

Търсене и извличане на информация. Приложение на дълбоко машинно обучение

Стоян Михов



Лекция 12: Приложения на рекурентните невронни мрежи.
Конволюционни невронни мрежи.

План на лекцията

- 1. Формалности за курса (5 мин)**
2. Многослойни и двупосочни архитектури с рекурентни невронни мрежи (5 мин)
3. Приложения на рекурентните невронни мрежи (10 мин)
4. Конволюционни невронни мрежи (25 мин)
5. Приложения на КНН (20 мин)
6. Разбиване на поддуми (10 мин)
7. Пренапасване, недонапасване и регуларизация (15 мин)

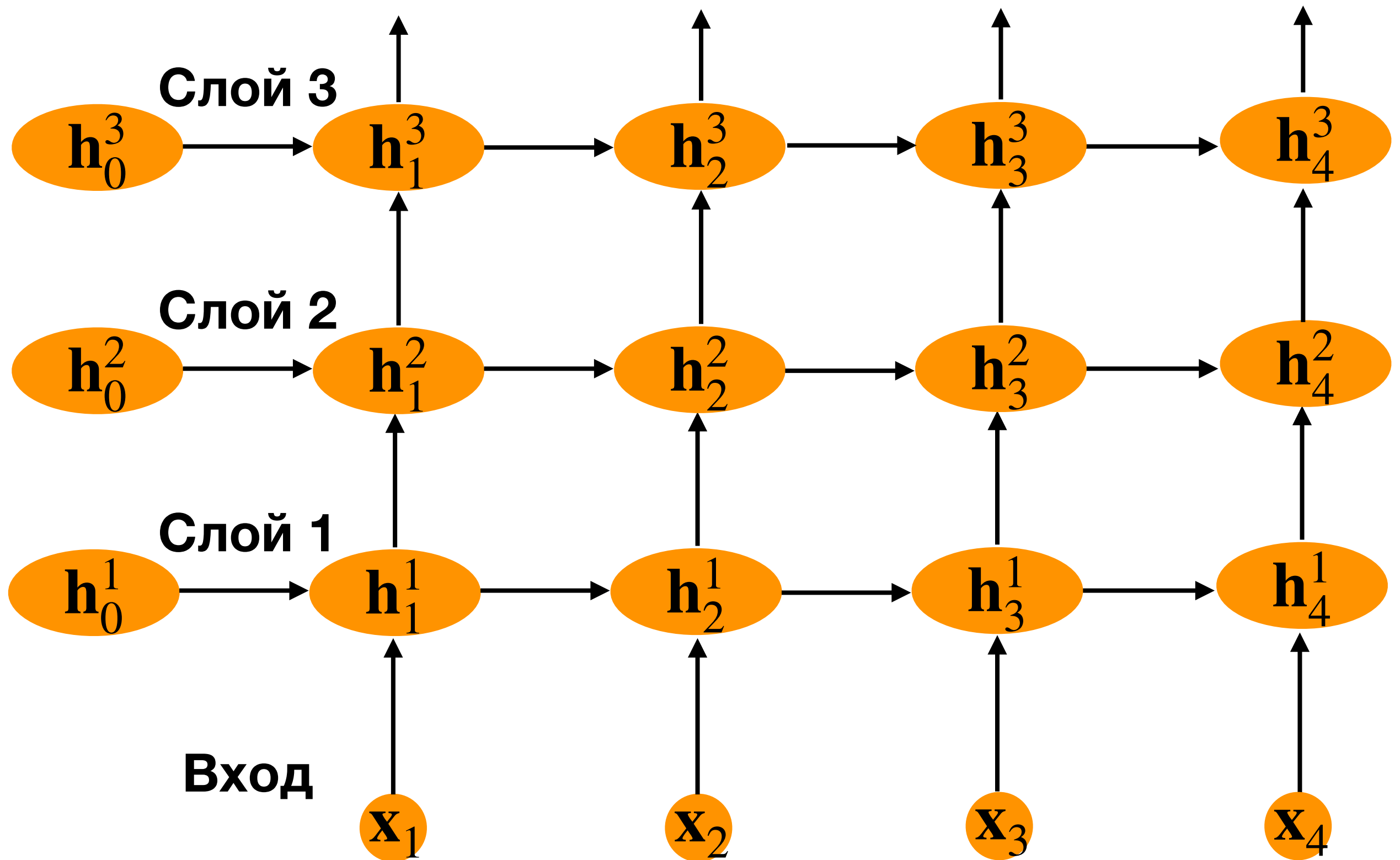
Формалности

- В Moodle тази седмица ще бъде публикувано условието за Домашно задание №2.
- Домашно задание №2 следва да бъде предадено в Moodle до края на деня на 29.12.2024 г.
- Лекция 12 се базира на глава 13 от втория учебник.

План на лекцията

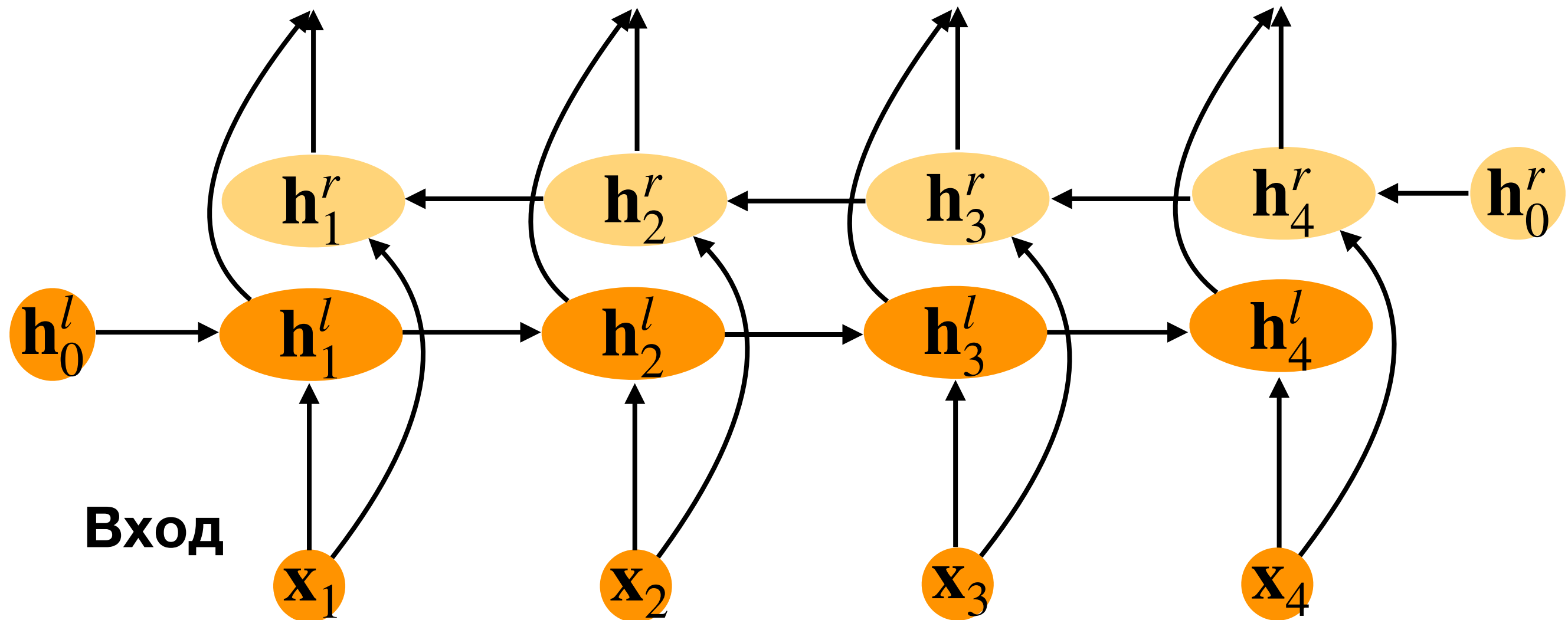
1. Формалности за курса (5 мин)
- 2. Многослойни и двупосочни архитектури с рекурентни невронни мрежи (5 мин)**
3. Приложения на рекурентните невронни мрежи (10 мин)
4. Конволюционни невронни мрежи (25 мин)
5. Приложения на КНН (20 мин)
6. Разбиване на поддуми (10 мин)
7. Пренапасване, недонапасване и регуларизация (15 мин)

Многослойни рекурентни невронни мрежи

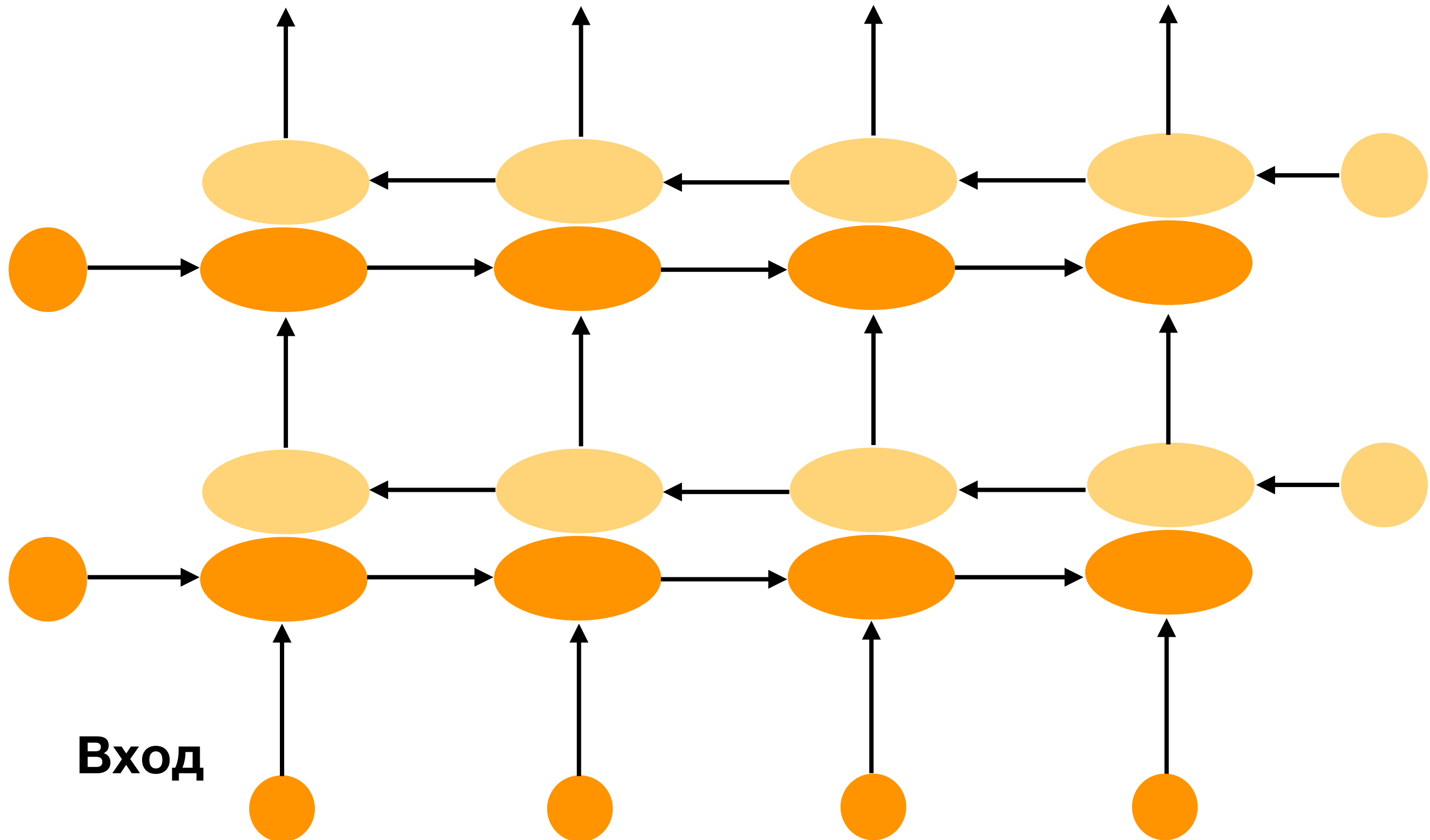


Двупосочни рекурентни невронни мрежи

ИЗХОД



Двупосочни многослойни рекурентни невронни мрежи



План на лекцията

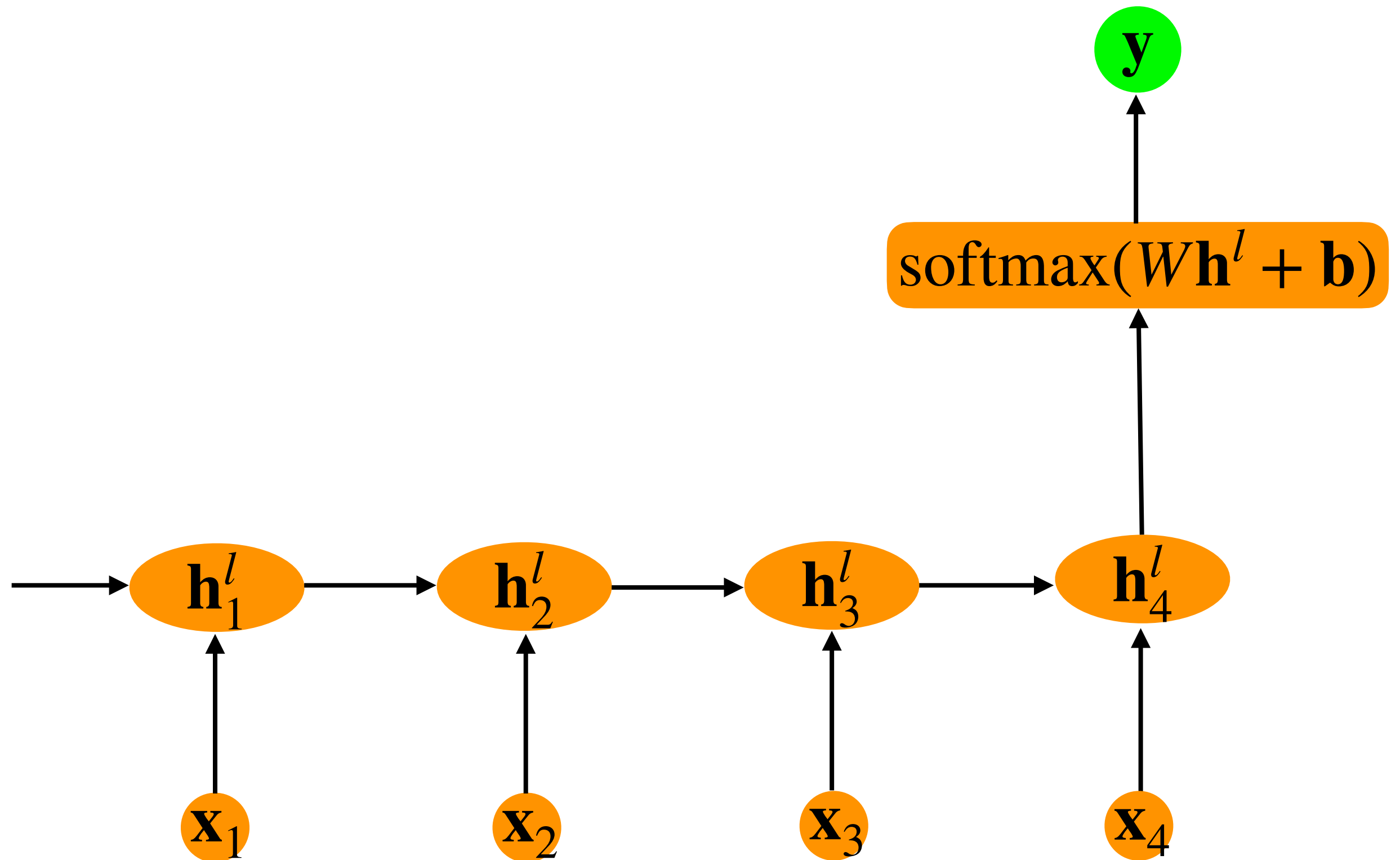
1. Формалности за курса (5 мин)
2. Многослойни и двупосочни архитектури с рекурентни невронни мрежи (5 мин)
- 3. Приложения на рекурентните невронни мрежи (10 мин)**
4. Конволюционни невронни мрежи (25 мин)
5. Приложения на КНН (20 мин)
6. Разбиване на поддуми (10 мин)
7. Пренапасване, недонапасване и регуларизация (15 мин)

