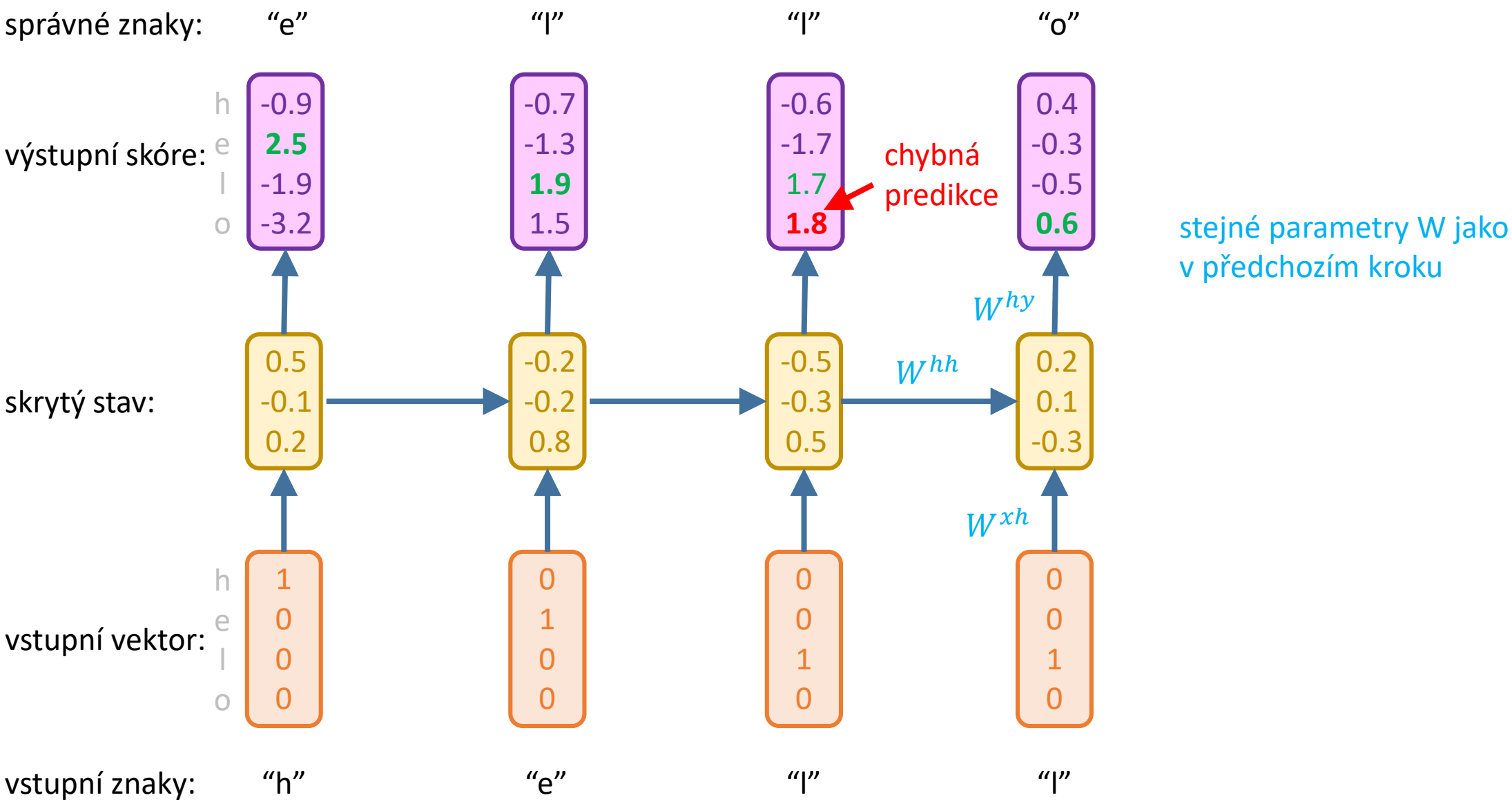


Aplikace neuronových sítí

Sequence-to-sequence

RNN znakový jazykový model



Písmena vs slova

- Místo znaků modelovat závislost slov
- U znaků jsme každé písmeno reprezentovali jako jednotkový vektor (one-hot kódování)
- Matice všech vektorů tedy jednotková s velikostí = počet symbolů
- Problém: slovník řádově $\approx 10^4, 10^5 \rightarrow$ matice $10^5 \times 10^5 = 10^{10} \cdot 4B \approx 40GB$!
 - matice je řídká, pouze jednotkové vektory \rightarrow nemusí být celá v paměti
 - ale: ovlivňuje i velikost matice parametrů $W^{xh} \rightarrow$ bude mít rozměr $10^5 \times H$
- Řešení: reprezentace znaků kratšími vektory \rightarrow matice např. $10^5 \times 300 \approx 12MB$
 - matice W^{xh} : $300 \times H$

