IN2110: Språkteknologiske metoder

Leksikalsk semantikk og ordvektorer

Helene Bøsei Olsen

Språkteknologigruppen (LTG)

18. februar, 2025



I dag

- Fokus på ord heller enn *dokumenter*
- Leksikalsk semantikk = studien av ords betydning
- Distribusjonelle modeller av semantikk
- Eksempeloppgaver for evaluering av ordvektorer
- + litt til

BoW representasjon av tekst

- Så langt har vi antatt BoW trekk for å representere dokumenter
- Brukes også for å representere andre enheter av tekst, for eksempel setninger
- Mange klassifikasjonsoppgaver på setningsnivå i NLP:
- Eksempel: Klassifisering av polaritet:

BoW representasjon av tekst

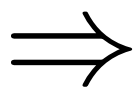
- Så langt har vi antatt BoW trekk for å representere dokumenter
- Brukes også for å representere andre enheter av tekst, for eksempel setninger
- Mange klassifikasjonsoppgaver på setningsnivå i NLP:
- Eksempel: Klassifisering av polaritet:

Jeg var imponert, dette var ikke dårlig!

BoW representasjon av tekst

- Så langt har vi antatt BoW trekk for å representere dokumenter
- Brukes også for å representere andre enheter av tekst, for eksempel setninger
- Mange klassifikasjonsoppgaver på setningsnivå i NLP:
- Eksempel: Klassifisering av polaritet:

Jeg var imponert, dette var ikke dårlig!



{var, var, jeg, dette, ikke, !, imponert, dårlig, dette}

BoW representasjon av tekst

- Så langt har vi antatt BoW trekk for å representere dokumenter
- Brukes også for å representere andre enheter av tekst, for eksempel setninger
- Mange klassifikasjonsoppgaver på setningsnivå i NLP:
- Eksempel: Klassifisering av polaritet:

Jeg var imponert, dette var ikke dårlig!

$$\Rightarrow$$

{var, var, jeg, dette, ikke, !, imponert, dårlig, dette}

- Hva mangler ved en BoW representasjon?

Komposisjonalitet

Jeg var imponert, dette var ikke dårlig!

Komposisjonalitet

Jeg var imponert, dette var ikke dårlig!

$\neq$

Jeg var ikke imponert, dette var dårlig!

Komposisjonalitet

Jeg var imponert, dette var ikke dårlig!

$\neq$

Jeg var ikke imponert, dette var dårlig!

- Setningene vil ha identisk BoW representasjon
- En forenklet, men mye brukt tilnærming for å fange opp rekkefølgsbegrensninger: n-gram (typisk bigrammer og trigrammer):

{var, var, jeg, dette, ikke, !, imponert, dårlig, dette}

vs

{'jeg var', 'var imponert' . . . 'var ikke', 'ikke dårlig', 'dårlig, !' }

Ingen informasjonsdeling

- Ingen informasjonsdeling mellom trekkene
- Alle trekk er like distinkte

- Pizzaen var fantastisk
- Margheritaen var utrolig
- Hunden var syk

- Hadde vært fint med en representasjon som visste at pizza og margerita likner hverandre, men forskjellig fra hund

Betydning

- A sense or concept is the meaning component of a word.

> Katt
>
> 1. lite husdyr og kjæledyr av kattefamilien

- Maskiner kan ikke (ennå) fange den faktiske referansen til et ord
- Vi fokuserer heller på å modellere forholdet mellom ulike ord og deres betydning
- Hva mener vi med et ords betydning?

Synonymi

- En relasjon som kan defineres på ord-nivå og betydningsnivå
- Holder for betydninger av ord når meningsinnholdet er (nært) identisk
- Holder mellom ord som kan erstatte hverandre uten å endre sannhetbetingelsen til en påstand eller en setning.

unge & barn, rask & hurtig, bikkje & hund, gøy & morsom, vann & H20

- Det lingvistiske prinsippet om kontrast: enhver endring i form markerer en endring i betydning
- Man kan argumentere for at perfekt synonymi ikke eksisterer

Beslekthet og likhet (*Relatedness and similarity*)

- Ekte synonymi er for strengt.
- Vi vil i stedet fokusere på ord med like eller beslektede/relaterte betydninger
- Ord som deler viktige betydnings-elementer
- Nyttig kunnskap for mange NLP-oppgaver

Beslekthet og likhet (*Relatedness and similarity*)

- Ekte synonymi er for strengt.
- Vi vil i stedet fokusere på ord med like eller beslektede/relaterte betydninger
- Ord som deler viktige betydnings-elementer
- Nyttig kunnskap for mange NLP-oppgaver
- Like bil: tog, sykkel, drosje, lastebil, kjøretøy, fly, buss.
- Beslektet bil: vei, drivstoff, hjul, trafikk, sjåfør, kjøre, førerkort.
- Beslektethet er bredere definert enn likhet: assosiasjon gjennom domene eller semantisk felt.

