

# Obravnava delno strukturiranih podatkov in vložitve (*embeddings*)

Ljupčo Todorovski  
Univerza v Ljubljani, Fakulteta za upravo

Maj 2018

# Pregled predavanja

## Delno strukturirani podatki

Besedila, slike, ...

## Model vreče besed (*bag of words*)

- Slovar terminov
- Frekvenci terminov in dokumentov
- Frekvenca terminov in obratna frekvenca dokumentov (*TFIDF*)

## Vložitve (*embeddings*) besedil

- Pomenska podobnost besed
- Vložitvi word2vec in doc2vec

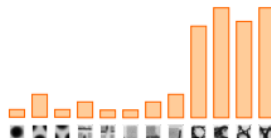
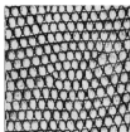
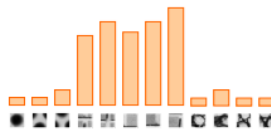
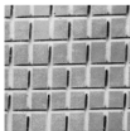
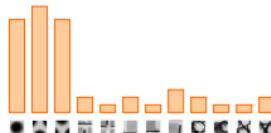
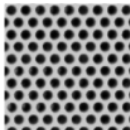
# Motivacija: klasifikacija besedil



# Motivacija: razpoznavanje slik



# Osnovna ideja: slovar besed



# Formalizacija modela

Tri komponente  $O$ ,  $E$  in  $f$

- Osnovna podatkovna množica  $O$ : delno strukturirani podatki
- Slovar  $E$ : množica osnovnih podatkovnih elementov
- Funkcija frekvence  $f : E \times O \rightarrow \mathbb{N}_0$ : vrne frekvenco (število) pojavitev  $e \in E$  v  $o \in O$

$|E|$  spremenljivk za predstavitev delno strukturiranih podatkov iz  $O$

- Vsakemu osnovnemu elementu  $e \in E$  priredimo spremenljivko  $X_e$
- Vrednost  $X_e$  za podatek  $o \in O$  je enaka  $f(e, o)$  ali njeni transformaciji

# Transformacije $f$ za besedila: frekvenca termina $TF$

## Besedila

- $O$  je množica besedil dokumentov  $d \in O$
- $E$  je slovar besed, imenovanih terminov (*terms*)  $t \in E$

Frekvenca termina, *term frequency*,  $TF(t, d) = tf_{t,d} = f(t, d)$

Imenujemo jo tudi naravna frekvenca besede oziroma termina.

## Alternativne mere $TF$

- Boolova  $TF_B(t, d) = I(TF(t, d) > 0)$
- Normalizirana  $TF_N(t, d) = TF(t, d)/|d|$ ,  $|d|$  je število besed v  $d$
- Logaritmična  $TF_L(t, d) = \log(1 + TF(t, d))$

# Transformacije $f$ za besedila: frekvenca v dokumentih $DF$

## Slabost frekvenca termina $TF$

Pogoste so besede brez pomena (prazne besede, *stop words*), ki se pojavijo v vsakem dokumentu iz  $O$ , na primer *in*, *ali* ali *biti*. Možni rešitvi:

- Praznih besed ne damo v slovar  $E$
- Upoštevamo frekvenco v dokumentih  $DF$

## Frekvenca v dokumentih, *document frequency* $DF$

$$DF(t) = |\{d \in O : TF(t, d) > 0\}|$$



# Transformacije $f$ za besedila: *TFIDF*

Obratna frekvenca v dokumentih, *inverse document frequency IDF*

$$IDF(t) = \log \frac{|O|}{DF(t)}$$

Za prazne besede  $t$  je  $IDF(t) \approx 0$ , saj za njih velja  $DF(t) \approx |O|$ .