Písmena vs slova

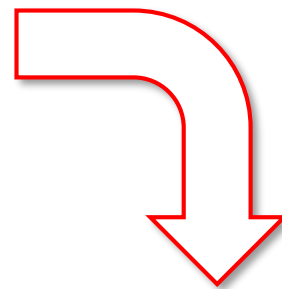
one-hot řídká reprezentace

symbol 1:	1	0	0	0	0	0
symbol 2:	0	1	0	...	0	0
symbol 3:	0	0	1		0	0
⋮		⋮		⋱	⋮	
symbol V-2:	0	0	0		1	0
symbol V-1:	0	0	0	...	0	1
symbol V:	0	0	0		0	1

počet symbolů

počet symbolů

Jak ale vektory zvolit?



hustá reprezentace

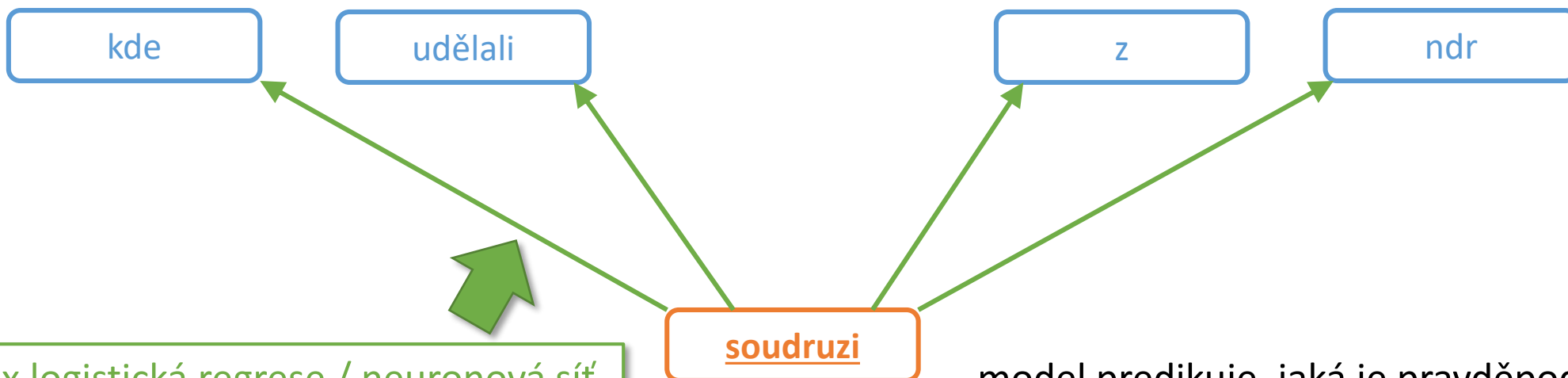
libovolná dimenze

symbol 1:	0.25	-0.11	1.14		-0.4	-2.0	2.1
symbol 2:	5.2	1.01	0.1	...	0.22	0.31	-4.44
symbol 3:	4.13	10.1	5.5		3.32	-0.12	-0.34
⋮		⋮		⋱		⋮	
symbol V-2:	-0.45	-0.56	0.67		1.64	2.32	-3.16
symbol V-1:	2.22	3.33	1.11	...	-0.01	2.74	3.14
symbol V:	2.81	-7.1	-0.05		1.13	-3.61	1.0

word2vec

- Tzv. skip-gram model → predikce okolních slov z aktuálního (prostředního)