Приложение на РНН за езиков модел

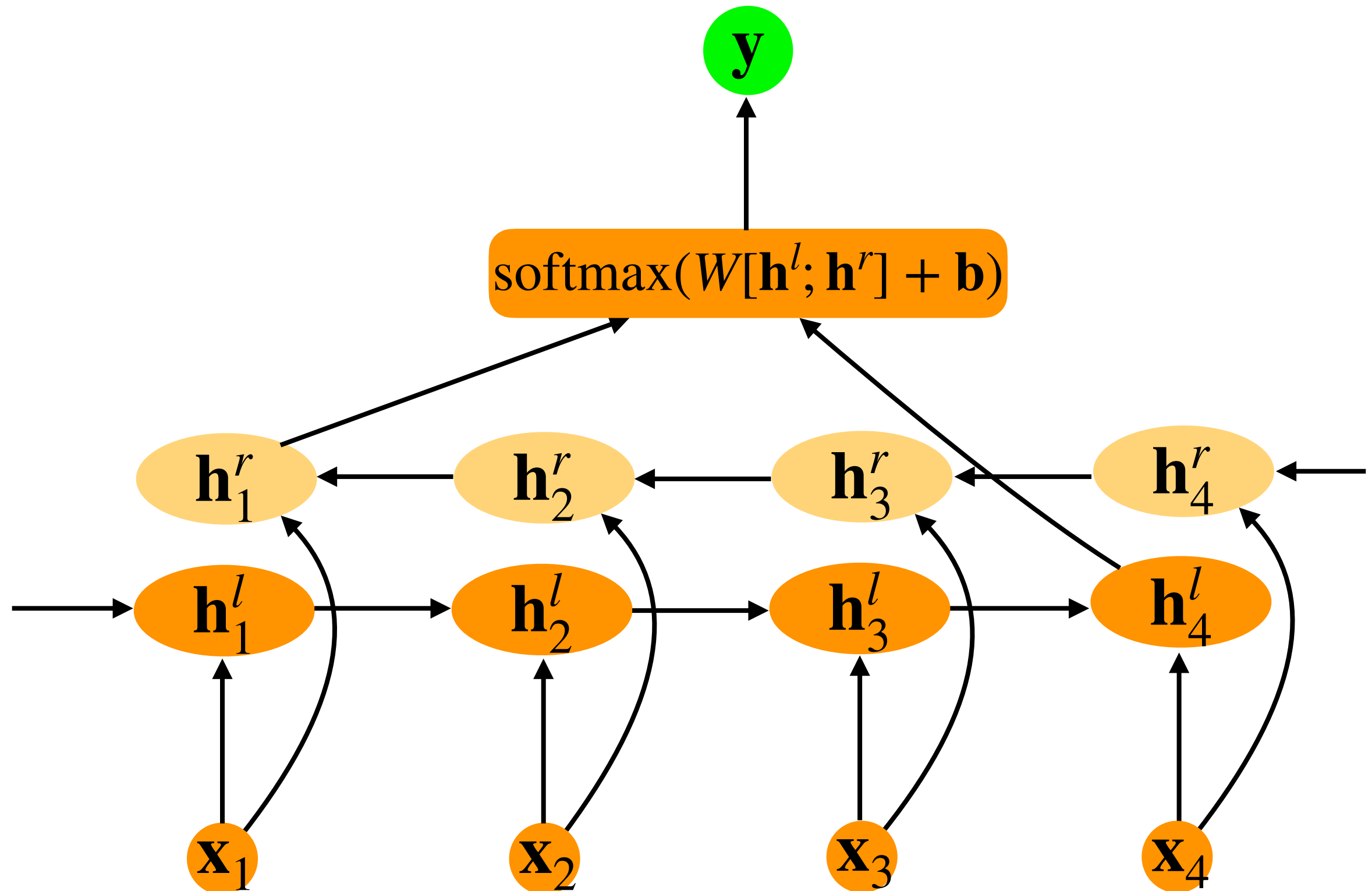
Модел	Перплексия
3-грамен с изглаждане	71
Word2Vec CBOW	56***
EM на Bengio et al.	39
LSTM RNN	32
LSTM Bi-RNN	11***

*** Перплексията изчислена при Word2Vec и двупосочен модел е за локални разпределения, които не определят езиков модел, поради което е некоректно да се сравняват с перплексията на другите модели.

Приложение на РНН за класификация на документи



Приложение на РНН за класификация на документи



Приложение на RNN класификация на документи

Модел	F1
Наивен Бейсов класификатор	89.9
Логистична регресия върху BOW	92.6
LSTM RNN	94.6
LSTM Bi-RNN	96.7

Приложения на езиковите модели

- Предсказване по време на изписване на заявка
- Корекция на текст
- Разпознаване на авторство
- Класификация на документи
- Резюмиране на документи
- Машинен превод
- Разпознаване на реч
- Отговаряне на въпроси
- и много други ...

Приложения при които се налага да се генерира текст

- В някои приложения е необходимо да се генерира текст.
 - Генериране на отговор на въпрос
 - Генериране на резюме за документ
 - Генериране на превод за дадено изречение.
- Бихме могли да използваме езиков модел за генерирането на съответен текст.
- Езиковият модел следва да отразява желания семантичен контекст.

Генериране на текст с езиков модел

- Започваме с текст съдържащ символа за начало или начален текст.
- На всяка стъпка избираме следващата дума случайно, като използваме разпределението за следващата дума при контекста досега, получено от езиковия модел.
- Когато изберем символа за край спираме процедурата.

Генериран текст	Вероятностно разпределение
<START>	
<START> днес	днес $\leftarrow \text{Pr}[X \mid \text{<START>}]$
<START> днес е	е $\leftarrow \text{Pr}[X \mid \text{<START> днес}]$
<START> днес е коледа	коледа $\leftarrow \text{Pr}[X \mid \text{<START> днес е}]$
<START> днес е коледа <STOP>	<STOP> $\leftarrow \text{Pr}[X \mid \text{<START> днес е коледа}]$

Базов алгоритъм за генериране на текст чрез езиков модел

```
generateText(P)
  t <- "<START>"
  w <- [t]
  while not t = "<END>" do
    t <- sample(P(t|w))
    w <- concatenate(w, [t])
  return w
```

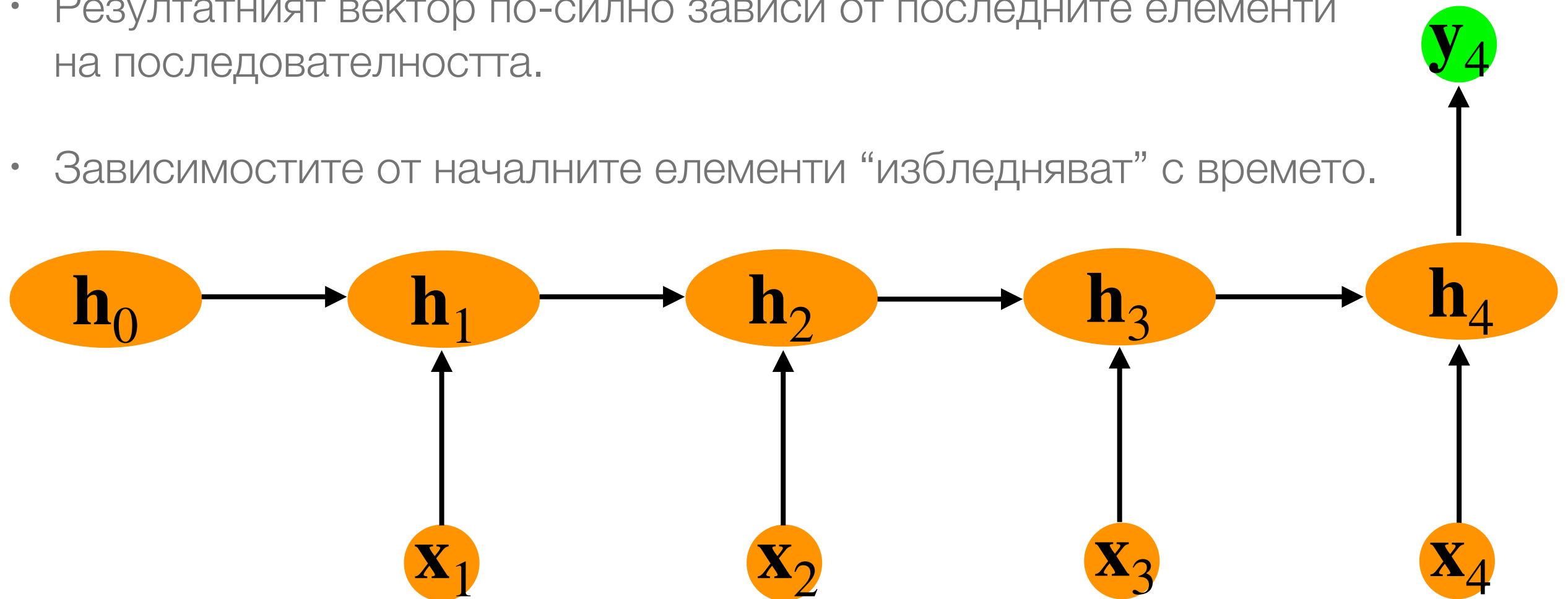
- Невронните езикови модели са мощен универсален инструмент за решаване на разнообразни задачи свързани с търсене, извличане, създаване, систематизиране, класифициране, превеждане и много други.
- Нашумялите в последно време системи за ИИ като ChatGPT, BARD, GEMINI, LLAMA и т.н. са всъщност големи генеративни невронни езикови модели.

План на лекцията

1. Формалности за курса (5 мин)
2. Многослойни и двупосочни архитектури с рекурентни невронни мрежи (5 мин)
3. Приложения на рекурентните невронни мрежи (10 мин)
- 4. Конволюционни невронни мрежи (25 мин)**
5. Приложения на КНН (20 мин)
6. Разбиване на поддуми (10 мин)
7. Пренапасване, недонапасване и регуларизация (15 мин)

Особености на рекурентните невронни мрежи

- На неограничена по дължина последователност съпоставяме вектор с фиксиран размер.
- Резултатът след прочитане на дадена фраза зависи от думите пред нея по различен начин.
- Резултатният вектор по-силно зависи от последните елементи на последователността.
- Зависимостите от началните елементи “избледняват” с времето.



Идея за конволюционни невронни мрежи

- Нека разгледаме всяка подпоследователност с дадена фиксирана дължина поотделно.
- Да обработим подпоследователностите една по една и да запомним съответните резултати.
- Накрая да обобщим получените резултати във фиксиран по размер вектор
- Например за последователността:
да запази възможността за решение между двете държави
- Разглеждаме 6-те подпоследователности с дължина 3:
да запази възможността | запази възможността за | възможността за решение | за решение между | решение между двете | между двете държави

КОНВОЛЮЦИЯ

- Конволюцията е интегрална трансформация, която възниква в различни дялове на математиката. Формалната математическа дефиниция е:

Нека $f, g : \mathbb{R} \rightarrow \mathbb{R}$ са функции. Тогава конволюцията $f * g : \mathbb{R} \rightarrow \mathbb{R}$ е дефинирана като:

$$(f * g)(t) = \int_{-\infty}^{\infty} f(t - \tau)g(\tau) d\tau$$

- Ако f и g са дискретни сигнали и g е дефиниран за $\{0, 1, \dots, M\}$, то конволюцията $f * g$ е:

$$(f * g)[n] = \sum_{m=0}^M f[n - m]g[m]$$

- Ако положим $m' = M - m$, $\tilde{g}[i] = g[M - i]$ и $n' = n - M$ получаваме:

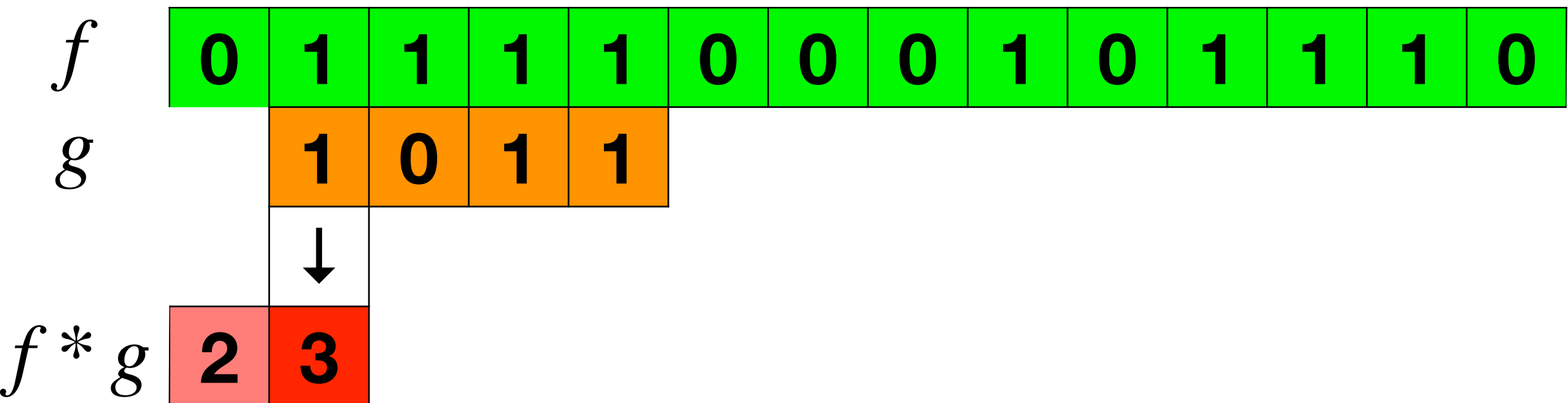
$$(f * g)[n] = \sum_{m'=0}^M f[n' + m']\tilde{g}[m']$$

- Ще наричаме g филтър.
- В цифровата обработката на сигнали и изображения конволюция се използва например за намиране на шаблони.