Sentiment

- Noen ganger er vi interessert i snevrere typer ordlikhet.
- F.eks. affektive betydninger eller konnotasjoner:
- Aspekter av et ords betydning knyttet til en forfatter/lesers følelser, sentiment, meninger eller vurderinger.
- Positiv/negativ konnotasjon: glad/trist.
- Positiv/negativ vurdering: fantastisk/forferdelig.
- Viktig for NLP-oppgaver som sentimentanalyse, holdningsdeteksjon (stance detection), argumentasjonsutvinning (argumentation mining), hatpratdeteksjon osv.

Modellering av leksikalsk semantikk

- Kan et program automatisk lære hvilke ord som har lignende betydninger?
- Bare ved å analysere data fra faktisk språkbruk?
- Uten noen forhåndskunnskap?
- Hvordan kan vi representere ords betydning i en matematisk modell?

Konsepter

- Distribusjonell semantikk
- Semantiske rom:
- Vektorrommodeller for ords betydning

Den distribusjonelle hypotesen

AKA den kontekstuelle betydningsteorien

- Meaning is use. (Wittgenstein, 1953)
- You shall know a word by the company it keeps. (Firth, 1957)
- The meaning of entities, and the meaning of grammatical relations among them, is related to the restriction of combinations of these entities relative to other entities. (Harris, 1968)

Den distribusjonelle hypotesen

retawerif

Den distribusjonelle hypotesen

retawerif

Han var bakfull etter å ha drukket for mange shots med retawerif på festen i går kveld.

Den distribusjonelle hypotesen

retawerif

Han var bakfull etter å ha drukket for mange shots med retawerif på festen i går kveld.

Den distribusjonelle hypotesen

- Hypotesen: Hvis to ord deler lignende kontekster, kan vi anta at de har lignende betydninger.
- Å sammenligne betydning reduseres til å sammenligne kontekst, – ingen behov for forhåndskunnskap!

Den distribusjonelle hypotesen

- Hypotesen: Hvis to ord deler lignende kontekster, kan vi anta at de har lignende betydninger.
- Å sammenligne betydning reduseres til å sammenligne kontekst, – ingen behov for forhåndskunnskap!
- Gitt prosesseringskraften til moderne datamaskiner og tilgjengeligheten av enorme mengder tekster digitalt...
- ...kan vi nå praktisk implementere de klassiske empiristiske påstandene til Firth, Harris, Wittgenstein, m.fl.

Distribusjonell semantikk i praksis

- Registrer kontekster for ord på tvers av et stort korpus
- Hvert ord representeres av et sett med trekk (features)
- Hvert trekk registrerer en viss egenskap ved de observerte kontekstene
- Ord som viser seg å ha lignende trekk, forventes også å ha liknende betydning
- Egenskaper kan representeres i en vektorrommodell, hvor likhet blir modellert som geometrisk avstand.

- Noen designvalg:
  - Hvordan definerer vi kontekst?
  - Hvordan definerer vi et ord?

Term–dokumentmatrise

- Vi har sett hvordan kolonnene representerer dokument-vektorer
- Vi kan la radene representere ord-vektorer

	As You Like It	Twelfth Night	Julius Caesar	Henry V
battle	1	0	7	13
good	114	80	62	89
fool	36	58	1	4
wit	20	15	2	3

Term–dokumentmatrise

- Vi har sett hvordan kolonnene representerer dokument-vektorer
- Vi kan la radene representere ord-vektorer

	As You Like It	Twelfth Night	Julius Caesar	Henry V
battle	1	0	7	13
good	114	80	62	89
fool	36	58	1	4
wit	20	15	2	3

Term–dokumentmatrise

- Vi har sett hvordan kolonnene representerer dokument-vektorer
- Vi kan la radene representere ord-vektorer

