Module 10.2: Distributed Representations of words

- *You shall know a word by the company it keeps - Firth, J. R. 1957:11*

A bank is a **financial** institution that accepts **deposits** from the public and creates **credit**.

- *You shall know a word by the company it keeps - Firth, J. R. 1957:11*
- Distributional similarity based representations

A bank is a **financial** institution that accepts **deposits** from the public and creates **credit**.

A bank is a **financial** institution that accepts **deposits** from the public and creates **credit**.

- *You shall know a word by the company it keeps - Firth, J. R. 1957:11*
- Distributional similarity based representations
- This leads us to the idea of co-occurrence matrix

- *You shall know a word by the company it keeps - Firth, J. R. 1957:11*
- Distributional similarity based representations
- This leads us to the idea of co-occurrence matrix

A bank is a **financial** institution that accepts **deposits** from the public and creates **credit**.

The idea is to use the accompanying words (financial, deposits, credit) to represent bank

Corpus:

- Human machine interface for computer applications
- User opinion of computer system response time
- User interface management system
- System engineering for improved response time

- A co-occurrence matrix is a **terms** $\times$ **terms** matrix which captures the number of times a term appears in the context of another term

Corpus:

- Human machine interface for computer applications
- User opinion of computer system response time
- User interface management system
- System engineering for improved response time

- A co-occurrence matrix is a **terms** $\times$ **terms** matrix which captures the number of times a term appears in the context of another term
- The context is defined as a window of k words around the terms

Corpus:

- Human machine interface for computer applications
- User opinion of computer system response time
- User interface management system
- System engineering for improved response time

- A co-occurrence matrix is a **terms** $\times$ **terms** matrix which captures the number of times a term appears in the context of another term
- The context is defined as a window of k words around the terms
- Let us build a co-occurrence matrix for this toy corpus with $k = 2$

Corpus:

- Human machine interface for computer applications
- User opinion of computer system response time
- User interface management system
- System engineering for improved response time

	human	machine	system	for	...	user
human	0	1	0	1	...	0
machine	1	0	0	1	...	0
system	0	0	0	1	...	2
for	1	1	1	0	...	0
.	.	.	.	.	.	.
.	.	.	.	.	.	.
.	.	.	.	.	.	.
user	0	0	2	0	...	0

Co-occurence Matrix

- A co-occurrence matrix is a **terms** $\times$ **terms** matrix which captures the number of times a term appears in the context of another term
- The context is defined as a window of k words around the terms
- Let us build a co-occurrence matrix for this toy corpus with $k = 2$
- This is also known as a **word** $\times$ **context** matrix

Corpus:

- Human machine interface for computer applications
- User opinion of computer system response time
- User interface management system
- System engineering for improved response time

	human	machine	system	for	...	user
human	0	1	0	1	...	0
machine	1	0	0	1	...	0
system	0	0	0	1	...	2
for	1	1	1	0	...	0
.	.	.	.	.	.	.
.	.	.	.	.	.	.
.	.	.	.	.	.	.
user	0	0	2	0	...	0

Co-occurence Matrix

- A co-occurrence matrix is a **terms** $\times$ **terms** matrix which captures the number of times a term appears in the context of another term
- The context is defined as a window of k words around the terms
- Let us build a co-occurrence matrix for this toy corpus with $k = 2$
- This is also known as a **word** $\times$ **context** matrix
- You could choose the set of **words** and **contexts** to be same or different

Corpus:

- Human machine interface for computer applications
- User opinion of computer system response time
- User interface management system
- System engineering for improved response time

	human	machine	system	for	...	user
human	0	1	0	1	...	0
machine	1	0	0	1	...	0
system	0	0	0	1	...	2
for	1	1	1	0	...	0
.	.	.	.	.	.	.
.	.	.	.	.	.	.
.	.	.	.	.	.	.
user	0	0	2	0	...	0

Co-occurence Matrix

- A co-occurrence matrix is a **terms** × **terms** matrix which captures the number of times a term appears in the context of another term
- The context is defined as a window of k words around the terms
- Let us build a co-occurrence matrix for this toy corpus with $k = 2$
- This is also known as a **word** × **context** matrix
- You could choose the set of **words** and **contexts** to be same or different
- Each row (column) of the co-occurrence matrix gives a vectorial representation of the corresponding word (context)

	human	machine	system	for	...	user
human	0	1	0	1	...	0
machine	1	0	0	1	...	0
system	0	0	0	1	...	2
for	1	1	1	0	...	0
.	.	.	.	.	.	.
.	.	.	.	.	.	.
.	.	.	.	.	.	.
user	0	0	2	0	...	0

Some (fixable) problems

- Stop words (a, the, for, etc.) are very frequent → these counts will be very high

	human	machine	system	...	user
human	0	1	0	...	0
machine	1	0	0	...	0
system	0	0	0	...	2
.	.	.	.	.	.
.	.	.	.	.	.
.	.	.	.	.	.
user	0	0	2	...	0

Some (fixable) problems

- Stop words (a, the, for, etc.) are very frequent → these counts will be very high
- Solution 1: Ignore very frequent words

	human	machine	system	for	...	user
human	0	1	0	x	...	0
machine	1	0	0	x	...	0
system	0	0	0	x	...	2
for	x	x	x	x	...	x
.	.	.	.	.	.	.
.	.	.	.	.	.	.
.	.	.	.	.	.	.
user	0	0	2	x	...	0

Some (fixable) problems

- Stop words (a, the, for, etc.) are very frequent $\rightarrow$ these counts will be very high
- Solution 1: Ignore very frequent words
- Solution 2: Use a threshold t (say, t = 100)

$$X_{ij} = min(count(w_i, c_j), t),$$

where w is word and c is context.