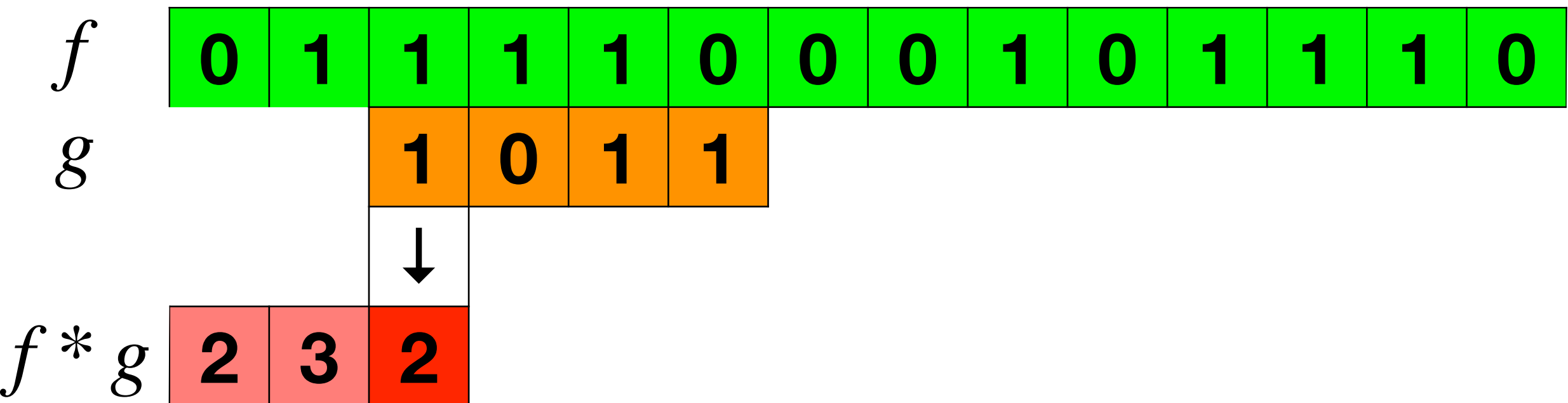
Пример за конволюция

f	0	1	1	1	1	0	0	0	1	0	1	1	1	0
g	1	0	1	1										
	↓													
$f * g$	2													

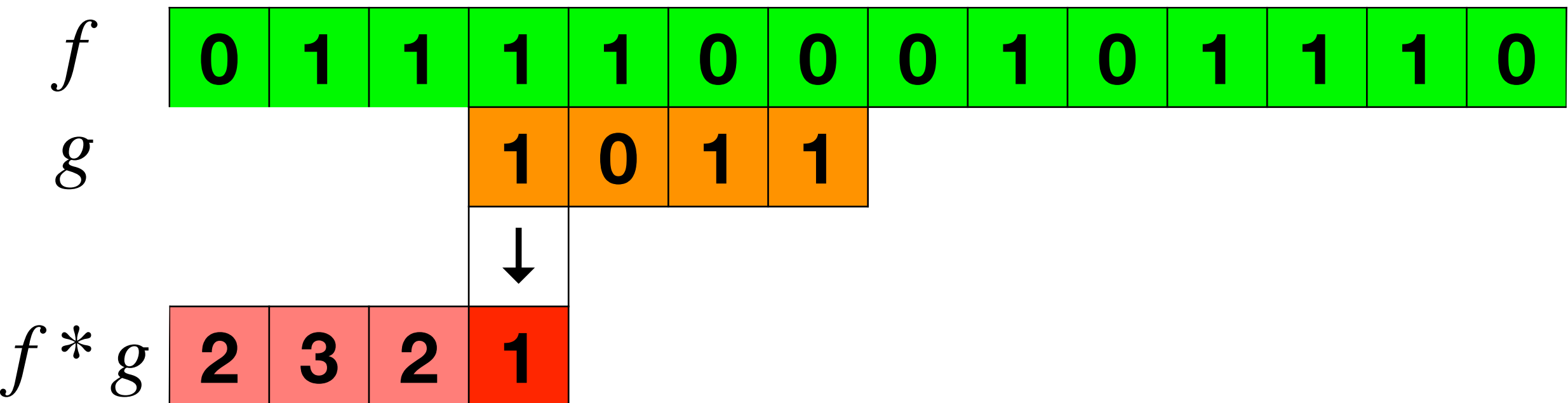
Пример за конволюция



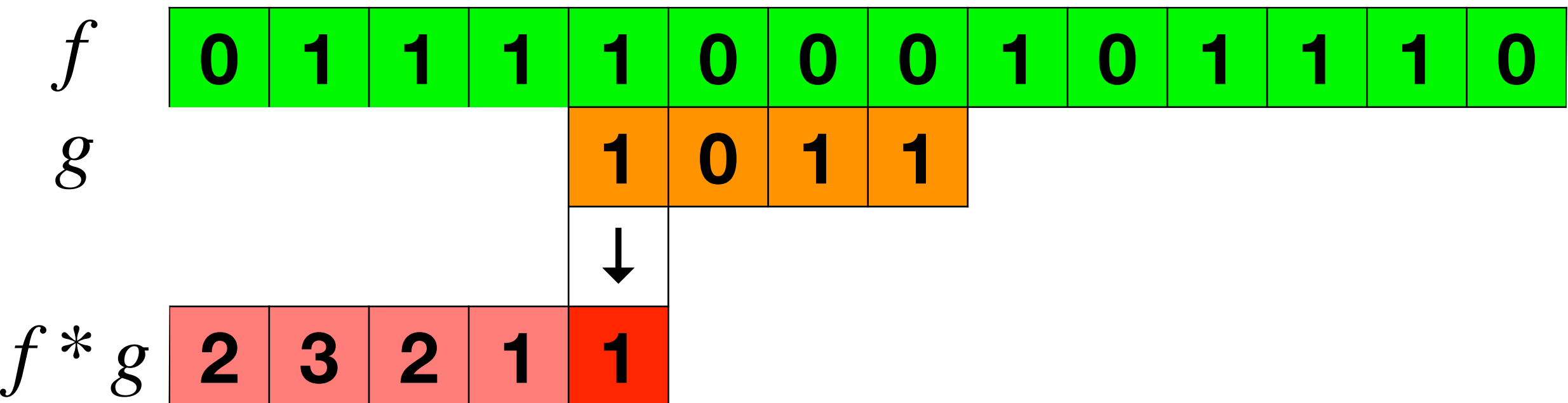
Пример за конволюция



Пример за конволюция



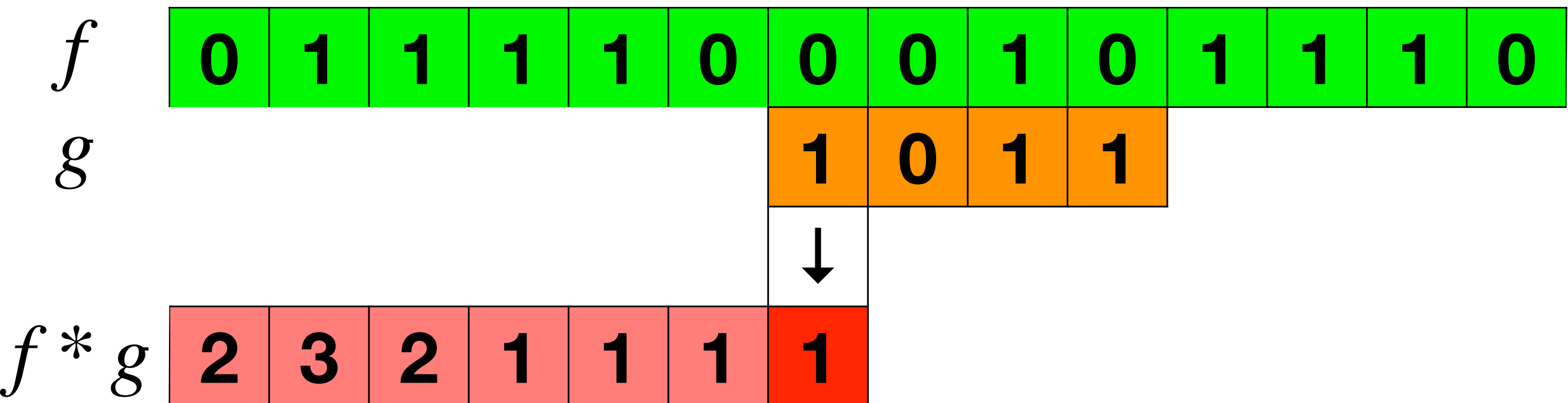
Пример за конволюция



Пример за конволюция

f	0	1	1	1	1	0	0	0	1	0	1	1	1	0
g						1	0	1	1					
						↓								
$f * g$	2	3	2	1	1	1								

Пример за конволюция



Пример за конволюция

f	0	1	1	1	1	0	0	0	1	0	1	1	1	0
g								1	0	1	1			
								↓						
$f * g$	2	3	2	1	1	1	1	1						

Пример за конволюция

f	0	1	1	1	1	0	0	0	1	0	1	1	1	0
g									1	0	1	1		
									↓					
$f * g$	2	3	2	1	1	1	1	1	3					

Пример за конволюция

f	0	1	1	1	1	0	0	0	1	0	1	1	1	0
g										1	0	1	1	
										↓				
$f * g$	2	3	2	1	1	1	1	1	3	2				

Пример за конволюция

f	0	1	1	1	1	0	0	0	1	0	1	1	1	0
g											1	0	1	1
											↓			
$f * g$	2	3	2	1	1	1	1	1	3	2	2			

Пример за конволюция

f

0	1	1	1	1	0	0	0	1	0	1	1	1	0
---	---	---	---	---	---	---	---	---	---	---	---	---	---

$f * g$

2	3	2	1	1	1	1	1	3	2	2
---	---	---	---	---	---	---	---	---	---	---

Пример за двумерна (2D) конволюция

Изображение

1	1	1	0	0
0	1	1	1	0
0	0	1	1	1
0	0	1	1	0
0	1	1	0	0

Филтър

1	0	1
0	1	0
1	0	1

Пример за двумерна (2D) конволюция

Изображение

1 _{x1}	1 _{x0}	1 _{x1}	0	0
0 _{x0}	1 _{x1}	1 _{x0}	1	0
0 _{x1}	0 _{x0}	1 _{x1}	1	1
0	0	1	1	0
0	1	1	0	0

Филтър

1	0	1
0	1	0
1	0	1

Конволюция

4		

Пример за двумерна (2D) конволюция

Изображение

1	1 _{x1}	1 _{x0}	0 _{x1}	0
0	1 _{x0}	1 _{x1}	1 _{x0}	0
0	0 _{x1}	1 _{x0}	1 _{x1}	1
0	0	1	1	0
0	1	1	0	0

Филтър

1	0	1
0	1	0
1	0	1

Конволюция

4	3	

Пример за двумерна (2D) конволюция

Изображение

1	1	1 _{x1}	0 _{x0}	0 _{x1}
0	1	1 _{x0}	1 _{x1}	0 _{x0}
0	0	1 _{x1}	1 _{x0}	1 _{x1}
0	0	1	1	0
0	1	1	0	0

Филтър

1	0	1
0	1	0
1	0	1

Конволюция

4	3	4

Пример за двумерна (2D) конволюция

Изображение

1	1	1	0	0
0 _{x1}	1 _{x0}	1 _{x1}	1	0
0 _{x0}	0 _{x1}	1 _{x0}	1	1
0 _{x1}	0 _{x0}	1 _{x1}	1	0
0	1	1	0	0

Филтър

1	0	1
0	1	0
1	0	1

Конволюция

4	3	4
2		

Пример за двумерна (2D) конволюция

Изображение

1	1	1	0	0
0	1 _{x1}	1 _{x0}	1 _{x1}	0
0	0 _{x0}	1 _{x1}	1 _{x0}	1
0	0 _{x1}	1 _{x0}	1 _{x1}	0
0	1	1	0	0

Филтър

1	0	1
0	1	0
1	0	1

Конволюция

4	3	4
2	4	

Пример за двумерна (2D) конволюция

Изображение

1	1	1	0	0
0	1	1 _{x1}	1 _{x0}	0 _{x1}
0	0	1 _{x0}	1 _{x1}	1 _{x0}
0	0	1 _{x1}	1 _{x0}	0 _{x1}
0	1	1	0	0

Филтър

1	0	1
0	1	0
1	0	1

Конволюция

4	3	4
2	4	3

Пример за двумерна (2D) конволюция

Изображение

1	1	1	0	0
0	1	1	1	0
0 _{x1}	0 _{x0}	1 _{x1}	1	1
0 _{x0}	0 _{x1}	1 _{x0}	1	0
0 _{x1}	1 _{x0}	1 _{x1}	0	0