Příklad: “Kde udělali soudruzi z NDR chybu?”



softmax logistická regrese / neuronová síť, která se učí vektory slov tak, aby maximalizovala klasifikační skóre

model predikuje, jaká je pravděpodobnost, že kdekoliv v sousedství slova “soudruzi” se vyskytuje např. slovo “udělali”

word2vec: trénovací data

kde	udělali	soudruzi	z	ndr	chybu
-----	---------	----------	---	-----	-------



(kde, udělali)
(kde, soudruzi)

kde	udělali	soudruzi	z	ndr	chybu
-----	---------	----------	---	-----	-------



(udělali, kde)
(udělali, soudruzi)
(udělali, z)

kde	udělali	soudruzi	z	ndr	chybu
-----	---------	----------	---	-----	-------



(soudruzi, kde)
(soudruzi, udělali)
(soudruzi, z)
(soudruzi, ndr)



velikost posuvného okna

⋮

kde	udělali	soudruzi	z	ndr	chybu
-----	---------	----------	---	-----	-------



(chybu, z)
(chybu, ndr)

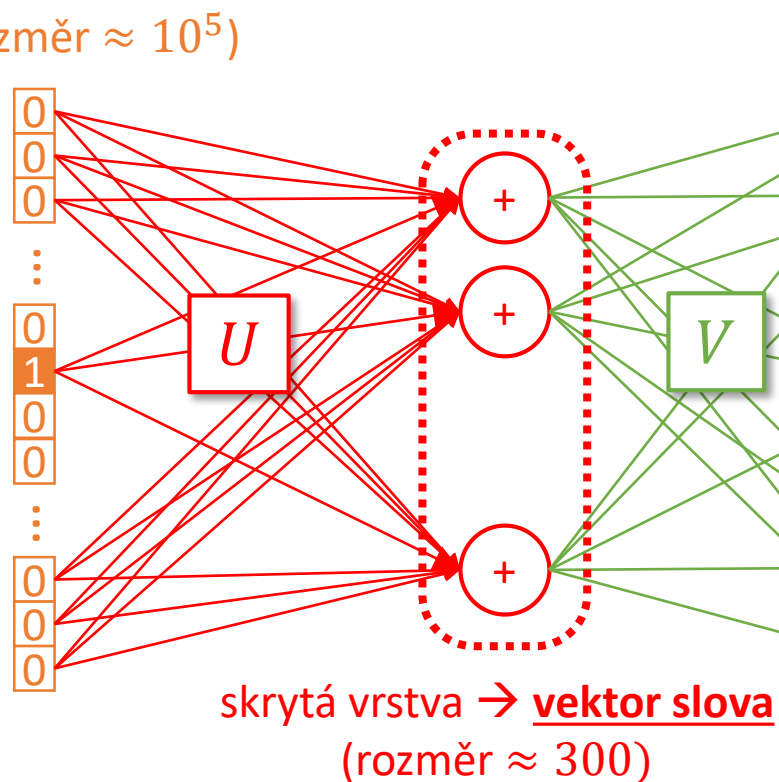
trénovací páry (x, y)



word2vec: základní architektura sítě

trénovací pár ("soudruzi", "udělali")

one-hot reprezentace
slova "soudruzi"
(rozměr $\approx 10^5$)



U matice word-vektorů $\approx 10^5 \times 300$

logistická regrese
(rozměr $\approx 10^5$)