Alternativi meri *IDF*

- Gladka  $IDF_G(t) = \log(1 + |O|/DF(t))$
- Verjetnostna  $IDF_P(t) = \log(|O|/DF(t) - 1)$

$$TFIDF(t, d) = TF(t, d) \cdot IDF(t)$$

# Orodja za pred-obdelavo besedil

## Lematizacija

- Pretvorba besede v njeno osnovno obliko
- Na primer: besede *je*, *sem* in *so* imajo osnovno obliko *biti*
- Lematizator je orodje, ki besedi pripiše njeno osnovno obliko
- Lematizator za slovenščino dostopen na [lemmatise.ijs.si](http://lemmatise.ijs.si)

## Krnenje (*stemming*)

- Poenostavljena in hitra lematizacija, pretvorba besede v njen koren
- Na primer: besedi *delati* in *delovanje* imata skupni koren *delo*
- Krnilnik: orodje, ki besedi pripiše njen koren

# Primer obdelave besedil: pet stavkov in lematizacija

## Osnovni stavki

- Hotel je večji gostinski obrat, v katerem se dobi prenočišče in hrana.
- Motel je gostinski obrat hotelskega tipa, navadno ob velikih cestah zunaj naselij.
- Bi izbrali hotel ali motel?
- Janez je izbral hotel.
- Metka je najela sobo 215 v motelu Medno.

## Lematizirani in pred-obdelani stavki

- hotel biti velik gostinski obrat v kateri se dobiti prenočišče in hrana
- motel biti gostinski obrat hotelski tip navadno ob velik cesta zunaj naselje
- biti izbrati hotel ali motel
- janez biti izbrati hotel
- metka biti najeti soba 215 v motel meden

Primer obdelave besedil: izračun  $TF_B$  in  $IDF$ 

$$E = \{biti, hotel, motel, obrat, hrana\}$$

ID	biti	hotel	hrana	motel	obrat
1	1	1	1	0	1
2	1	0	0	1	1
3	1	1	0	1	0
4	1	1	0	0	0
5	1	0	0	1	0
$DF$	5	3	1	3	2
$IDF = \log(5/DF)$	0	0.7370	2.3219	0.7370	1.3219

Primer obdelave besedil: izračun *TFIDF*

ID	biti	hotel	hrana	motel	obrat
1	0	0.7370	2.3219	0	1.3219
2	0	0	0	0.7370	1.3219
3	0	0.7370	0	0.7370	0
4	0	0.7370	0	0	0
5	0	0	0	0.7370	0

# Kako pridemo do slovarja $E$ ?

## Za besedila

- Običajno vse besede v dokumentih iz  $O$
- Včasih tudi izbrane besede in njih kombinacije,  $n$ -grami
- Osnovne (korenske) oblike za lematizirana (krnenja) besedila

## Za slike

- Vizualni slovar: koščki slik, ki jih lahko najdemo v slikah iz  $O$
- Lahko ga dobimo avtomatsko z iskanjem pogostih vzorcev v slikah

# Kako implementiramo $f$ ?

## Za besedila

Preprosto primerjanje besed.

## Za slike

Konvolucija: konvolucijske ravni nevronske mreže pravzaprav avtomatsko odkrivajo vizualni slovar.

# Omejitve

## Pomen besed popolnoma nepomemben

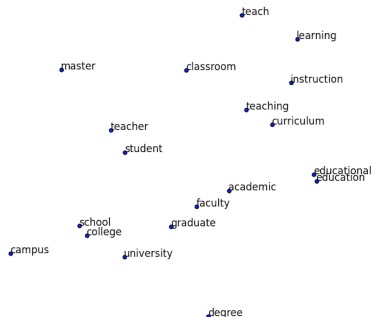
Podobne besede iz slovarja predstavljene z različnimi spremenljivkami.  
Rešitev: vložitve, drugi del današnjega predavanja.

## Veliko število spremenljivk

Rešitev: metode za izbiro in konstrukcijo napovednih spremenljivk, naslednja predavanja.



# Osnovna ideja



Pretvorba besed v vektorje realnih števil

Omejitev: Podobni vektorji ustrezajo podobnim besedam.

Porazdelitvena predpostavka (*distributional hypothesis*)

Besede s podobnim pomenom uporabljamo v istem kontekstu.