Филтър

1	0	1
0	1	0
1	0	1

Конволюция

4	3	4
2	4	3
2		

Пример за двумерна (2D) конволюция

Изображение

1	1	1	0	0
0	1	1	1	0
0	0 _{x1}	1 _{x0}	1 _{x1}	1
0	0 _{x0}	1 _{x1}	1 _{x0}	0
0	1 _{x1}	1 _{x0}	0 _{x1}	0

Филтър

1	0	1
0	1	0
1	0	1

Конволюция

4	3	4
2	4	3
2	3	

Пример за двумерна (2D) конволюция

Изображение

1	1	1	0	0
0	1	1	1	0
0	0	1 _{x1}	1 _{x0}	1 _{x1}
0	0	1 _{x0}	1 _{x1}	0 _{x0}
0	1	1 _{x1}	0 _{x0}	0 _{x1}

Филтър

1	0	1
0	1	0
1	0	1

Конволюция

4	3	4
2	4	3
2	3	4

Едномерна (1D) конволюция върху текст

да	-0,30
запази	0,43
възможността	0,27
за	-0,08
решение	0,00
между	-0,25
двете	0,09
държави	-0,06

3
-1
1

Едномерна (1D) конволюция върху текст

много канали

да	-0,30	-0,04	-0,18	0,33
запази	0,43	0,46	-0,40	-0,32
възможността	0,27	-0,03	0,29	0,15
за	-0,08	-0,34	0,10	-0,42
решение	0,00	-0,06	-0,46	0,27
между	-0,25	0,43	-0,05	-0,40
двете	0,09	-0,26	0,12	0,48
държави	-0,06	-0,39	-0,19	0,29

д, з, в	
з, в, з	
в, з, р	
з, р, м	
р, м, д	
м, д, д	

3	1	2	-3
-1	2	1	-3
1	1	-1	1

Едномерна (1D) конволюция върху текст

да	-0,30	-0,04	-0,18	0,33
запази	0,43	0,46	-0,40	-0,32
възможността	0,27	-0,03	0,29	0,15
за	-0,08	-0,34	0,10	-0,42
решение	0,00	-0,06	-0,46	0,27
между	-0,25	0,43	-0,05	-0,40
двете	0,09	-0,26	0,12	0,48
държави	-0,06	-0,39	-0,19	0,29



д, з, в	-1,13
з, в, з	
в, з, р	
з, р, м	
р, м, д	
м, д, д	

3	1	2	-3
-1	2	1	-3
1	1	-1	1

Едномерна (1D) конволюция върху текст

да	-0,30	-0,04	-0,18	0,33
запази	0,43	0,46	-0,40	-0,32
възможността	0,27	-0,03	0,29	0,15
за	-0,08	-0,34	0,10	-0,42
решение	0,00	-0,06	-0,46	0,27
между	-0,25	0,43	-0,05	-0,40
двете	0,09	-0,26	0,12	0,48
държави	-0,06	-0,39	-0,19	0,29



д, з, в	-1,13
з, в, з	0,47
в, з, р	
з, р, м	
р, м, д	
м, д, д	

3	1	2	-3
-1	2	1	-3
1	1	-1	1

Едномерна (1D) конволюция върху текст

да	-0,30	-0,04	-0,18	0,33
запази	0,43	0,46	-0,40	-0,32
възможността	0,27	-0,03	0,29	0,15
за	-0,08	-0,34	0,10	-0,42
решение	0,00	-0,06	-0,46	0,27
между	-0,25	0,43	-0,05	-0,40
двете	0,09	-0,26	0,12	0,48
държави	-0,06	-0,39	-0,19	0,29



д, з, в	-1,13
з, в, з	0,47
в, з, р	2,35
з, р, м	
р, м, д	
м, д, д	

3	1	2	-3
-1	2	1	-3
1	1	-1	1

Едномерна (1D) конволюция върху текст

да	-0,30	-0,04	-0,18	0,33
запази	0,43	0,46	-0,40	-0,32
възможността	0,27	-0,03	0,29	0,15
за	-0,08	-0,34	0,10	-0,42
решение	0,00	-0,06	-0,46	0,27
между	-0,25	0,43	-0,05	-0,40
двете	0,09	-0,26	0,12	0,48
държави	-0,06	-0,39	-0,19	0,29



д, з, в	-1,13
з, в, з	0,47
в, з, р	2,35
з, р, м	-0,65
р, м, д	
м, д, д	

3	1	2	-3
-1	2	1	-3
1	1	-1	1

Едномерна (1D) конволюция върху текст

да	-0,30	-0,04	-0,18	0,33
запази	0,43	0,46	-0,40	-0,32
възможността	0,27	-0,03	0,29	0,15
за	-0,08	-0,34	0,10	-0,42
решение	0,00	-0,06	-0,46	0,27
между	-0,25	0,43	-0,05	-0,40
двете	0,09	-0,26	0,12	0,48
държави	-0,06	-0,39	-0,19	0,29



д, з, в	-1,13
з, в, з	0,47
в, з, р	2,35
з, р, м	-0,65
р, м, д	0,70
м, д, д	

3	1	2	-3
-1	2	1	-3
1	1	-1	1

Едномерна (1D) конволюция върху текст

да	-0,30	-0,04	-0,18	0,33
запази	0,43	0,46	-0,40	-0,32
възможността	0,27	-0,03	0,29	0,15
за	-0,08	-0,34	0,10	-0,42
решение	0,00	-0,06	-0,46	0,27
между	-0,25	0,43	-0,05	-0,40
двете	0,09	-0,26	0,12	0,48
държави	-0,06	-0,39	-0,19	0,29



д, з, в	-1,13
з, в, з	0,47
в, з, р	2,35
з, р, м	-0,65
р, м, д	0,70
м, д, д	-1,09

3	1	2	-3
-1	2	1	-3
1	1	-1	1

Едномерна (1D) конволюция върху текст

да	-0,30	-0,04	-0,18	0,33
запази	0,43	0,46	-0,40	-0,32
възможността	0,27	-0,03	0,29	0,15
за	-0,08	-0,34	0,10	-0,42
решение	0,00	-0,06	-0,46	0,27
между	-0,25	0,43	-0,05	-0,40
двете	0,09	-0,26	0,12	0,48
държави	-0,06	-0,39	-0,19	0,29

д, з, в	-1,13
з, в, з	0,47
в, з, р	2,35
з, р, м	-0,65
р, м, д	0,70
м, д, д	-1,09

3	1	2	-3
-1	2	1	-3
1	1	-1	1

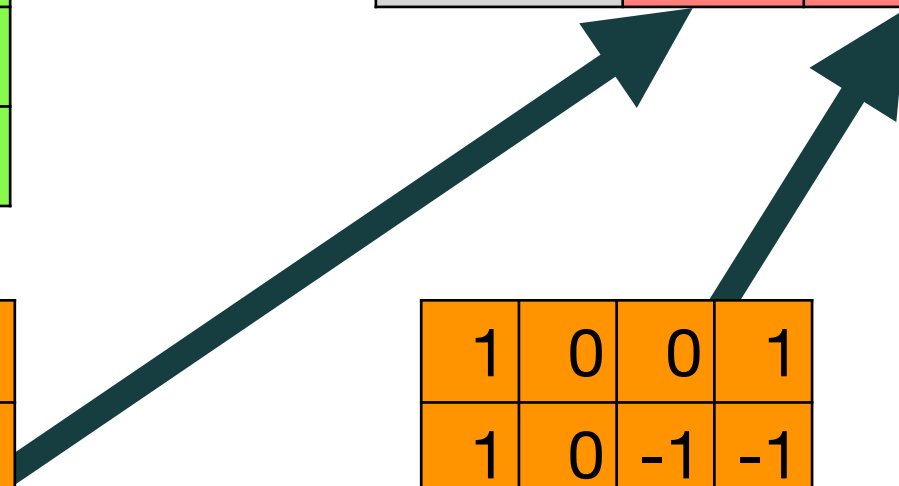
Едномерна (1D) конволюция върху текст много филтри

да	-0,30	-0,04	-0,18	0,33
запази	0,43	0,46	-0,40	-0,32
възможността	0,27	-0,03	0,29	0,15
за	-0,08	-0,34	0,10	-0,42
решение	0,00	-0,06	-0,46	0,27
между	-0,25	0,43	-0,05	-0,40
двете	0,09	-0,26	0,12	0,48
държави	-0,06	-0,39	-0,19	0,29

д, з, в	-1,13	1,31
з, в, з	0,47	-0,82
в, з, р	2,35	0,87
з, р, м	-0,65	-0,28
р, м, д	0,70	0,68
м, д, д	-1,09	-1,26

3	1	2	-3
-1	2	1	-3
1	1	-1	1

1	0	0	1
1	0	-1	-1
0	1	0	1



Едномерна (1D) конволюция върху текст

много филтри

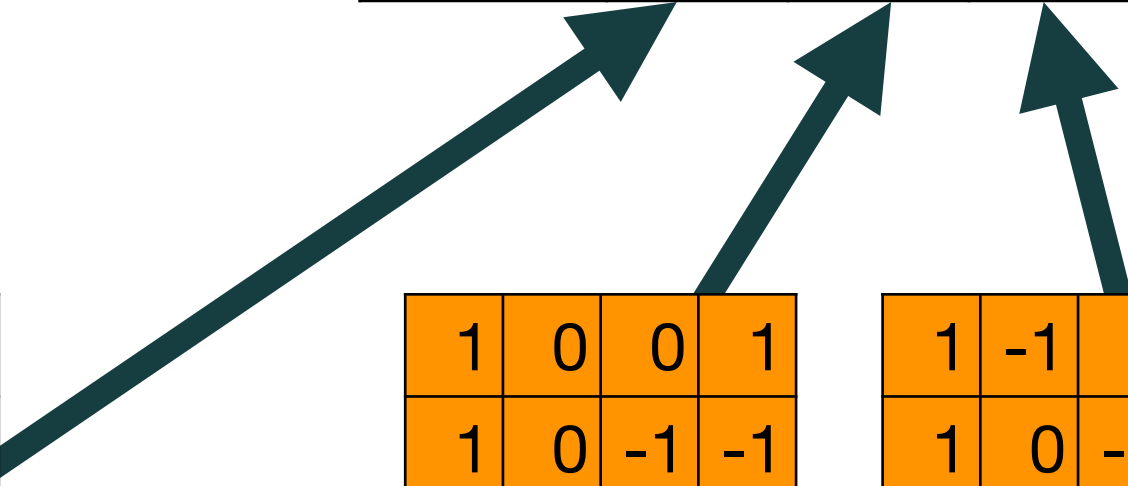
да	-0,30	-0,04	-0,18	0,33
запази	0,43	0,46	-0,40	-0,32
възможността	0,27	-0,03	0,29	0,15
за	-0,08	-0,34	0,10	-0,42
решение	0,00	-0,06	-0,46	0,27
между	-0,25	0,43	-0,05	-0,40
двете	0,09	-0,26	0,12	0,48
държави	-0,06	-0,39	-0,19	0,29

д, з, в	-1,13	1,31	-0,43
з, в, з	0,47	-0,82	-0,94
в, з, р	2,35	0,87	-1,50
з, р, м	-0,65	-0,28	2,51
р, м, д	0,70	0,68	-2,32
м, д, д	-1,09	-1,26	0,15

3	1	2	-3
-1	2	1	-3
1	1	-1	1

1	0	0	1
1	0	-1	-1
0	1	0	1

1	-1	2	-1
1	0	-1	3
0	2	2	1



Едномерна (1D) конволюция върху текст извличане на максимално (max pooling over time)

да	-0,30	-0,04	-0,18	0,33
запази	0,43	0,46	-0,40	-0,32
възможността	0,27	-0,03	0,29	0,15
за	-0,08	-0,34	0,10	-0,42
решение	0,00	-0,06	-0,46	0,27
между	-0,25	0,43	-0,05	-0,40
двете	0,09	-0,26	0,12	0,48
държави	-0,06	-0,39	-0,19	0,29

д, з, в	-1,13	1,31	-0,43
з, в, з	0,47	-0,82	-0,94
в, з, р	2,35	0,87	-1,50
з, р, м	-0,65	-0,28	2,51
р, м, д	0,70	0,68	-2,32
м, д, д	-1,09	-1,26	0,15

max p	2,35	1,31	2,51
-------	------	------	------

3	1	2	-3
-1	2	1	-3
1	1	-1	1

1	0	0	1
1	0	-1	-1
0	1	0	1

1	-1	2	-1
1	0	-1	3
0	2	2	1

Едномерна (1D) конволюция върху текст извличане на k-максимални (k-max pooling over time)

да	-0,30	-0,04	-0,18	0,33
запази	0,43	0,46	-0,40	-0,32
възможността	0,27	-0,03	0,29	0,15
за	-0,08	-0,34	0,10	-0,42
решение	0,00	-0,06	-0,46	0,27
между	-0,25	0,43	-0,05	-0,40
двете	0,09	-0,26	0,12	0,48
държави	-0,06	-0,39	-0,19	0,29

д, з, в	-1,13	1,31	-0,43
з, в, з	0,47	-0,82	-0,94
в, з, р	2,35	0,87	-1,50
з, р, м	-0,65	-0,28	2,51
р, м, д	0,70	0,68	-2,32
м, д, д	-1,09	-1,26	0,15

2-max	2,35	1,31	2,51
	0,70	0,87	0,15

3	1	2	-3
-1	2	1	-3
1	1	-1	1

1	0	0	1
1	0	-1	-1
0	1	0	1

1	-1	2	-1
1	0	-1	3
0	2	2	1

Едномерна (1D) конволюция върху текст извличане на средно (average pooling over time)

да	-0,30	-0,04	-0,18	0,33
запази	0,43	0,46	-0,40	-0,32
възможността	0,27	-0,03	0,29	0,15
за	-0,08	-0,34	0,10	-0,42
решение	0,00	-0,06	-0,46	0,27
между	-0,25	0,43	-0,05	-0,40
двете	0,09	-0,26	0,12	0,48
държави	-0,06	-0,39	-0,19	0,29

д, з, в	-1,13	1,31	-0,43
з, в, з	0,47	-0,82	-0,94
в, з, р	2,35	0,87	-1,50
з, р, м	-0,65	-0,28	2,51
р, м, д	0,70	0,68	-2,32
м, д, д	-1,09	-1,26	0,15

ave p	0,11	0,08	-0.42
--------------	-------------	-------------	--------------

3	1	2	-3
-1	2	1	-3
1	1	-1	1

1	0	0	1
1	0	-1	-1
0	1	0	1

1	-1	2	-1
1	0	-1	3
0	2	2	1

Едномерна (1D) конволюция с попълване (padding=1)

