Vrije Universiteit Brussel

Faculteit Wetenschappen
Departement Computerwetenschappen

Gevoelsanalyse in het Nederlands

Yannick Merckx

Bachelorproef

Rolnummer: 500294

Promotor: Yann-Michaël De Hauwere
Begeleiders: Maarten Deville
Peter Vrancx

Juni 2015

Samenvatting

Gevoelsanalyse is een populaire gegeven binnen de Machine Learning. Over gevoelsanalyse op de Engelse taal vindt men voldoende naslag werk (http://nlp.stanford.edu/sentiment/), maar in het Nederlands is dit eerder beperkt. Dit heeft deels te maken met het kleine bereik van de Nederlandse taal wat het onderzoek ernaar al minder interessant maakt en als men er onderzoek naar zou doen, dit vaak door een bedrijf wordt uitgevoerd waarbij de resultaten onder bedrijfsgeheim vallen. Daarom concentreren wee bij deze bachelorproef op gevoelsanalyse op de Nederlandse taal. We experimenteren met enkele algemene technieken uit de Machine Learning die ons een degelijke gevoelsanalyse kunnen opleveren en proberen ten slotte een besluit te trekken of we met enkele algemene technieken effectief een acceptabele gevoelsanalyse op het Nederlands kunnen uitvoeren.

Dank woord

Ik wil graag mijn begeleiders, Maarten Deville en Peter Vranckx, van harte bedanken voor de grote steun en inzet gedurende het hele jaar. Ze waren altijd beschikbaar gedurende het hele jaar om op al mijn vragen een snel antwoord te geven. Als laatste wens ik mijn dank uit te drukken aan mijn promotor, Yann-Michaël De Hauwere. Bij de korte evaluaties was hij altijd aanwezig en stond altijd bij voor raad en daad.

Inhoud

Hoofdstuk 1

Introductie

Vandaag de dag is informatie nog nooit zo belangrijk geweest. Iedere dag komt er ook enorm veel informatie bij. Kijk maar naar social media, waar iedere dag duizende gebruikers hun mening uiten over alledaags dingen. Het is dan ook zeer interessant om die data te analyseren en daar een zekere kennis uit te vergaren. Vanwege de grote hoeveelheid aan data is het onmogelijk om een programma manueel te schrijven, dat enige kennis uit die data kan halen. Machine learning biedt hier de oplossing. Dit is een onderzoeksdomein binnen de Artificiële Intelligentie dat zich toespitst op zelflerende algoritmes. In deze Bachelorproef onderzoeken we of we met enkele technieken uit de machine learning een goede analyse kunnen uitvoeren op tekst. Concreter wordt er gefocust op de gevoelsanalyse van Nederlandse tekst, waarbij een programma beslist of de gegeven tekst een positief of negatief gevoel uitdrukt. Deze thesis is opgesplitst in drie grote delen namelijke de achtergrondinformatie, het experiment en de conclusie. Als achtergrondinformatie worden alle technieken die tijdens die het experiment worden toegepast besproken. Vervolgens bespreken we het experiment en de resultaten. Als laatste vormen we een conclusie over de mogelijkheid om met enkele technieken uit de machine learning een geslaagde gevoelsanalyse kunnen uitvoeren op Nederlandse tekst.

Hoofdstuk 2

Achtergrondinformatie

In dit hoofdstuk bespreken we de technieken, classifiers en de valkuilen voor het onderzoek, waar zeker rekening mee moet gehouden worden. Zoals eerder vermeld is het doel van deze bachelorproef om met behulp van enkele gekende technieken uit de machine learning een gevoelsanalyse uit te voeren op Nederlandse tekst. En vervolgens analyseren hoe deze prestaties zijn.

Voor het onderzoek gebruiken we supervised learning. Hierbij weten we al onze oplossingen van onze dataset. De dataset die we meegeven aan ons programma bevat alle oplossingen over hoe welke tekst positief is en welke negatief. Het programma moet dan aan de hand van de tekst en de oplossing verbanden proberen te leggen, zodanig dat wanneer het algoritme een onbekende tekst binnen krijgt deze kan toewijzen naar het concept positief of negatief.