# Kaj je kontekst kontekst?

## Dva lematizirana stavka

- hotel biti velik gostinski obrat v kateri se dobiti prenočišče in hrana
- motel biti gostinski obrat hotelski tip navadno ob velik cesta zunaj naselje

## Štiri-besedni kontekst besede *obrat*

- Prvi stavek: *velik, gostinski, v, kateri*
- Drugi stavek: *biti, gostinski, hotelski, tip*

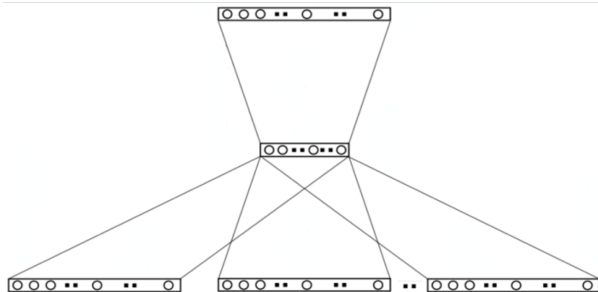
## Dva možna napovedna modela

- Napovedne spremenljivke: kontekst, ciljna spremenljivka: beseda
- Napovedna spremenljivka: beseda, ciljne spremenljivke: kontekst

# Model *CBOW*, *Continuous Bag-of-Words*

## Struktura nevronske mreže

- Vhodna raven:  $k \cdot |E|$  nevronov,  $k$  je velikost konteksta,  $E$  slovar
- Skrita raven:  $m$  nevronov,  $m$  je dolžina predstavitevnihih vektorjev
- Izhodna raven:  $|E|$  nevronov



# Opis nevronske mreže *CBOW*

Vrednosti vhodnih ( $x$ ), izhodnih ( $y$ ) in skritih ( $h$ ) nevronov

- $x_{i,e}$ : vrednost 1, če je  $e$   $i$ -ta beseda v kontekstu, sicer 0
- $y_e$ : verjetnost, da je centralna beseda  $e$
- $h_i$ : vrednost  $i$ -te dim. predstavitvenega vektorja centralne besede

Ciljna funkcija za optimizacijo

$$Err = -y_w = -\frac{e^{v_w}}{\sum_{e \in E} e^{v_e}},$$

kjer je  $w$  je centralna beseda v opazovanem kontekstu.

# Opis nevronske mreže *CBOW*

Funkcija aktivacije skritih nevronov  $\phi$

$$\phi(v) = \frac{1}{1 - e^{-v}}$$

Funkcija aktivacije izhodnih nevronov  $\phi$

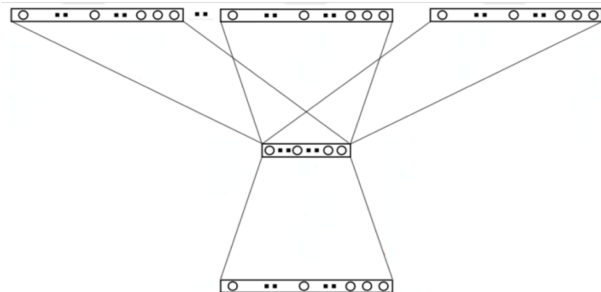
$$\phi(v) = \frac{e^v}{\sum_{e \in E} e^{v_e}}$$

Učni algoritem: običajno vzvratno razširjanje napake.

# Model *Skip-Gram*

## Struktura nevronske mreže

- Vhodna raven:  $|E|$  nevronov
- Skrita raven:  $m$  nevronov,  $m$  je dolžina predstavitvenih vektorjev
- Izhodna raven:  $k \cdot |E|$  nevronov,  $k$  je velikost konteksta,  $E$  slovar



# Opis nevronske mreže *Skip-Gram*

Vrednosti vhodnih ( $x$ ), izhodnih ( $y$ ) in skritih ( $h$ ) nevronov

- $x_e$ : vrednost 1, če je centralna beseda  $e$
- $y_{i,e}$ : verjetnost, da je  $e$   $i$ -ta beseda v kontekstu
- $h_i$ : vrednost  $i$ -te dim. predstavitevnega vektorja centralne besede

Ciljna funkcija za optimizacijo

$$Err = - \prod_{i=1}^k y_{i,w_i} = - \prod_{i=1}^k \frac{e^{v_{i,w_i}}}{\sum_{e \in E} e^{v_{i,e}}},$$

kjer so  $w_i$  besede v opazovanem kontekstu.