\$	0,00	0,00	0,00	0,00
да	-0,30	-0,04	-0,18	0,33
запази	0,43	0,46	-0,40	-0,32
възможността	0,27	-0,03	0,29	0,15
за	-0,08	-0,34	0,10	-0,42
решение	0,00	-0,06	-0,46	0,27
между	-0,25	0,43	-0,05	-0,40
двете	0,09	-0,26	0,12	0,48
държави	-0,06	-0,39	-0,19	0,29
\$	0,00	0,00	0,00	0,00

\$, д, з	-0,31	0,67	-0,00
д, з, в	-1,13	1,31	-0,43
з, в, з	0,47	-0,82	-0,94
в, з, р	2,35	0,87	-1,50
з, р, м	-0,65	-0,28	2,51
р, м, д	0,70	0,68	-2,32
м, д, д	-1,09	-1,26	0,15
д, д, \$	-1,13	0,41	1,10


3	1	2	-3
-1	2	1	-3
1	1	-1	1

1	0	0	1
1	0	-1	-1
0	1	0	1

1	-1	2	-1
1	0	-1	3
0	2	2	1

Едномерна (1D) конволюция върху текст разкращ=2 (stride=2)

да	-0,30	-0,04	-0,18	0,33
запази	0,43	0,46	-0,40	-0,32
възможността	0,27	-0,03	0,29	0,15
за	-0,08	-0,34	0,10	-0,42
решение	0,00	-0,06	-0,46	0,27
между	-0,25	0,43	-0,05	-0,40
двете	0,09	-0,26	0,12	0,48
държави	-0,06	-0,39	-0,19	0,29



д, з, в	-1,13	1,31	-0,43
в, з, р			
р, м, д			

3	1	2	-3
-1	2	1	-3
1	1	-1	1

1	0	0	1
1	0	-1	-1
0	1	0	1

1	-1	2	-1
1	0	-1	3
0	2	2	1

Едномерна (1D) конволюция върху текст разкращ=2 (stride=2)

да	-0,30	-0,04	-0,18	0,33
запази	0,43	0,46	-0,40	-0,32
възможността	0,27	-0,03	0,29	0,15
за	-0,08	-0,34	0,10	-0,42
решение	0,00	-0,06	-0,46	0,27
между	-0,25	0,43	-0,05	-0,40
двете	0,09	-0,26	0,12	0,48
държави	-0,06	-0,39	-0,19	0,29

д, з, в	-1,13	1,31	-0,43
в, з, р	2,35	0,87	-1,50
р, м, д			

3	1	2	-3
-1	2	1	-3
1	1	-1	1

1	0	0	1
1	0	-1	-1
0	1	0	1

1	-1	2	-1
1	0	-1	3
0	2	2	1

Едномерна (1D) конволюция върху текст разкращ=2 (stride=2)

да	-0,30	-0,04	-0,18	0,33	
запази	0,43	0,46	-0,40	-0,32	
възможността	0,27	-0,03	0,29	0,15	
за	-0,08	-0,34	0,10	-0,42	
решение	0,00	-0,06	-0,46	0,27	
между	-0,25	0,43	-0,05	-0,40	
двете	0,09	-0,26	0,12	0,48	
държави	-0,06	-0,39	-0,19	0,29	

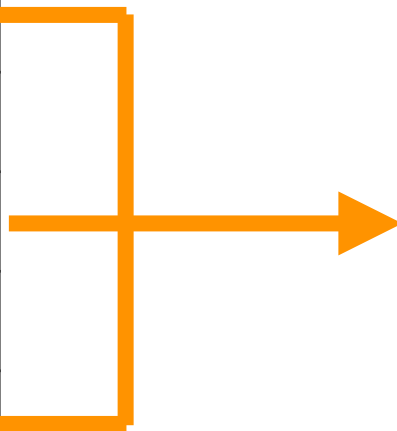
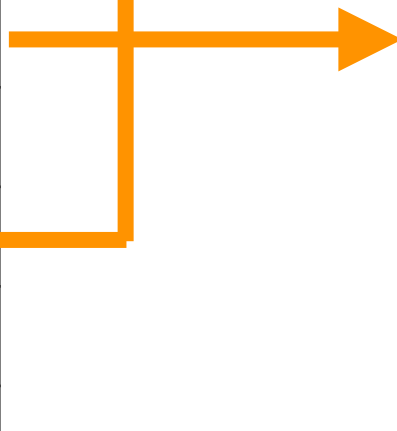
д, з, в	-1,13	1,31	-0,43
в, з, р	2,35	0,87	-1,50
р, м, д	0,70	0,68	-2,32

3	1	2	-3
-1	2	1	-3
1	1	-1	1

1	0	0	1
1	0	-1	-1
0	1	0	1

1	-1	2	-1
1	0	-1	3
0	2	2	1

Едномерна (1D) конволюция върху текст разширяване = 2 (dilation = 2)

да	-0,30	-0,04	-0,18	0,33	
запази	0,43	0,46	-0,40	-0,32	
възможността	0,27	-0,03	0,29	0,15	
за	-0,08	-0,34	0,10	-0,42	
решение	0,00	-0,06	-0,46	0,27	
между	-0,25	0,43	-0,05	-0,40	
двете	0,09	-0,26	0,12	0,48	
държави	-0,06	-0,39	-0,19	0,29	

д, в, р	-2,14	0,07	-1,28
з, з, м			
в, р, д			
з, м, д			

3	1	2	-3
-1	2	1	-3
1	1	-1	1

1	0	0	1
1	0	-1	-1
0	1	0	1

1	-1	2	-1
1	0	-1	3
0	2	2	1

Едномерна (1D) конволюция върху текст разширяване = 2 (dilation = 2)

да	-0,30	-0,04	-0,18	0,33
запази	0,43	0,46	-0,40	-0,32
възможността	0,27	-0,03	0,29	0,15
за	-0,08	-0,34	0,10	-0,42
решение	0,00	-0,06	-0,46	0,27
между	-0,25	0,43	-0,05	-0,40
двете	0,09	-0,26	0,12	0,48
държави	-0,06	-0,39	-0,19	0,29

д, в, р	-2,14	0,07	-1,28
з, з, м	2,52	0,37	-1,58
в, р, д			
з, м, д			

3	1	2	-3
-1	2	1	-3
1	1	-1	1

1	0	0	1
1	0	-1	-1
0	1	0	1

1	-1	2	-1
1	0	-1	3
0	2	2	1

Едномерна (1D) конволюция върху текст разширяване = 2 (dilation = 2)

да	-0,30	-0,04	-0,18	0,33
запази	0,43	0,46	-0,40	-0,32
възможността	0,27	-0,03	0,29	0,15
за	-0,08	-0,34	0,10	-0,42
решение	0,00	-0,06	-0,46	0,27
между	-0,25	0,43	-0,05	-0,40
двете	0,09	-0,26	0,12	0,48
държави	-0,06	-0,39	-0,19	0,29

д, в, р	-2,14	0,07	-1,28
з, з, м	2,52	0,37	-1,58
в, р, д	-0,28	0,84	2,18
з, м, д			

3	1	2	-3
-1	2	1	-3
1	1	-1	1

1	0	0	1
1	0	-1	-1
0	1	0	1

1	-1	2	-1
1	0	-1	3
0	2	2	1

Едномерна (1D) конволюция върху текст разширяване = 2 (dilation = 2)

да	-0,30	-0,04	-0,18	0,33
запази	0,43	0,46	-0,40	-0,32
възможността	0,27	-0,03	0,29	0,15
за	-0,08	-0,34	0,10	-0,42
решение	0,00	-0,06	-0,46	0,27
между	-0,25	0,43	-0,05	-0,40
двете	0,09	-0,26	0,12	0,48
държави	-0,06	-0,39	-0,19	0,29

д, в, р	-2,14	0,07	-1,28
з, з, м	2,52	0,37	-1,58
в, р, д	-0,28	0,84	2,18
з, м, д	3,21	-0,40	-1,39

3	1	2	-3
-1	2	1	-3
1	1	-1	1

1	0	0	1
1	0	-1	-1
0	1	0	1

1	-1	2	-1
1	0	-1	3
0	2	2	1

Едномерна (1D) конволюция върху текст разширяване = 2 (dilation = 2)

да	-0,30	-0,04	-0,18	0,33
запази	0,43	0,46	-0,40	-0,32
възможността	0,27	-0,03	0,29	0,15
за	-0,08	-0,34	0,10	-0,42
решение	0,00	-0,06	-0,46	0,27
между	-0,25	0,43	-0,05	-0,40
двете	0,09	-0,26	0,12	0,48
държави	-0,06	-0,39	-0,19	0,29

д, в, р	-2,14	0,07	-1,28
з, з, м	2,52	0,37	-1,58
в, р, д	-0,28	0,84	2,18
з, м, д	3,21	-0,40	-1,39

3	1	2	-3
-1	2	1	-3
1	1	-1	1

1	0	0	1
1	0	-1	-1
0	1	0	1

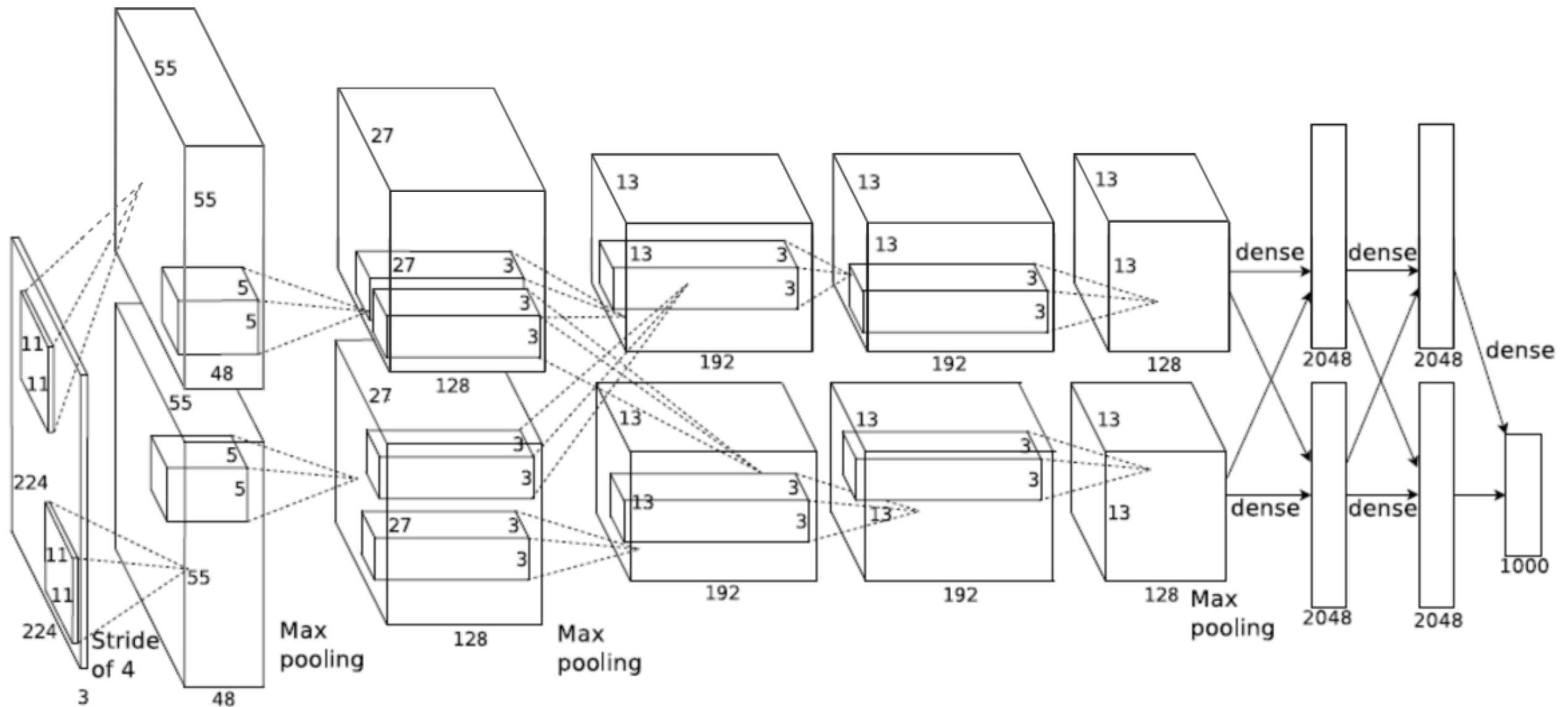
1	-1	2	-1
1	0	-1	3
0	2	2	1

План на лекцията

1. Формалности за курса (5 мин)
2. Многослойни и двупосочни архитектури с рекурентни невронни мрежи (5 мин)
3. Приложения на рекурентните невронни мрежи (10 мин)
4. Конволюционни невронни мрежи (25 мин)
- 5. Приложения на КНН (20 мин)**
6. Разбиване на поддуми (10 мин)
7. Пренапасване, недонапасване и регуларизация (15 мин)