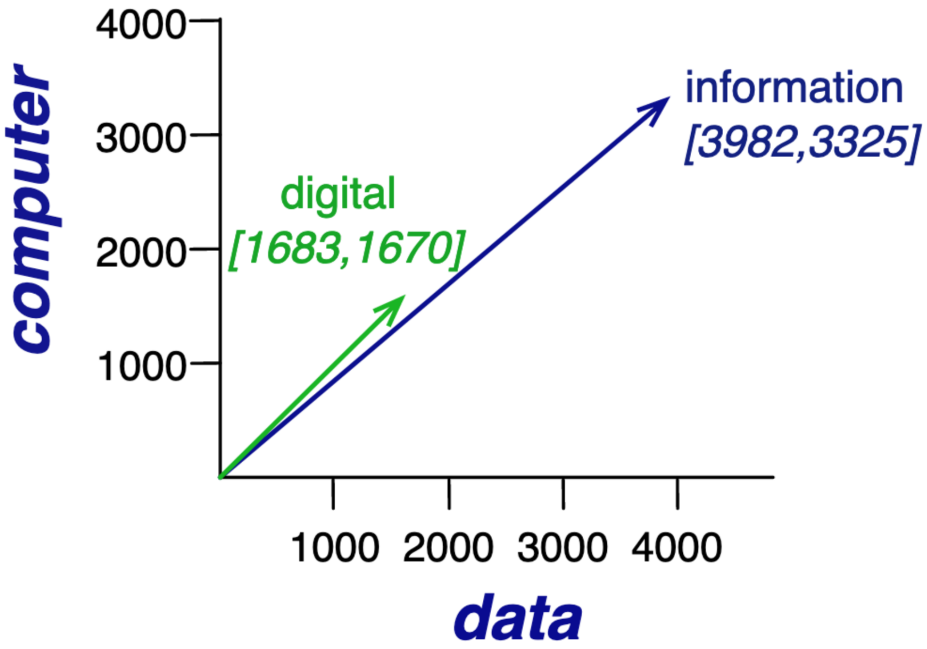
	As You Like It	Twelfth Night	Julius Caesar	Henry V
battle	1	0	7	13
good	114	80	62	89
fool	36	58	1	4
wit	20	15	2	3

- Men dokumenter er litt grove som kontekst for modellering av ords betydning.
- Vi ønsker heller å bruke en ord-koeksistensmatrise (word co-occurrence matrix)
- (aka term–term eller term–kontekstmatrise)

Term–term matrise

	aardvark	...	computer	data	result	pie	sugar
cherry	0	...	2	8	9	442	25
strawberry	0	...	0	0	1	60	19
digital	0	...	1670	1683	85	5	4
information	0	...	3325	3982	378	5	13





Definere 'kontekst'

- For å hente ut trekkvektorene (feature vectors), registrerer vi alle ord som koeksisterer med målordet i dets kontekster på tvers av et korpus.
- Men hva er 'kontekst'?
- For BoW: Dokument, setning, eller vindu ($\pm k$ ord venstre/høyre).

Definere ‘kontekst’

- For å hente ut trekkvektorene (feature vectors), registrerer vi alle ord som koeksisterer med målordet i dets kontekster på tvers av et korpus.
- Men hva er ‘kontekst’?
- For BoW: Dokument, setning, eller vindu ($\pm k$ ord venstre/høyre).
- Kan også bruke n-grammer eller grammatiske relasjoner.
- *Bredere* kontekst har en tendens til å fange beslektethet (*relatedness*).
- *Smalere* kontekst har en tendens til å fange likhet (*similarity*).
- Som for dokumentvektorer, blir de rå telleverdiene vanligvis vektet;
- ved bruk av for eksempel TF-IDF eller pointwise mutual information, eller en annen assosiasjonsmåling.

Recap: Hva er et ord?

Raw: "The programmer's programs had been programmed."

- Tokenisering: Deler opp en tekst i setninger og ord eller andre enheter
- Ulike abstraksjonsnivåer og morfologisk normalisering:
  - Hva skal vi gjøre med store bokstaver, tall, tegnsetting, sammensatte ord, osv.?, . . . ?
  - Fullformord vs. lemmatiserte ord vs. stammer . . .
- Stoppliste: Filtrere ut lukkede ordklasser eller funksjonsord.
  - Ideen er at kun *innholdsord* gir relevant kontekst

Recap: Hva er et ord?

Raw:	"The programmer's programs had been programmed."
Tokenized:	the programmer 's programs had been programmed .

- Tokenisering: Deler opp en tekst i setninger og ord eller andre enheter
- Ulike abstraksjonsnivåer og morfologisk normalisering:
  - Hva skal vi gjøre med store bokstaver, tall, tegnsetting, sammensatte ord, osv.?, . . . ?
  - Fullformord vs. lemmatiserte ord vs. stammer . . .
- Stoppliste: Filtrere ut lukkede ordklasser eller funksjonsord.
  - Ideen er at kun *innholdsord* gir relevant kontekst

Recap: Hva er et ord?

Raw:	"The programmer's programs had been programmed."
Tokenized:	the programmer 's programs had been programmed .
Lemmatized:	the programmer 's program have be program .

- Tokenisering: Deler opp en tekst i setninger og ord eller andre enheter
- Ulike abstraksjonsnivåer og morfologisk normalisering:
  - Hva skal vi gjøre med store bokstaver, tall, tegnsetting, sammensatte ord, osv.?, . . . ?
  - Fullformord vs. lemmatiserte ord vs. stammer . . .
- Stoppliste: Filtrere ut lukkede ordklasser eller funksjonsord.
  - Ideen er at kun *innholdsord* gir relevant kontekst

Recap: Hva er et ord?