Some (fixable) problems

- Solution 3: Instead of $count(w, c)$ use $PMI(w, c)$

Some (fixable) problems

- Solution 3: Instead of $count(w, c)$ use $PMI(w, c)$

$$PMI(w, c) = \log \frac{p(c|w)}{p(c)}$$
$$= \log \frac{count(w, c) * N}{count(c) * count(w)}$$

N *is the total number of words*

	human	machine	system	for	...	user
human	0	2.944	0	2.25	...	0
machine	2.944	0	0	2.25	...	0
system	0	0	0	1.15	...	1.84
for	2.25	2.25	1.15	0	...	0
.	.	.	.	.	.	.
.	.	.	.	.	.	.
.	.	.	.	.	.	.
user	0	0	1.84	0	...	0

Some (fixable) problems

- Solution 3: Instead of $count(w, c)$ use $PMI(w, c)$

$$PMI(w, c) = \log \frac{p(c|w)}{p(c)}$$
$$= \log \frac{count(w, c) * N}{count(c) * count(w)}$$

N *is the total number of words*

	human	machine	system	for	...	user
human	0	2.944	0	2.25	...	0
machine	2.944	0	0	2.25	...	0
system	0	0	0	1.15	...	1.84
for	2.25	2.25	1.15	0	...	0
.	.	.	.	.	.	.
.	.	.	.	.	.	.
.	.	.	.	.	.	.
user	0	0	1.84	0	...	0

Some (fixable) problems

- Solution 3: Instead of $count(w, c)$ use $PMI(w, c)$

$$PMI(w, c) = \log \frac{p(c|w)}{p(c)}$$
$$= \log \frac{count(w, c) * N}{count(c) * count(w)}$$

N *is the total number of words*

- If $count(w, c) = 0$, $PMI(w, c) = -\infty$

	human	machine	system	for	...	user
human	0	2.944	0	2.25	...	0
machine	2.944	0	0	2.25	...	0
system	0	0	0	1.15	...	1.84
for	2.25	2.25	1.15	0	...	0
.	.	.	.	.	.	.
.	.	.	.	.	.	.
.	.	.	.	.	.	.
user	0	0	1.84	0	...	0

Some (fixable) problems

- Solution 3: Instead of $count(w, c)$ use $PMI(w, c)$

$$PMI(w, c) = \log \frac{p(c|w)}{p(c)}$$
$$= \log \frac{count(w, c) * N}{count(c) * count(w)}$$

N *is the total number of words*

- If $count(w, c) = 0$, $PMI(w, c) = -\infty$

Instead use,

$$PMI_0(w, c) = PMI(w, c) \quad if\ count(w, c) > 0$$
$$= 0 \quad otherwise$$

	human	machine	system	for	...	user
human	0	2.944	0	2.25	...	0
machine	2.944	0	0	2.25	...	0
system	0	0	0	1.15	...	1.84
for	2.25	2.25	1.15	0	...	0
.	.	.	.	.	.	.
.	.	.	.	.	.	.
.	.	.	.	.	.	.
user	0	0	1.84	0	...	0

Some (fixable) problems

- Solution 3: Instead of $count(w, c)$ use $PMI(w, c)$

$$PMI(w, c) = \log \frac{p(c|w)}{p(c)}$$
$$= \log \frac{count(w, c) * N}{count(c) * count(w)}$$

N *is the total number of words*

- If $count(w, c) = 0$, $PMI(w, c) = -\infty$

Instead use,

$$PMI_0(w, c) = PMI(w, c) \quad if\ count(w, c) > 0$$
$$= 0 \quad otherwise$$

or

$$PPMI(w, c) = PMI(w, c) \quad if\ PMI(w, c) > 0$$
$$= 0 \quad otherwise$$

Some (severe) problems

- Very high dimensional ($|V|$)

	human	machine	system	for	...	user
human	0	2.944	0	2.25	...	0
machine	2.944	0	0	2.25	...	0
system	0	0	0	1.15	...	1.84
for	2.25	2.25	1.15	0	...	0
.	.	.	.	.	.	.
.	.	.	.	.	.	.
.	.	.	.	.	.	.
user	0	0	1.84	0	...	0

	human	machine	system	for	...	user
human	0	2.944	0	2.25	...	0
machine	2.944	0	0	2.25	...	0
system	0	0	0	1.15	...	1.84
for	2.25	2.25	1.15	0	...	0
.	.	.	.	.	.	.
.	.	.	.	.	.	.
.	.	.	.	.	.	.
user	0	0	1.84	0	...	0

Some (severe) problems

- Very high dimensional ($|V|$)
- Very sparse

	human	machine	system	for	...	user
human	0	2.944	0	2.25	...	0
machine	2.944	0	0	2.25	...	0
system	0	0	0	1.15	...	1.84
for	2.25	2.25	1.15	0	...	0
.	.	.	.	.	.	.
.	.	.	.	.	.	.
.	.	.	.	.	.	.
user	0	0	1.84	0	...	0

Some (severe) problems

- Very high dimensional ($|V|$)
- Very sparse
- Grows with the size of the vocabulary

	human	machine	system	for	...	user
human	0	2.944	0	2.25	...	0
machine	2.944	0	0	2.25	...	0
system	0	0	0	1.15	...	1.84
for	2.25	2.25	1.15	0	...	0
.	.	.	.	.	.	.
.	.	.	.	.	.	.
.	.	.	.	.	.	.
user	0	0	1.84	0	...	0

Some (severe) problems

- Very high dimensional ($|V|$)
- Very sparse
- Grows with the size of the vocabulary
- **Solution:** Use dimensionality reduction (SVD)