ImageNet

Разпознаване на образи с дълбоки КНН



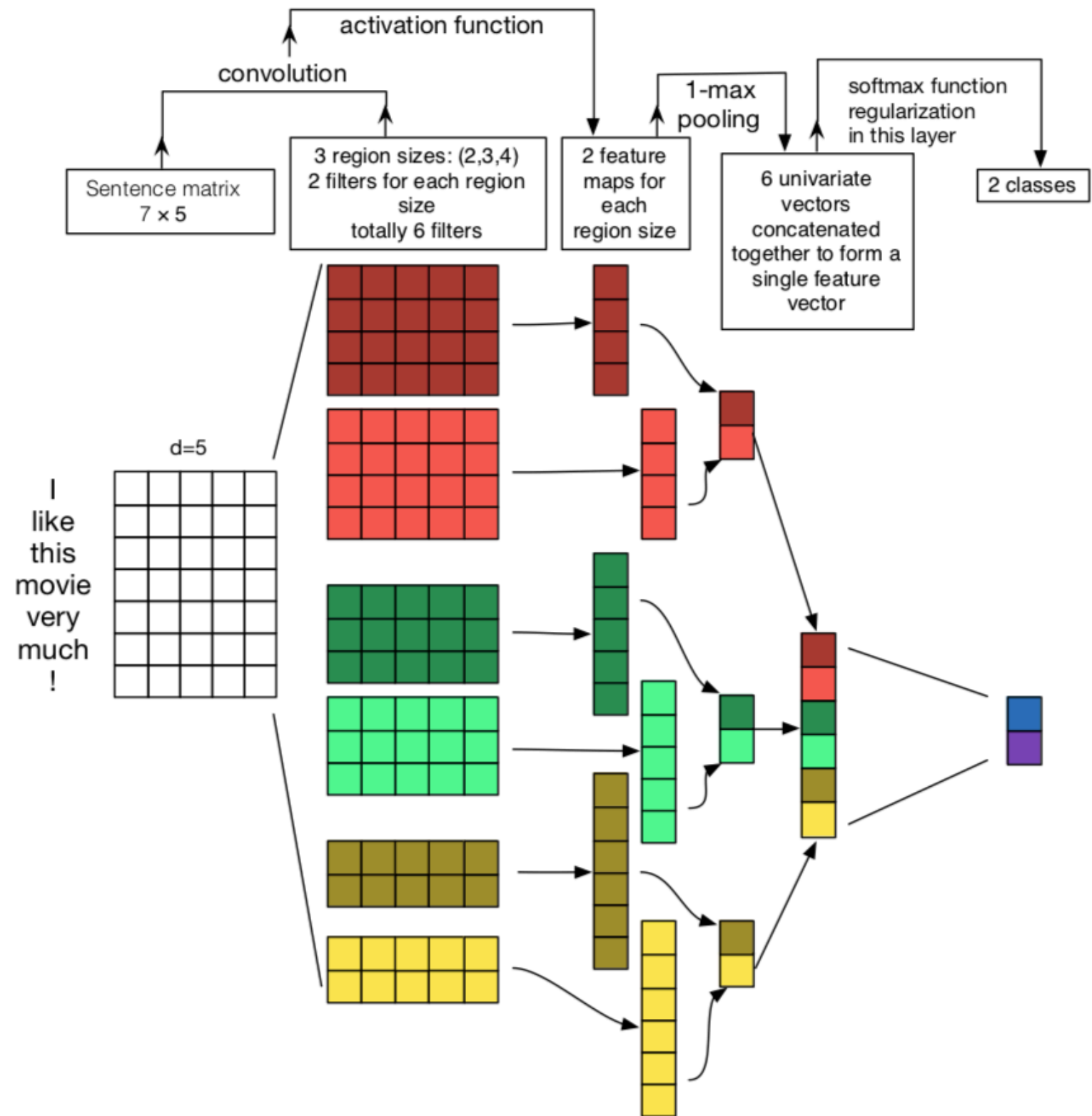
Krizhevsky, Sutskever, and Hinton (2012). ImageNet classification with deep convolutional neural networks.

<http://www.cs.toronto.edu/~hinton/absps/imagenet.pdf>

Приложение на КНН за класифициране на документи

Zhang and Wallace
(2015): A Sensitivity
Analysis of (and
Practitioners' Guide to)
Convolutional Neural
Networks for Sentence
Classification

<https://arxiv.org/pdf/1510.03820.pdf>



Приложение на КНН за класифициране на документи

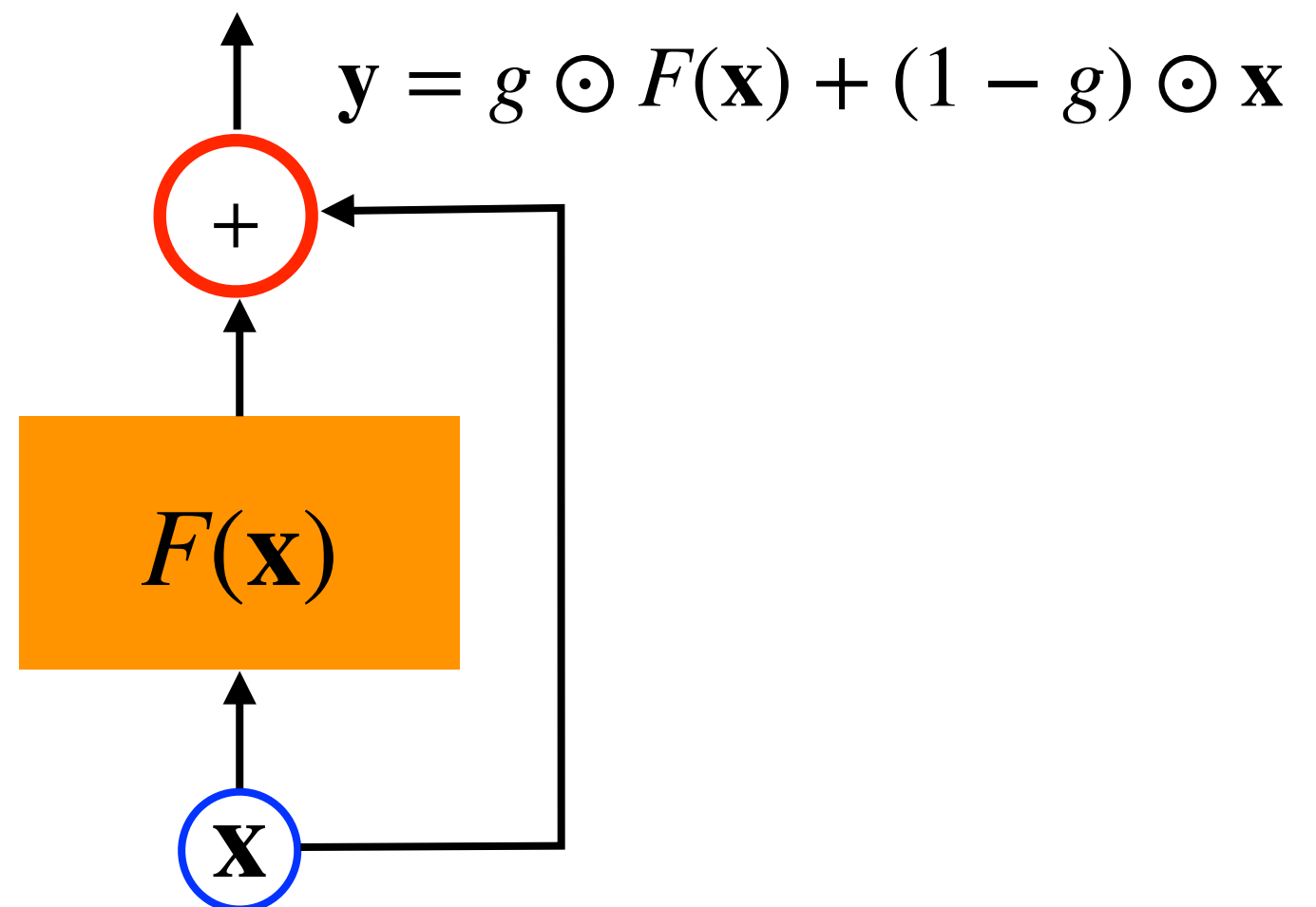
Модел	F1
Наивен Бейсов класификатор	89.9
Логистична регресия върху BOW	92.6
LSTM RNN	94.6
LSTM Bi-RNN	96.7
Конволюционна НМ	97.5

Невронни архитектури с вертикални порти (Highway NN)

- Използването на преки връзки с порти за контрол на пропагирането, което видяхме в LSTM и GRU, е много по-общ метод.
- Този метод има ключово значение при реализирането на невронни мрежи с голяма дълбочина.

Srivastava, Greff and Schmidhuber (2015):
Training Very Deep Networks

<https://proceedings.neurips.cc/paper/2015/file/215a71a12769b056c3c32e7299f1c5ed-Paper.pdf>

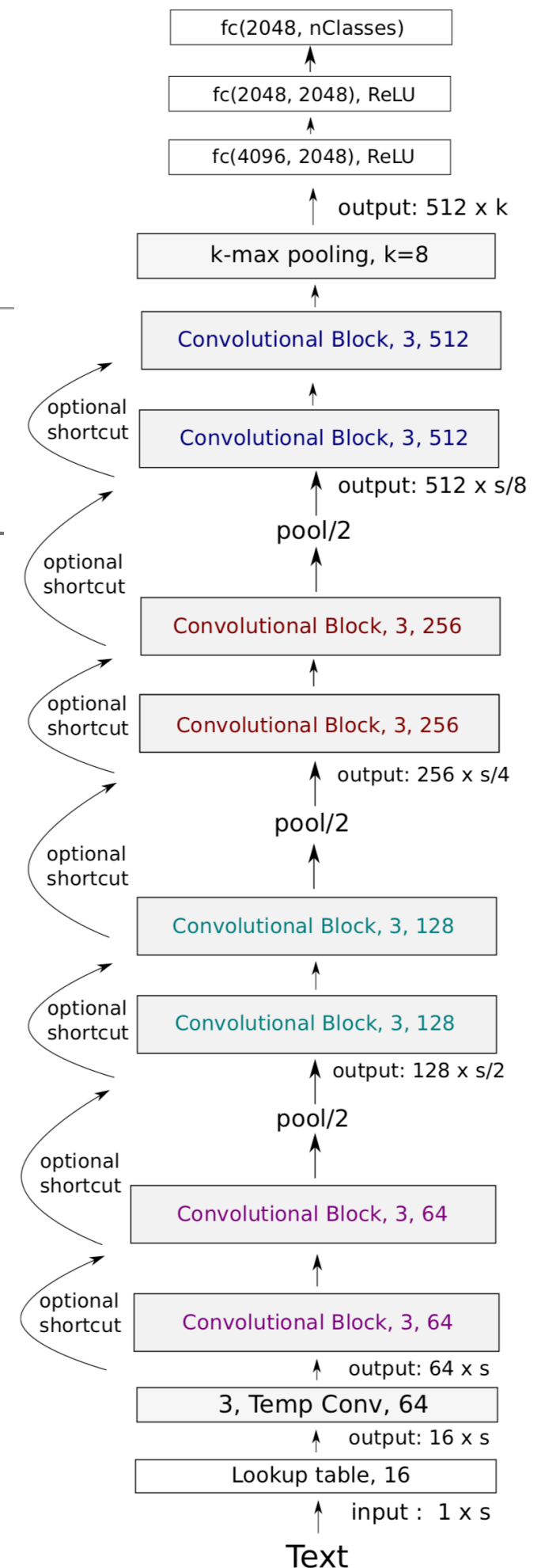


Приложение на многослойни конволюционни НН за класифициране на документи

- Мрежата обработва текста на ниво СИМВОЛИ
- Използва се архитектура от множество слоеве от конволюция и извличане на максимален елемент за влагане на документи
- За да работи тази архитектура е от съществено значение използването на преки връзки

Conneau, Schwenk, Barrault and Yann Lecun (2016):
Very Deep Convolutional Networks for Text
Classification

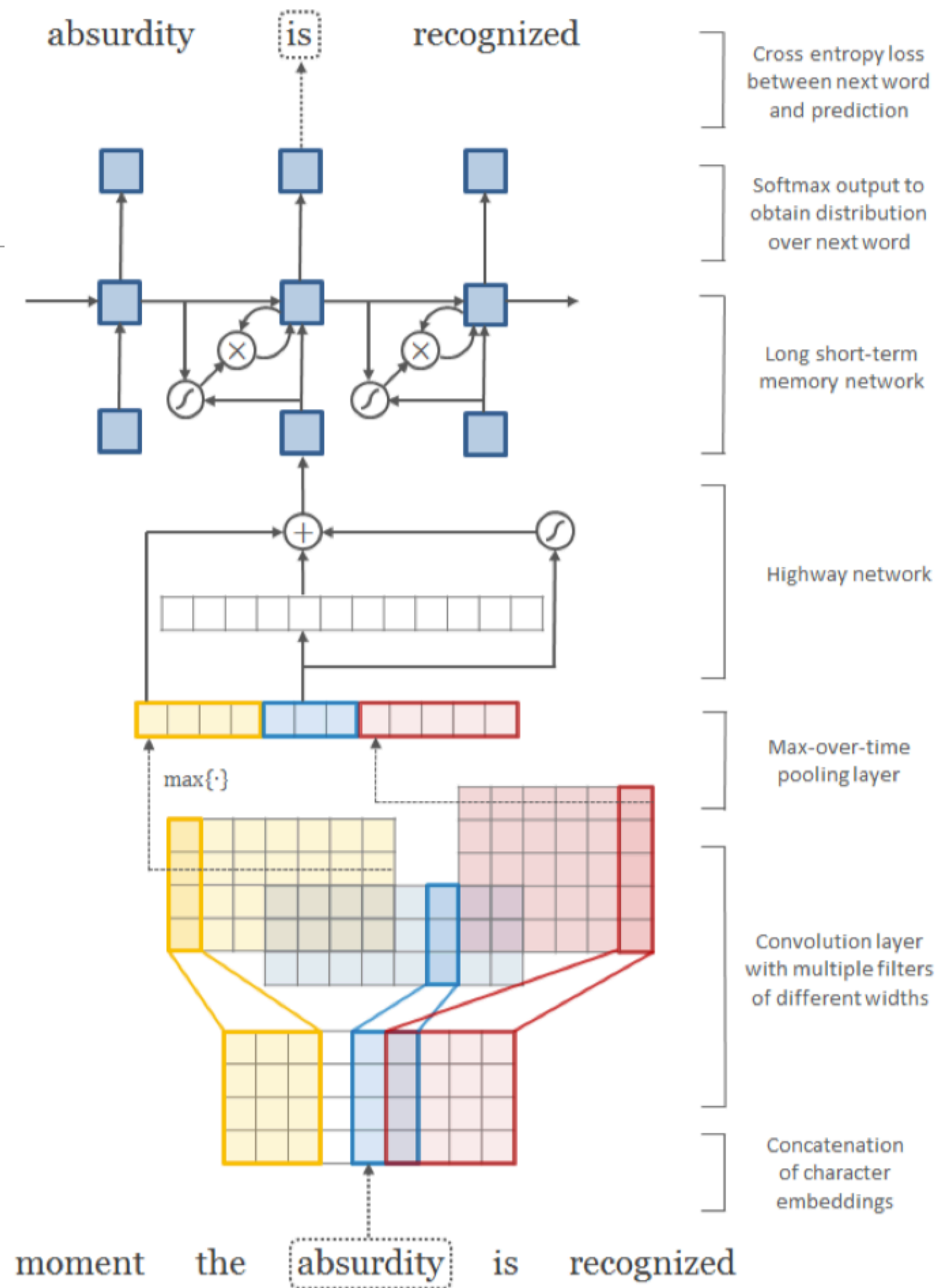
<https://arxiv.org/abs/1606.01781>



Приложение на КНН за ПОСИМВОЛОВ ЕЗИКОВ МОДЕЛ

Kim, Jernite, Sontag and Rush
(2016): Character-Aware
Neural Language Models

[https://arxiv.org/abs/
1508.06615](https://arxiv.org/abs/1508.06615)



Приложение на КНН за посимволов езиков модел

Модел	Перплексия
3-грамен с изглаждане	71
Word2Vec CBOW	56***
EM на Bengio et al.	39
LSTM	32
Симв влагане + LSTM	23
Bi-LSTM	11***
Симв влагане + Bi-LSTM	8.2***

*** Перплексията изчислена при Word2Vec и двупосочен модел е за локални разпределения, които не определят езиков модел, поради което е некоректно да се сравнява с перплексията на другите модели.