Raw:	"The programmer's programs had been programmed."
Tokenized:	the programmer 's programs had been programmed .
Lemmatized:	the programmer 's program have be program .
W/ stop-list:	programmer program program

- Tokenisering: Deler opp en tekst i setninger og ord eller andre enheter
- Ulike abstraksjonsnivåer og morfologisk normalisering:
  - Hva skal vi gjøre med store bokstaver, tall, tegnsetting, sammensatte ord, osv.?, . . . ?
  - Fullformord vs. lemmatiserte ord vs. stammer . . .
- Stoppliste: Filtrere ut lukkede ordklasser eller funksjonsord.
  - Ideen er at kun *innholdsord* gir relevant kontekst

Recap: Hva er et ord?

Raw:	"The programmer's programs had been programmed."
Tokenized:	the programmer 's programs had been programmed .
Lemmatized:	the programmer 's program have be program .
W/ stop-list:	programmer program program
Stemmed:	program program program

- Tokenisering: Deler opp en tekst i setninger og ord eller andre enheter
- Ulike abstraksjonsnivåer og morfologisk normalisering:
  - Hva skal vi gjøre med store bokstaver, tall, tegnsetting, sammensatte ord, osv.?, . . . ?
  - Fullformord vs. lemmatiserte ord vs. stammer . . .
- Stoppliste: Filtrere ut lukkede ordklasser eller funksjonsord.
  - Ideen er at kun *innholdsord* gir relevant kontekst

Ord embeddings

- tradisjonelle tellebaserte vektorer er:
  - høydimensjonale (lange, $|V| =$ tusenvis eller millioner)
  - sparse (de fleste elementene er null)
  - diskrete