Nu kunnen we het programma een handje helpen door, voor dat men de dataset meegeeft aan het algoritme, de dataset al eens voor te verwerken of Pre-processen. Hoe we dit juist kunnen doen wordt in de volgenden sectie besproken.

2.1 Technieken voor Pre-Processing

Het voor verwerken of pre-processen van een dataset kan op verschillend manieren gebeuren. We willen classificeren voor het algoritme vergemakkelijken. Hoe we dit gaan doen wordt in deze sectie uitgelegd.

2.1.1 Bag of Words

Bag of Words is de eenvoudigste methode die er is. Ieder document wordt beschouwd als een zak met woorden, waarbij de woorden in het document de kenmerken of de features van het document voorstellen. Bij de volgende technieken wordt Bag of Words vaak als startpunt genomen en wordt er vervolgens nog andere optimalisatie op toegepast.

2.1.2 Verwijderen van stopwoorden

Wat we vaak zien in het Nederlands,maar ook in taal algemeen, is dat er veel stopwoorden gebruiken. Stopwoorden als k̈loptën ëigenlijkz̈eggen niet veel over of de tekst nu positief of negatief is. Als het niet bijdraagt voor het algoritme kunnen stopwoorden beschouwen als noise in de dataset en verwijdert men ze beter. Het verwijderen van stopwoorden en leestekens is ook een manier van pre-processing.

2.1.3 Bigran Collocaties

Bigrams Collocaties is een techniek waarbij men op zoek gaat naar paren woorden die een hoge waarschijnlijkheid hebben om samen voor te komen en een extra bron van informatie kunnen vormen voor de gevoelsanalyse. De bepaling van de significantie van een paar woorden is gebaseerd op de op de interne frequentie van de woorden de frequentie van de combinatie van de woorden in de tekst. Als men een overzicht krijgt over de frequentie kan men deze associeren met een score aan de hand van een scorefunctie, zoals bijvoorbeeld de Chi-kwadraattoets. De Chi-kwadraattoets is een statistische toets die het mogelijk maakt om de onafhankelijkheid tussen waarnemingen te onderzoeken. Bij Bigram Collocaties onderzoekt men via de Chi-kwadraattoets de afhankelijkheid tussen twee woorden. Hoe grotere de afhankelijkheid, hoe hoger de score.

Chi-Kwadraattoets

De Chi-Kwadraattoets is een techniek uit de statistiek die gebruikt kan worden als een onafhankelijkheidstoets voor waarnemingen. De reden waarom we deze toets ondermeer voor Bigram collocatie gebruiken is dat het parametervrije toets is. Hiermee bedoeld men dat er voor de chi-kwadraattoets bij de start van de toets geen aannames over de populatie of gemiddelde verwacht. In deze sectie leggen we aan de hand van een voorbeeld uit hoe de chi-kwadraattoets juist deze afhankelijkheid bepaald.

Neem als voorbeeld het bigram (heel , goed)∴ Zoals bij iedere statistische test neemt men eerst een nulhypothese aan. Voor de chi-kwadraattoets is dit ook het geval. De toets neemt als nulhypothese aan dat beide woorden onafhankelijk van elkaar zijn. Men vergelijkt de waargenomen frequenties van de woorden met de verwachte frequenties wanneer de woorden onafhankelijk zouden zijn. Als deze waarden te veel verschillen kan men de nulhypothese verwerpen en de alternative hypothese aannemen, namelijk dat de woorden afhankelijk zijn van elkaar.

De toetsingsgrootheid om de geobserveerde frequentie te toetsen met de verwachte frequenties volgt volgende formule:

$$\chi^2 = \frac{1}{d} \sum_{i,j} \frac{(O_i j - E_i j)^2}{E_i j}$$

Waarbij $O_i j$ het aantal keer dat het paar (i, j) voorkomt. $E_i j$ stelt de voorspelde waarden voor als de woorden onafhankelijk moesten voorkomen
$E_i j$ wordt bepaald door volgende formule:

$$E_i j^2 = \frac{O_i * O_* j}{n}$$

Hoofdstuk 3

Het Experiment

3.1 Proefopstelling

3.2 Werkwijze

krijgen bij meer features.