План на лекцията

1. Формалности за курса (5 мин)
2. Многослойни и двупосочни архитектури с рекурентни невронни мрежи (5 мин)
3. Приложения на рекурентните невронни мрежи (10 мин)
4. Конволюционни невронни мрежи (25 мин)
5. Приложения на КНН (20 мин)
- 6. Разбиване на поддуми (10 мин)**
7. Пренапасване, недонапасване и регуларизация (15 мин)

Следствия при различните резолюции на разбиване на текста (text tokenization)

- Разбиване на текста на думи:
 - Често се срещат думи извън речника
 - Много голям речник
 - Изчислително скъп softmax
- Разбиване на текста на символи:
 - Решава проблема с думите извън речника
 - Сравнително дълги последователности — затруднява намирането на зависимости на по-големи разстояния
 - По-бавна генерация

Решение: Разбиване на поддуми

- Идея: да се комбинират преимуществата на двата подхода — текста да се разбие на ограничен брой поддуми.
- Тези поддуми могат да съответстват на морфеми, срички, цели думи или отделни символи.
- Поддумите следва да се подберат така че:
 - да се срещат често;
 - да може да се композира всяка дума.

Byte Pair Encoding (BPE)

```
import re, collections

def get_stats(vocab):
    pairs = collections.defaultdict(int)
    for word, freq in vocab.items():
        symbols = word.split()
        for i in range(len(symbols)-1):
            pairs[symbols[i], symbols[i+1]] += freq
    return pairs

def merge_vocab(pair, v_in):
    v_out = {}
    bigram_pattern = re.escape(' '.join(pair))
    p = re.compile(r'(?!\S)' + bigram_pattern + r'(?!\S)')
    for word in v_in:
        w_out = p.sub(' '.join(pair), word)
        v_out[w_out] = v_in[word]
    return v_out

vocab = {'l o w</w>' : 5, 'l o w e r</w>' : 2,
         'n e w e s t</w>' : 6, 'w i d e s t</w>' : 3}
num_merges = 10
for i in range(num_merges):
    pairs = get_stats(vocab)
    best = max(pairs, key=pairs.get)
    vocab = merge_vocab(best, vocab)
    print(best, vocab)
```

('e', 's')	{ 'l o w</w>' : 5, 'l o w e r</w>' : 2, 'n e w e s t</w>' : 6, 'w i d e s t</w>' : 3 }
('es', 't</w>')	{ 'l o w</w>' : 5, 'l o w e r</w>' : 2, 'n e w e s t</w>' : 6, 'w i d e s t</w>' : 3 }
('l', 'o')	{ 'l o w</w>' : 5, 'l o w e r</w>' : 2, 'n e w e s t</w>' : 6, 'w i d e s t</w>' : 3 }
('n', 'e')	{ 'l o w</w>' : 5, 'l o w e r</w>' : 2, 'n e w e s t</w>' : 6, 'w i d e s t</w>' : 3 }
('ne', 'w')	{ 'l o w</w>' : 5, 'l o w e r</w>' : 2, 'n e w e s t</w>' : 6, 'w i d e s t</w>' : 3 }
('new', 'est</w>')	{ 'l o w</w>' : 5, 'l o w e r</w>' : 2, 'n e w e s t</w>' : 6, 'w i d e s t</w>' : 3 }
('lo', 'w</w>')	{ 'l o w</w>' : 5, 'l o w e r</w>' : 2, 'n e w e s t</w>' : 6, 'w i d e s t</w>' : 3 }
('w', 'i')	{ 'l o w</w>' : 5, 'l o w e r</w>' : 2, 'n e w e s t</w>' : 6, 'w i d e s t</w>' : 3 }
('wi', 'd')	{ 'l o w</w>' : 5, 'l o w e r</w>' : 2, 'n e w e s t</w>' : 6, 'w i d e s t</w>' : 3 }
('wid', 'est</w>')	{ 'l o w</w>' : 5, 'l o w e r</w>' : 2, 'n e w e s t</w>' : 6, 'w i d e s t</w>' : 3 }

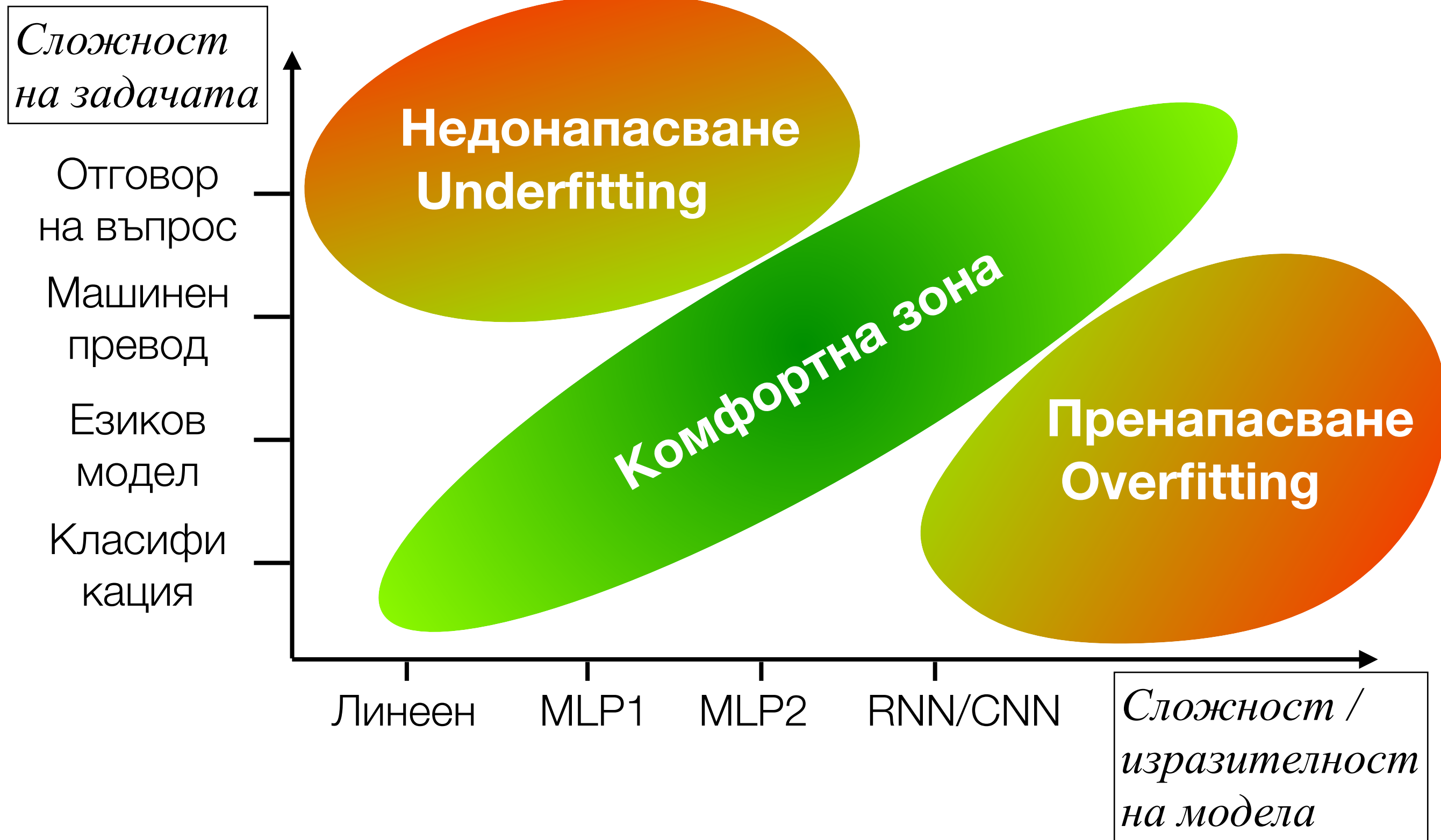
Методи за разбиване на текст на поддуми

- Byte Pair Encoding (BPE):
Sennrich, R., Haddow, B., & Birch, A. (2015). Neural machine translation of rare words with subword units. arXiv preprint arXiv:1508.07909.
- BPE with Dropout:
Provilkov, I., Emelianenko, D., & Voita, E. (2019). BPE-dropout: Simple and effective subword regularization. arXiv preprint arXiv:1910.13267.
- WordPiece:
Kudo, T. (2018). Subword regularization: Improving neural network translation models with multiple subword candidates. arXiv preprint arXiv:1804.10959.

План на лекцията

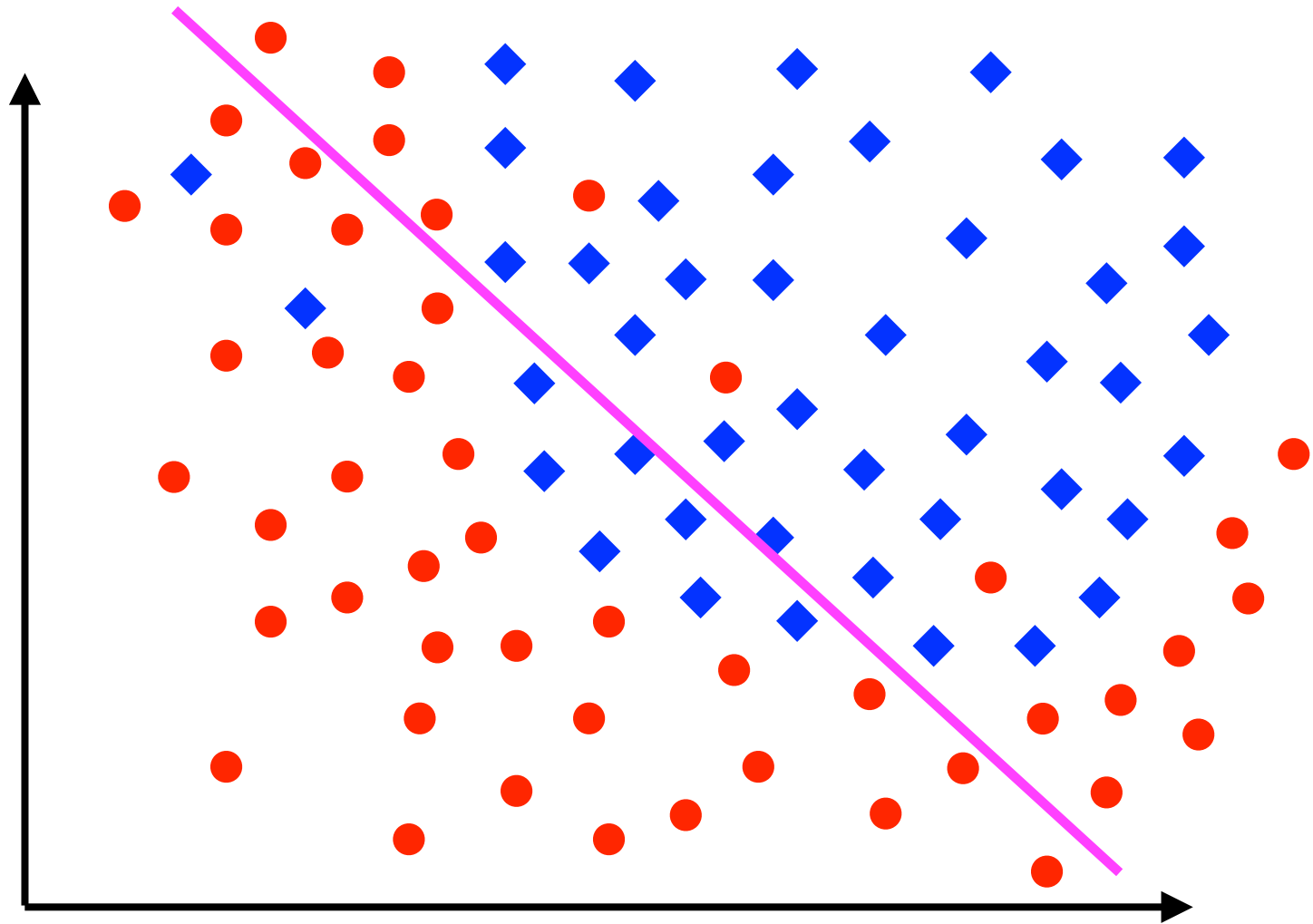
1. Формалности за курса (5 мин)
2. Многослойни и двупосочни архитектури с рекурентни невронни мрежи (5 мин)
3. Приложения на рекурентните невронни мрежи (10 мин)
4. Конволюционни невронни мрежи (25 мин)
5. Приложения на КНН (20 мин)
6. Разбиване на поддуми (10 мин)
- 7. Пренапасване, недонапасване и регуларизация (15 мин)**