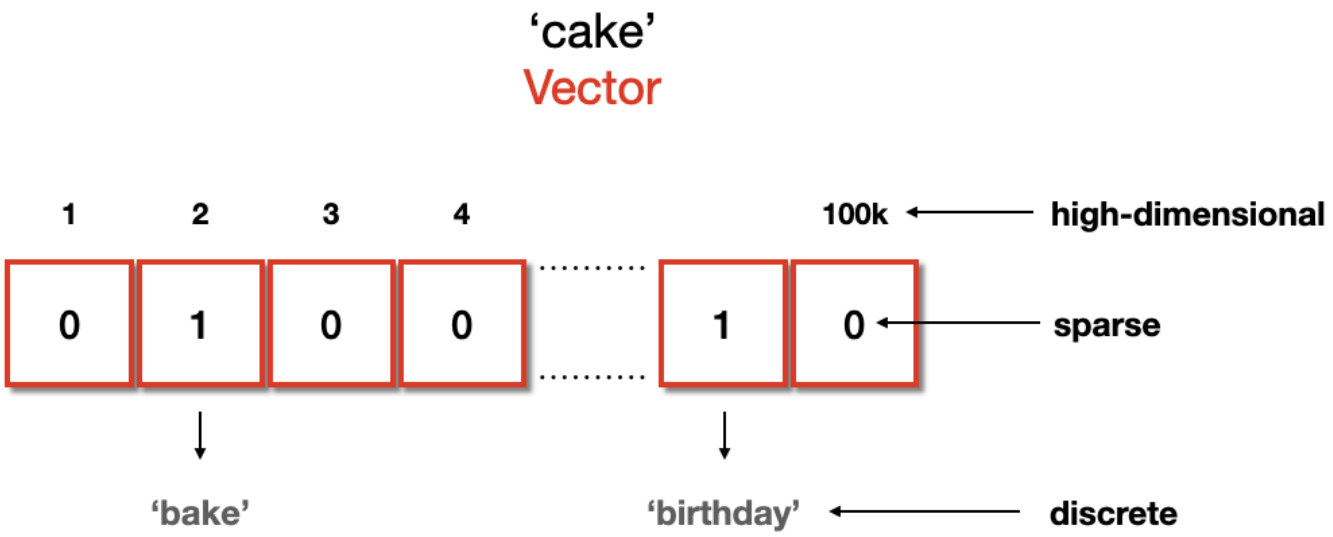
Ord embeddings

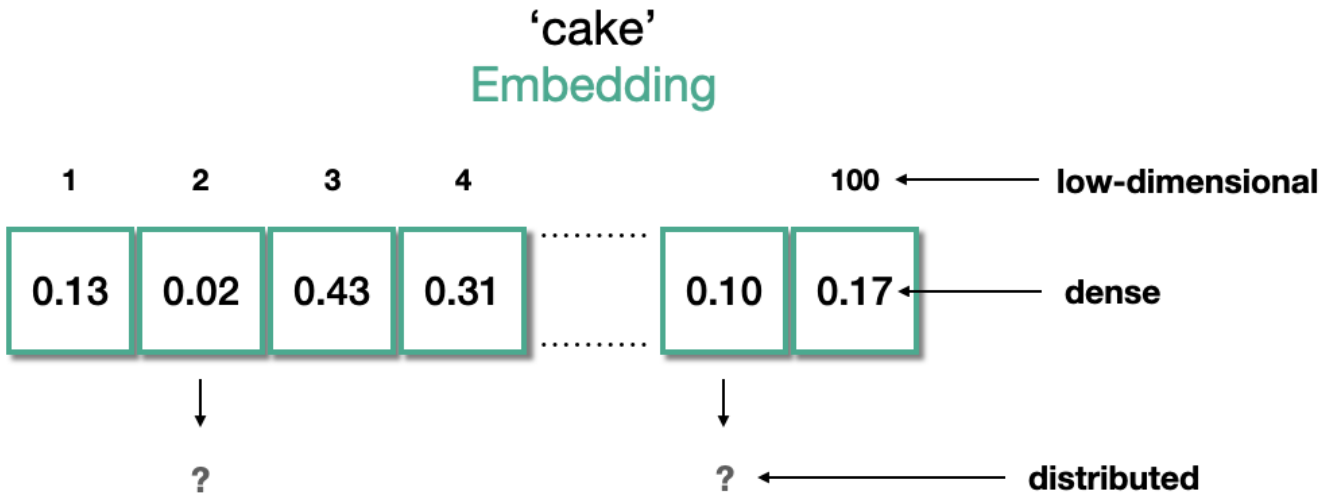
- tradisjonelle tellebaserte vektorer er:
  - høydimensjonale (lange, $|V|$ = tusenvis eller millioner)
  - sparse (de fleste elementene er null)
  - diskrete
- En ord embedding er en vektor med noen spesifikke egenskaper:
  - lavdimensjonal (kort, $|V|$ = titalls eller hundretalls)
  - dense (de fleste elementene er ikke null)
  - Kontinuerlig / distribuert

Ord embeddings

- tradisjonelle tellebaserte vektorer er:
  - høydimensjonale (lange, $|V| =$ tusenvis eller millioner)
  - sparse (de fleste elementene er null)
  - diskrete
- En ord embedding er en vektor med noen spesifikke egenskaper:
  - lavdimensjonal (kort, $|V| =$ titalls eller hundretalls)
  - dense (de fleste elementene er ikke null)
  - Kontinuerlig / distribuert
- Kan lages gjennom dimensjonsreduksjon av det opprinnelige rommet.
- Men ofte estimert gjennom prediksjonsoppgaver heller enn korpus-tellinger
- (NB: J&M bruker som regel 'embedding' og 'vector' om hverandre.)