3.3 Hypothese

3.4 Resultaten

Hoofdstuk 4

Conclusie

In deze voorbereiding hebben we omschreven wat Machine Learning juist omvat. Namelijk het onderzoeken en ontwikkeling van zelflerende algoritmes, die hoofdzakelijk uit drie stappen bestaan, namelijk data verzamelen, verwerken en analyseren. Naargelang het soort data en wat deze weergeeft bestaan er binnen het domein van Machine Learning verschillende technieken met hun specifieke eigenschappen en voorbeelden. Voor situaties waarin we op voorhand over voldoende data beschikken en we duidelijk weten wat deze data betekend, bespraken we in sectie 2.2 verschillende supervised learning technieken die in staat zijn om een hypothese te formuleren op basis van de gegeven data. We maakte een duidelijk onderscheid welke problemen er zich kunnen voordoen zoals een classificatie probleem versus een regressie probleem en hoe men deze moet oplossen.
We hebben gezien dat een classificatie probleem zich onderscheidt van een regressie probleem door dat de output van de hypothese zich beperkt tot een kleine set van mogelijkheden, wat bij een regressie probleem een hele reeks van mogelijkheden is. Vervolgens hebben we een specifieke techniek besproken, namelijk de vector space methode. Een methode die van toepassing is bij text mining. Deze methode kan men op verschillende manieren verfijnen. Zo raadde we aan in sectie 3.1 om document pre-processing toe te passen op de dataset voor de verwerking van de data. Verder werd er ook aangeraden in sectie 3.2.1 om de standaard vector space methode te optimaliseren met technieken zoals Latent Semantic Analysis (LSA) en Term weighting. Ten slotte hebben we een proefopstelling opgesteld waarbij de techniek LSA wordt toepast en de efficiëntie en werking van Latent Semantic Analysis nogmaals wordt bevestigd.

Literatuur

Decomposing signals in components (matrix factorization problems). (z. j.). http://scikit-learn.org/stable/modules/decomposition.html. (Accessed: 2014-30-11)

Landauer, T. K., Foltz, P. W. & Laham, D. (1998). An introduction to latent semantic analysis. *Discourse Processes*, *25*(2-3), 259-284. Verkregen van http://dx.doi.org/10.1080/01638539809545028 doi: 10.1080/01638539809545028

Latent semantic analysis (lsa) tutorial. (z. j.). http://www.puffinwarellc.com/index.php/news-and-articles/articles/33-latent-semantic-analysis-tutorial.html?showall=1. (Accessed: 2014-15-11)

Maas, A. L., Daly, R. E., Pham, P. T., Huang, D., Ng, A. Y. & Potts, C. (2011). Learning word vectors for sentiment analysis. In *Proceedings of the 49th annual meeting of the association for computational linguistics: Human language technologies-volume 1* (pp. 142–150).

Manning, C. D., Raghavan, P. & Schütze, H. (2008). *Introduction to information retrieval* (Dl. 1). Cambridge university press Cambridge.

Mantrach Amin, H. B. M. S., Nicolas Vanzeebroeck. (z. j.). *Machine learning course ulb: Text mining.* https://ai.vub.ac.be/sites/default/files/textmining2011.pdf. (Accessed: 2014-15-11)

McKinney, W. (2012). *Python for data analysis: Data wrangling with pandas, numpy, and ipython.* "O'Reilly Media, Inc.
.

Mitchell, T. M. (1997). Machine learning. 1997. *Burr Ridge, IL: McGraw Hill*, *45*.

Ng, A. (z. j.). *Machine learning course.* https://class.coursera.org/ml-005/lecture/preview. (Accessed: 2014-15-11)

Petitpierre, D. & Russell, G. (1995). Mmorph-the multext morphology program. *Multext deliverable report for the task*, *2*(1).

Samuel, A. L. (2000). Some studies in machine learning using the game of checkers. *IBM Journal of research and development*, *44*(1.2), 206–226.

A tutorial on clustering algorithms. (z. j.). http://home.deib.polimi.it/matteucc/Clustering/tutorial_html/index.html. (Accessed: 2015-01-11)