0.04 learning

0.03 machine learning

0.03 machine

0.03 database

0.02 database system

0.01 data

0.01 reinforcement

0.01 reinforcement learning

0.01 deep

Cluster 5: 2.80%

0.11 stream

0.05 data stream

0.04 stream processing

0.03 join

0.03 distributed

0.03 processing

0.03 data

0.02 similarity join

0.02 similarity

0.02 distributed stream

0.01 set similarity join

0.01 detection

0.01 stream join

0.01 set similarity

0.01 event

0.01 event stream

0.01 processing system