Word embeddings





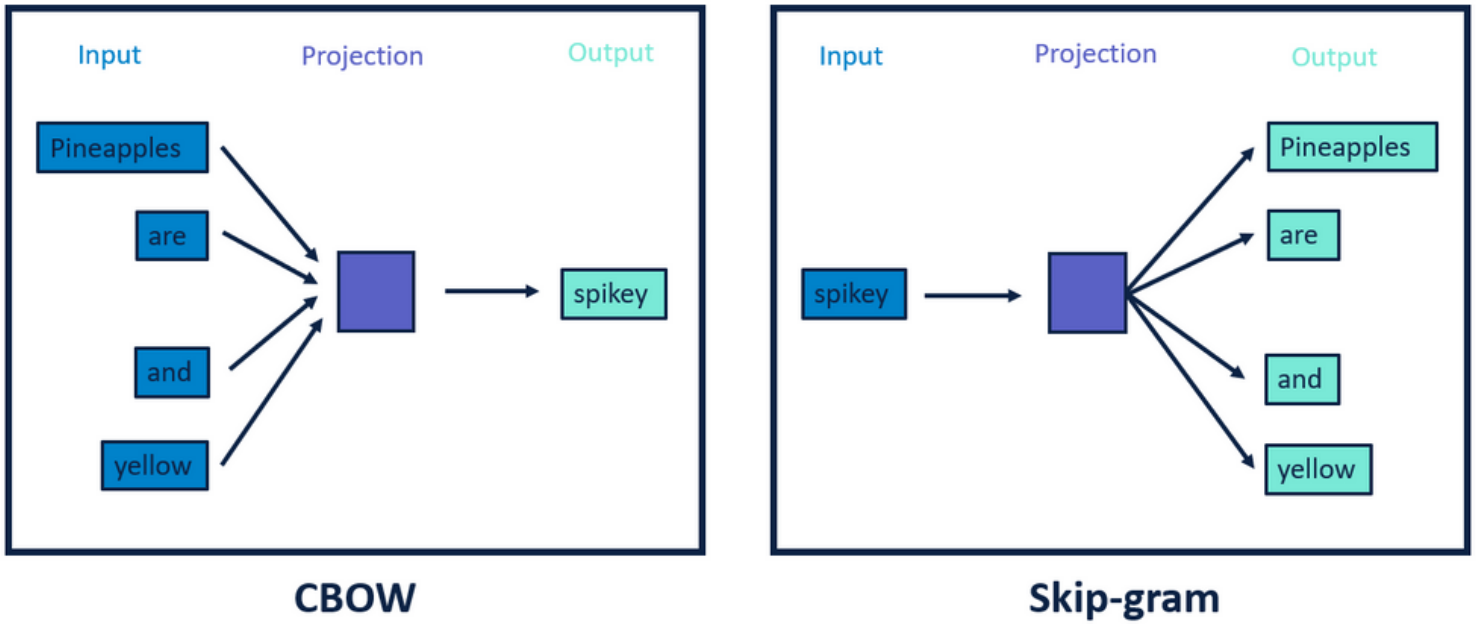
Embeddings fra språkmodellering

- Språkmodellering; har som mål å forutsi neste ord i en sekvens.
- '*Det var en mørk og stormfull . . .*'

Embeddings fra språkmodellering

- Språkmodellering; har som mål å forutsi neste ord i en sekvens.
- '*Det var en mørk og stormfull . . .*'
- Det neste observerte ordet i sekvensen gir det gyldige svaret .
- Ingen behov for hånd-merket data hand-merket data; bruk løpende tekst som treningsdata.
- Self-veiledet ('Self-supervision')
- Klassifikasjonsmodeller som logistisk regresjon (neste forelesning!) trener en vektor av vekter for å gjøre en prediksjon..
- Kan ta de lærte vektene til klassifikatoren som ord embeddings! (Her bryr vi oss egentlig ikke om selve prediksjonen)

word2vec



- Eksempel på en tilnærming som har som mål å estimere ordembeddings.
- Implementerer to algoritmer: CBOW og Skip-Gram
- I stedet for å telle hvor ofte hvert ord w forekommer nær *pineapple*, trener vi en klassifikator for å predikere: Er w sannsynlig å dukke opp nær *pineapple*?
- Et ord s som er sett nær *pineapple*, fungerer som positive eksempler
- Tilfeldig plukk ut andre ord fra leksikonet som negative eksempler.
- Bruk de lærte vektene som embeddings.

Ord embeddings

- Mange alternative algoritmer og verktøysett finnes:
- word2vec, fastText, Glove, ELMo, BERT, etc.
- Typisk basert på kunstige nevrale nettverk
- Et skifte fra feature engineering to representasjonslæring.
- Er standard input representasjon for nevrale metoder
- En viktig del av IN5550 – Neural Methods in NLP

Intrinsisk evaluering av ordvektorer

- Flere standardmåter å evaluere kvaliteten på ordvektorer.
- Hovedsakelig basert på likhetstester og analogisk resonnering.

Evaluering av embeddings (1)

- Sammenlign med menneskelige vurderinger av ordlikhet.
- Flere standard testsett tilgjengelig.
- Eksempel fra SimLex-999 (Hill et al., 2015):

word1	**word2**	**similarity**
vanish	disappear	9.8
behave	obey	7.3
belief	impression	5.95
muscle	bone	3.65
modest	flexible	0.98
hole	agreement	0.3

Evaluering av embeddings (2)

- Eksempel fra Google analogy test set (Mikolov et al. 2013)
- Q: *man* is to *woman* as *king* is to ???

Evaluering av embeddings (2)

- Eksempel fra Google analogy test set (Mikolov et al. 2013)
- Q: *man* is to *woman* as *king* is to ??? (A: *queen*)

Evaluering av embeddings (2)

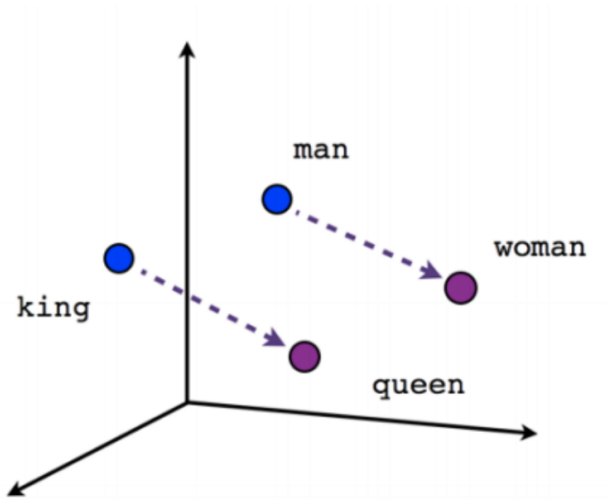
- Eksempel fra Google analogy test set (Mikolov et al. 2013)
- Q: *man* is to *woman* as *king* is to ??? (A: *queen*)
- Q: *walking* is to *walked* as *swimming* is to ???