Проблеми свързани със сложността на модела



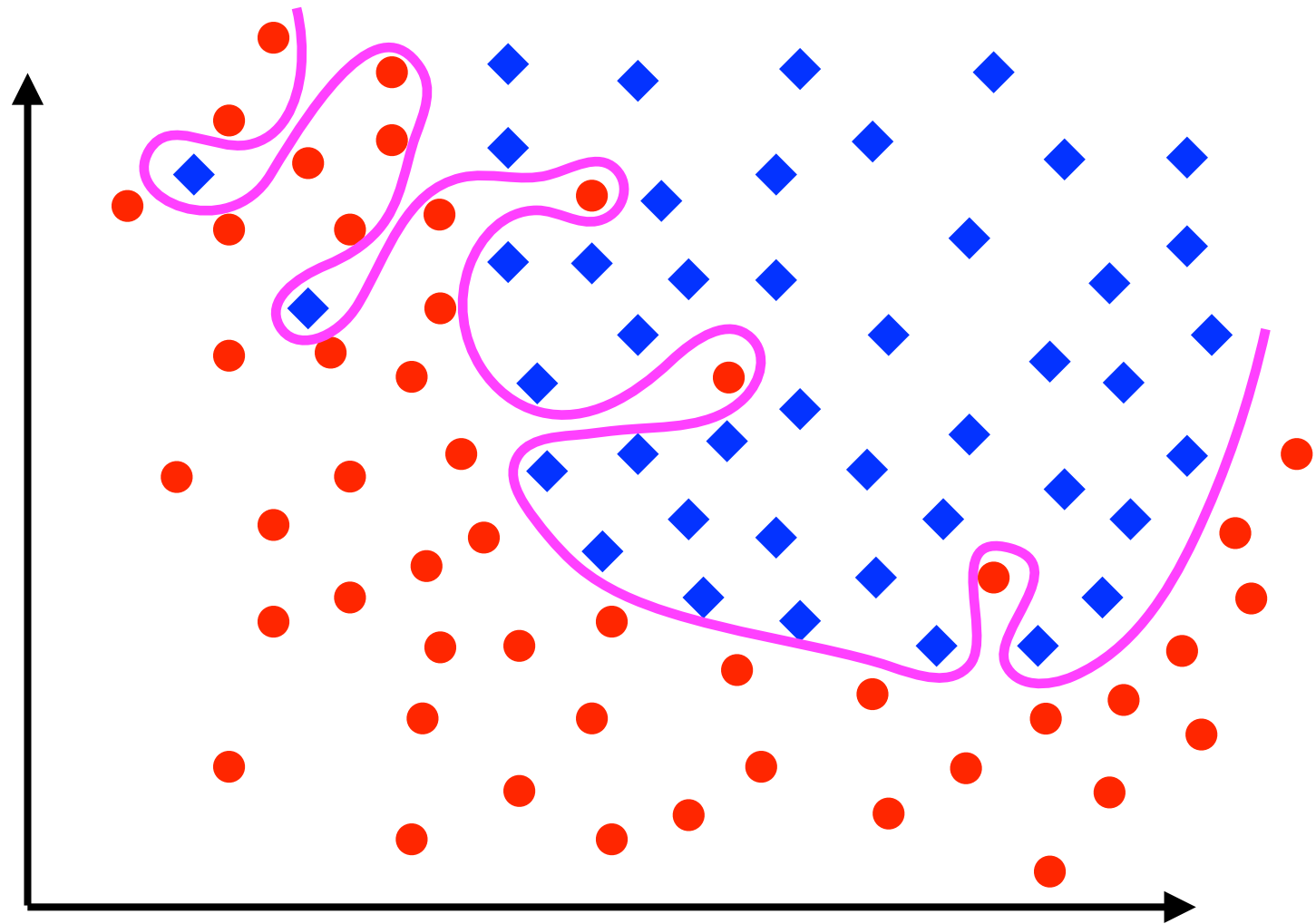
Проблеми свързани при недонапасване

- При недонапасване — модела не е достатъчно изразителен за да научи от данните функция, която достатъчно добре да описва данните.
- Решение: Използване на по-изразителен модел



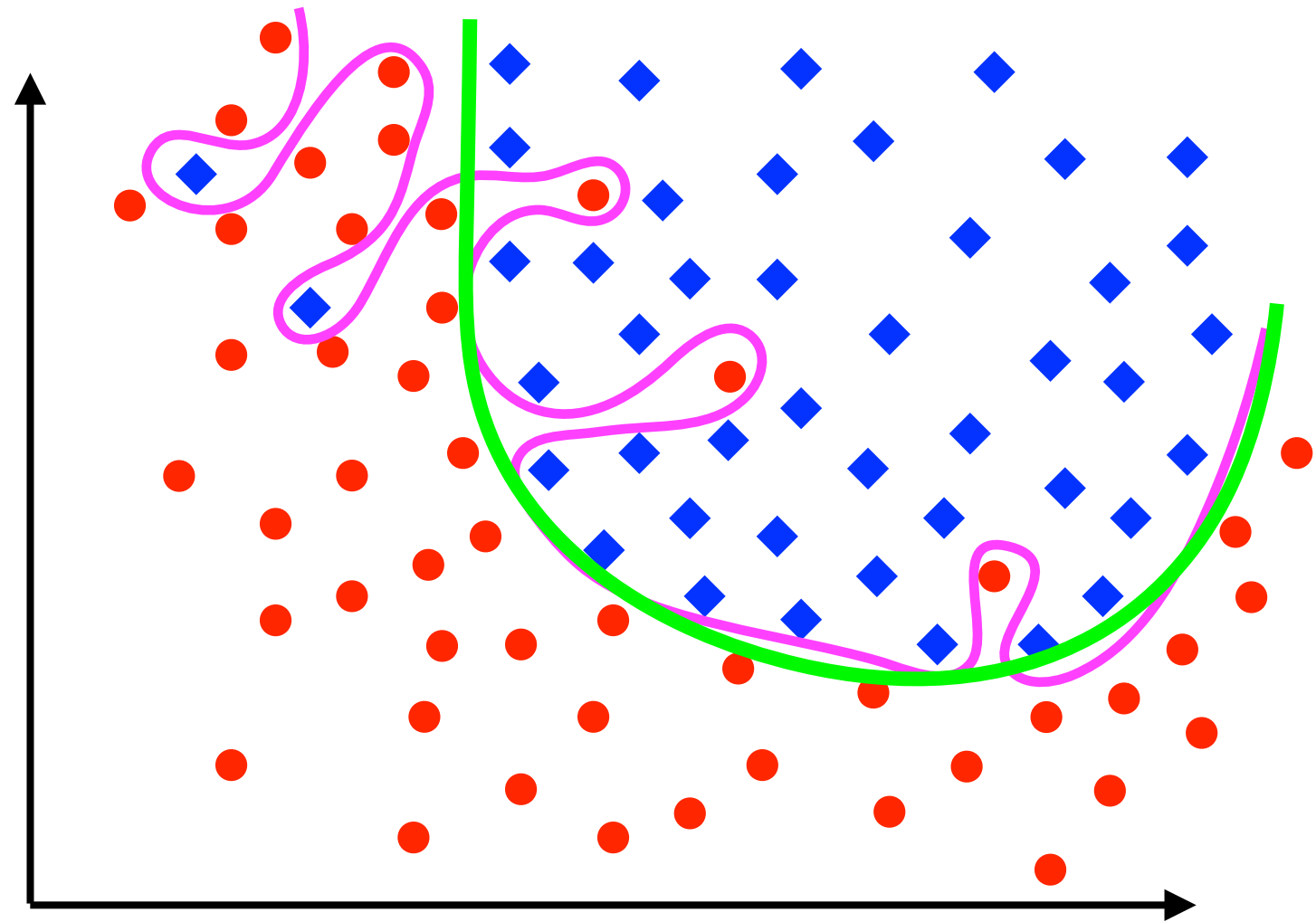
Проблеми свързани при пренапасване

- При пренапасване — модела е предостатъчно изразителен но от малкото данни научава грешна функция. В този случай функцията много добре описва тренировачните данни но не се обобщава добре за ненаблюдавани данни.



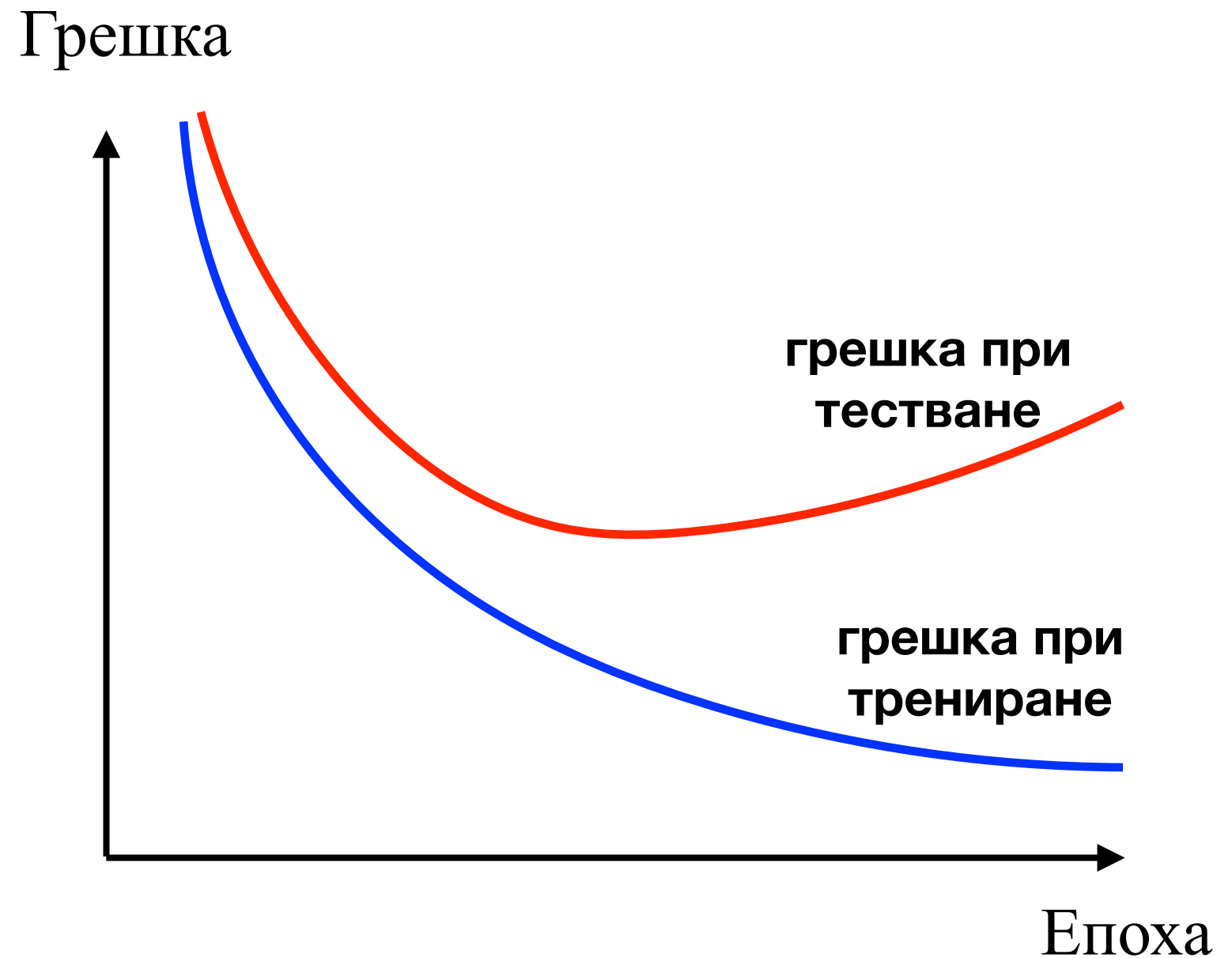
Проблеми свързани при пренапасване

- При пренапасване — модела е предостатъчно изразителен но от малкото данни научава грешна функция. В този случай функцията много добре описва тренировачните данни но не се обобщава добре за ненаблюдавани данни.



Проблеми свързани при пренапасване

- При пренапасване — по-дългото обучение на модела води до увеличаване на грешката върху ненаблюдавани данни.
- Решения:
 - Използване на по-прост модел (опасност от недонапасване)
 - Осигуряване на повече данни за обучение (скъпо и трудоемко)
 - Регуларизация



L_2 Регуляризация

- Досега се стремяхме да минимизираме кросентропията:

$$H_X[\text{Pr} \parallel \text{Pr}_\theta] = -\frac{1}{|X|} \sum_{i=1}^{|X|} \log \text{Pr}_\theta[\mathbf{x}^{(i)}], \text{ където } \theta \in \mathbb{R}^K \text{ е векторът от параметрите на модела.}$$

- Към целевата функция добавяме регуляризационен член:

$$J_X(\theta) = H_X[\text{Pr} \parallel \text{Pr}_\theta] + \lambda \|\theta\|^2 = -\frac{1}{|X|} \sum_{i=1}^{|X|} \log \text{Pr}_\theta[\mathbf{x}^{(i)}] + \lambda \sum_k \theta_k^2$$

- Идеята е, да се стремим да използваме възможно най-малко от параметрите.
- Освен L_2 регуляризация често се използва и L_1 , а също и комбинация.
- Параметърът λ е метапараметър и следва да се напасне с валидиране.

Dropout регуларизация

- Нека кросентропията в точката (наблюдението) i е:
 $H_i = -\log \text{softmax}(W\mathbf{x}_i + \mathbf{b})_{c_i}$, където $\mathbf{x}_i \in \mathbb{R}^N$
- По време на обучение за случайно избираме вектор $\mathbf{r} \in \{0,1\}^N$ с N -мерно Бернулиево разпределение с вероятност p и използваме поточкова кросентропия: $H_i = -\log \text{softmax}(W\hat{\mathbf{x}}_i + \mathbf{b})_{c_i}$, където $\hat{\mathbf{x}}_i = \mathbf{r} \odot \mathbf{x}_i$.
- Идеята е, че ако по случаен начин изтрием част от входа, моделът няма да разчита твърде много на всеки отделен параметър, а ще трябва да научи по-обща закономерности.
- По време на приложение на модела вместо dropout се скалира: $\hat{\mathbf{x}}_i = p\mathbf{x}_i$.
- Параметърът p е метапараметър и се напасва с валидиране.
- Dropout регуларизация се използва често и в други слоеве (не само накрая).

Сравнение между различни архитектури

- **SBOW**: Изненадващо добро изходно ниво за прости класификационни проблеми. Особено ако са последвани от няколко слоя перцептрони с ReLU активация.
- **Модели с прозорец**: Подходящи за класификация на думи при задачи, които не се нуждаят от широк контекст. Например, определяне частите на речта (Part-of-Speech), разпознаване на изрази за названия (Named entity recognition).
- **КНН**: подходящи за класификация, нуждаят се от допълване за по-кратки фрази, трудни за интерпретиране, лесни за паралелизиране на графичните процесори. Ефективни и сравнително универсални.
- **РНН**: Когнитивно правдоподобни (чете се отляво надясно), не са най-доброто за класификация (ако се използва само последно състояние), много по-бавни от КНН, добри за аотиране на последователности, чудесни за езикови модели, може да бъдат невероятно ефективни при допълване с механизми за внимание (ще разгледаме на следващата лекция).