# Opis nevronske mreže *Skip-Gram*

Funkcija aktivacije skritih nevronov  $\phi$

$$\phi(v) = \frac{1}{1 - e^{-v}}$$

Funkcija aktivacije izhodnih nevronov  $\phi$

$$\phi(v) = \frac{e^v}{\sum_{e \in E} e^{v_e}}$$

Učni algoritem: običajno vzvratno razširjanje napake.



# Ohranjanje pomena besed

FRANCE	JESUS	XBOX	REDDISH	SCRATCHED	MEGABITS
AUSTRIA	GOD	AMIGA	GREENISH	NAILED	OCTETS
BELGIUM	SATI	PLAYSTATION	BLUISH	SMASHED	MB/S
GERMANY	CHRIST	MSX	PINKISH	PUNCHED	BIT/S
ITALY	SATAN	IPOD	PURPLISH	POPPED	BAUD
GREECE	KALI	SEGA	BROWNISH	CRIMPED	CARATS
SWEDEN	INDRA	PSNUMBER	GREYISH	SCRAPED	KBIT/S
NORWAY	VISHNU	HD	GRAYISH	SCREWED	MEGAHERTZ
EUROPE	ANANDA	DREAMCAST	WHITISH	SECTIONED	MEGAPIXELS
HUNGARY	PARVATI	GEFORCE	SILVERY	SLASHED	GBIT/S
SWITZERLAND	GRACE	CAPCOM	YELLOWISH	RIPPED	AMPERES

# Ohranjanje razmerij analogije med besedami

Relationship	Example 1	Example 2	Example 3
France - Paris	Italy: Rome	Japan: Tokyo	Florida: Tallahassee
big - bigger	small: larger	cold: colder	quick: quicker
Miami - Florida	Baltimore: Maryland	Dallas: Texas	Kona: Hawaii
Einstein - scientist	Messi: midfielder	Mozart: violinist	Picasso: painter
Sarkozy - France	Berlusconi: Italy	Merkel: Germany	Koizumi: Japan
copper - Cu	zinc: Zn	gold: Au	uranium: plutonium
Berlusconi - Silvio	Sarkozy: Nicolas	Putin: Medvedev	Obama: Barack
Microsoft - Windows	Google: Android	IBM: Linux	Apple: iPhone
Microsoft - Ballmer	Google: Yahoo	IBM: McNealy	Apple: Jobs
Japan - sushi	Germany: bratwurst	France: tapas	USA: pizza

# Praktična uporaba za besedilne podatke

- ① Naučimo word2vec model iz čim večjega nabora besedil
- ② Uporabljamo ga za pretvorbo posameznih besed v vektorje
- ③ Od vektorjev za besede do vektorja za besedilo
  - Izračunamo povprečje vektorjev za vse besede v besedilu
  - Izračunamo povprečje vektorjev za nekaj izbranih besed, npr. tistih z največjim *TFIDF*
  - Naredimo stik vektorjev za nekaj izbranih besed

Ali pa uporabimo doc2vec, verzijo word2vec za besedila.

# Enostavna nadgradnja modela *CBOW*

## Nadgradimo vhodno raven nevronov

- Kontekstu dodamo še eno vhodno spremenljivko: ID besedila
- Iz konteksta in ID besedila, napovedujemo centralno besedo

## Skriti nevroni

Enako kot pri word2vec, podajo predstavitev besedila z vektorjem realnih števil: podobna besedila predstavljena s podobnimi vektorji.

## Učenje mreže za vsako novo besedilo

Spreminjajo se le tiste uteži, ki se nanašajo na ID besedila, ne pa tiste, ki se nanašajo na kontekst.

# Algoritmi in implementacije

word2vec (Mikolov in ost 2013)

Implementacija v R [github.com/bmschmidt/wordVectors](https://github.com/bmschmidt/wordVectors)

doc2vec (Le in Mikolov 2014)

Implementacija v R: paket textTinyR