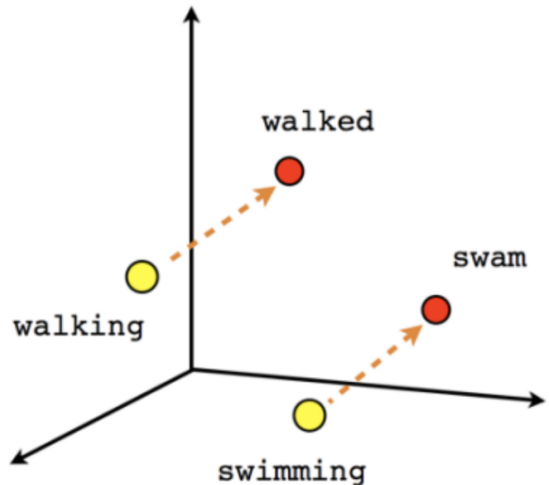
Evaluering av embeddings (2)

- Eksempel fra Google analogy test set (Mikolov et al. 2013)
- Q: *man* is to *woman* as *king* is to ??? (A: *queen*)
- Q: *walking* is to *walked* as *swimming* is to ??? (A: *swam*)

Evaluering av embeddings (2)

- Eksempel fra Google analogy test set (Mikolov et al. 2013)
- Q: *man* is to *woman* as *king* is to ??? (A: *queen*)
- Q: *walking* is to *walked* as *swimming* is to ??? (A: *swam*)
- En overraskende egenskap ved ordembeddings er at ord analogier ofte kan løses med vektoraritmetikk.
- vec('king') − vec('man') + vec('woman') ≈ vec('queen')
- vec('swimming') − vec('walking') + vec('swim') ≈ vec('swam')



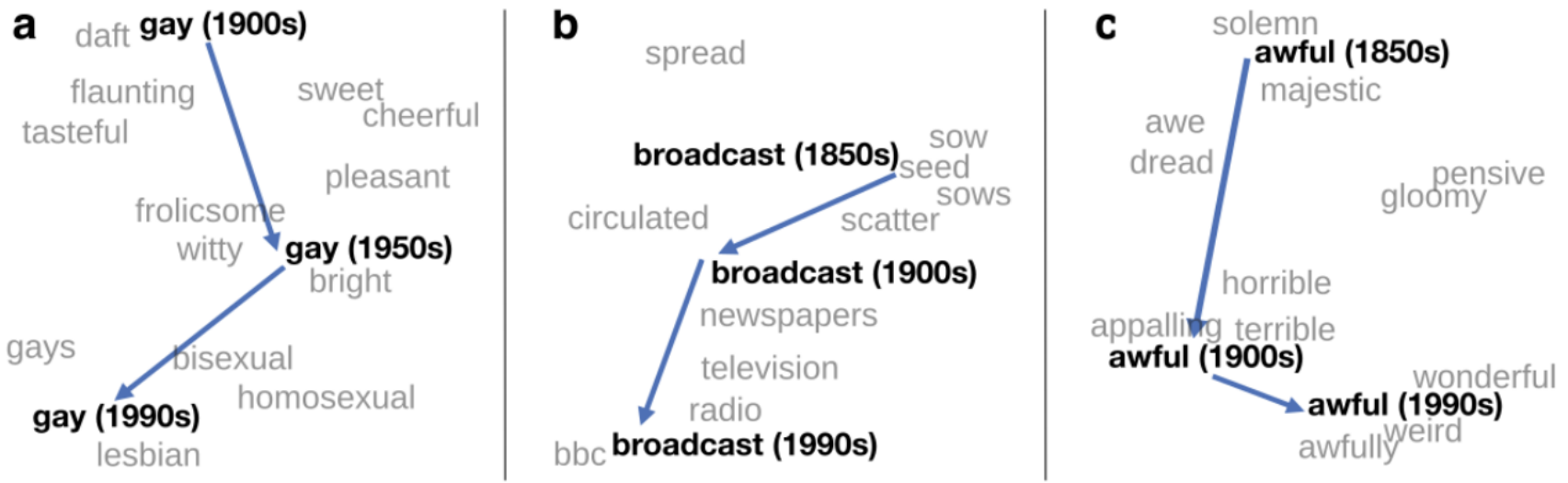


Examples from Google analogy test set

	Relation type	# Questions	Word pair 1		Word pair 2	
5* Semantic	Common capital city	506	Athens	Greece	Baghdad	Iraq
	All capital cities	4,524	Abuja	Nigeria	Accra	Ghana
	Currency	866	Algeria	dinar	Angola	kwanza
	City-in-state	2,467	Chicago	Illinois	Houston	Texas
	Man–woman	506	boy	girl	brother	sister
9* Syntactic	Adjective-to-adverb	992	amazing	amazingly	apparent	apparently
	Opposite	812	acceptable	unacceptable	aware	unaware
	Comparative	1,332	bad	worse	big	bigger
	Superlative	1,122	bad	worst	big	biggest
	Present participle	1,056	code	coding	dance	dancing
	Nationality adjective	1,599	Albania	Albanian	Argentina	Argentinean
	Past tense	1,560	dancing	danced	decreasing	decreased
	Plural nouns	1,332	banana	bananas	bird	birds
	Plural verbs	870	decrease	decreases	describe	describes

Diakroniske ordembeddings: språkendring

- Et annet eksempel på bruksområde eller evalueringsoppgave:
- Embeddings kan hjelpe til med å studere ords historie!
- Tren embeddings på korpus fra forskjellige tidsperioder.
- Følg hvordan betydningen endrer seg over tid ved å sammenligne endringer i vektoravstander og nabo-relasjoner.



Ord vektor demo

http://vectors.nlpl.eu/explore/embeddings/en/

Utfordringer med ordvektorer

Sammenslåing av ords betydninger

- En enkelt vektor for hvert ord
- Problem med polysemi og homonymi.

Utfordringer med ordvektorer

Sammenslåing av ords betydninger

- En enkelt vektor for hvert ord
- Problem med polysemi og homonymi.

Bias

- Menneskelig bias i treningsdataen vil bli reflektert i ordvektorene
- F.eks. assosiasjoner mellom kjønn og stereotypiske yrker.

Utfordringer med ordvektorer

Sammenslåing av ords betydninger

- En enkelt vektor for hvert ord
- Problem med polysemi og homonymi.

Bias

- Menneskelig bias i treningsdataen vil bli reflektert i ordvektorene
- F.eks. assosiasjoner mellom kjønn og stereotypiske yrker.

Antonymer

- Betydninger som er motsetninger
- *lang/kort*, *rask/treg*, *mørk/lys*, *opp/ned*, *stige/falle*, *varm/kald*, *inn/ut*
- Vanskelig å distribusjonelt skille fra synonymer

Tilbake til representasjon av dokumenter og setninger



- Ordembeddings gir en god representasjon av ord.
- Lar oss modellere ordlikhet.
- Hva med setningslikhet eller dokumentlikhet?
- Fra ord embeddings til tekst embeddings.

Tilbake til representasjon av dokumenter og setninger



- Ordembeddings gir en god representasjon av ord.
- Lar oss modellere ordlikhet.
- Hva med setningslikhet eller dokumentlikhet?
- Fra ord embeddings til tekst embeddings.
- Sentimentanalyse som et eksempel:
- Vi ønsker å trene og bruke en setningsklassifikator for å predikere positiv / negativ polaritet i setninger.

Recap: Informasjonsdeling mellom trekk

- BoW-representasjoner av setninger:
- Ingen informasjonsdeling mellom trekk
- Alle trekk er like distinkte

- Pizzaen var fantastisk
- Margheritaen var utrolig
- Hunden var syk

- Det hadde vært fint om våre BoW-representasjoner visste at *pizza* og *margherita* er like hverandre (men ikke like *hund*).

Embeddings to the rescue

- Kan representere dokumenter ved å konkatenere eller ta gjennomsnittet av ordembeddingene
- For en sentning $s = w_1, \ldots w_n$ med ord embeddings $x_1, \ldots x_n$, så kan vi beregne setningsembedding som $\text{vec}(s) = \frac{1}{n} \sum_{i=1}^{n} x_i$
- Kan deretter bruke $\text{vec}(s)$ som input til en setningsklassifikator.
- Fordel; bevarer likhetsrelasjon mellom trekk
- Lar oss 'forhåndstrene' (*pre-train)* ord represantasjoner på umerkede (*unlabeled* data), før vi trener f.eks. en setningsklassifikator på merkede data (*labeled* data).

Semantiske vektorer; oppsummering

- Både tradisjonelle ordvektorer og mer moderne embeddings lar oss modellere ordlikhet ved hjelp av råtekst; ingen merkede data nødvendig.
- Semantisk likhet modelleres som distribusjonell likhet, som igjen beregnes som vektoravstand.
- Tellebaserte vektorer (høy-dimensjonale, sparse, diskrete)
- Embeddings (lav-dimensjonale, dense, distribuerte)
- I moderne NLP spiller embeddings en svært sentral rolle
- Standard inputrepresentasjon for nevrale klassifikasjonsmodeller.

Neste uke

- Logistisk Regresjon
- Veiledet maskinlæringsmodell for klassifisering
- Et av de viktigste verktøyene vi har i språkteknologi
- Viktig byggekloss i mer avanserte maskinlæringsmodeller (bl.a. nevrale nettverk)