V matice klasifikátoru $\approx 300 \times 10^5$

predikce

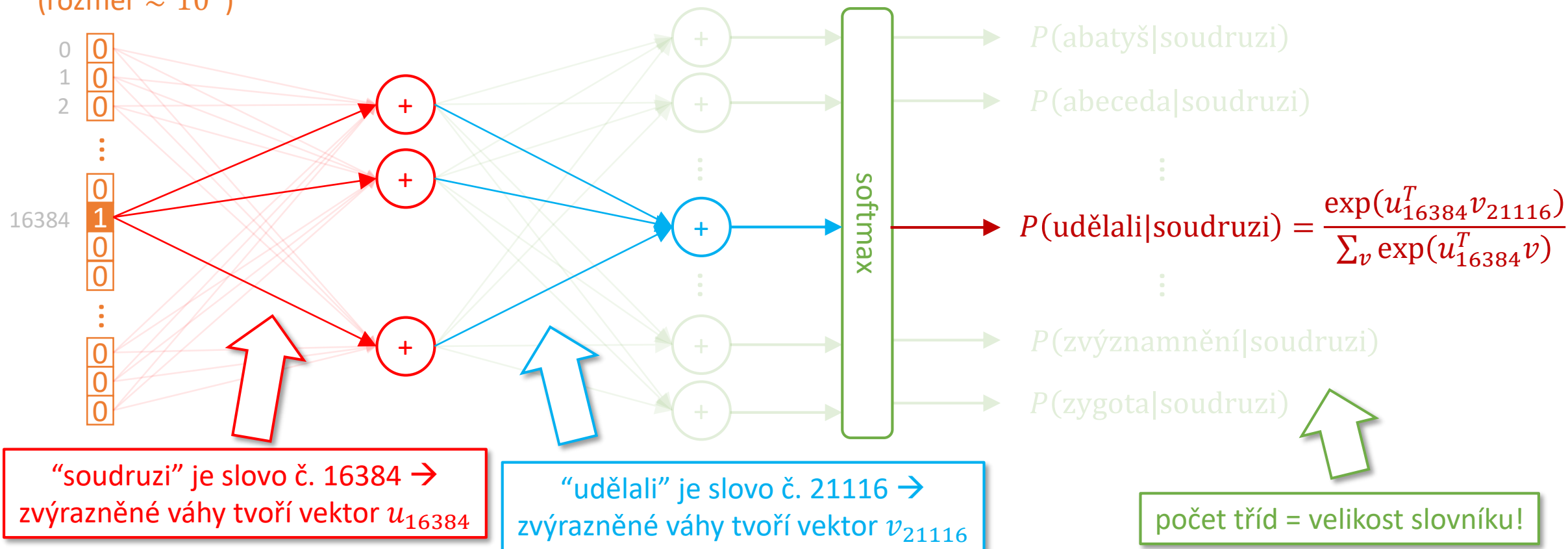
správná hodnota

predikce	správná hodnota
$P(\text{abatyš} \text{soudruzi})$	$= 0$
$P(\text{abeceda} \text{soudruzi})$	$= 0$
\vdots	\vdots
$P(\text{udělali} \text{soudruzi})$	$= 1$
\vdots	\vdots
$P(\text{zvýznamnění} \text{soudruzi})$	$= 0$
$P(\text{žvýkačka} \text{soudruzi})$	$= 0$

word2vec: základní architektura sítě

trénovací pár (“soudruzi”, “udělali”)

one-hot reprezentace
slova “soudruzi”
(rozměr $\approx 10^5$)



word2vec: kritérium

- Maximalizujeme pravděpodobnost správného slova (třídy), tzn. **pro jeden krok** (jedno aktuální slovo u “v čase t ”) je kritérium

$$L_t(U, V) = \prod_{v \in \text{okolí}(u)} P(v|u) = \prod_{v \in \text{okolí}(u)} \frac{\exp(u^T v)}{\sum_{w \in \text{slovník}} \exp(u^T w)}$$

- Což je zjednodušená forma zápisu, který je např. v přednáškách [Stanford CS224D \(2017\)](#)
- **word2vec je ale jen logistická regrese** → prostě minimalizujeme křížovou entropii

$$L_{u,v}(U, V) = \underbrace{-u^T v}_{\text{skóre správné třídy}} + \log \sum_{w \in \text{slovník}} \exp(u^T w)$$

- a to opakujeme pro každé slovo u z trénovacího korpusu a pro každé $v \in \text{okolí}(u)$

word2vec: záporné vzorkování

- Problém:

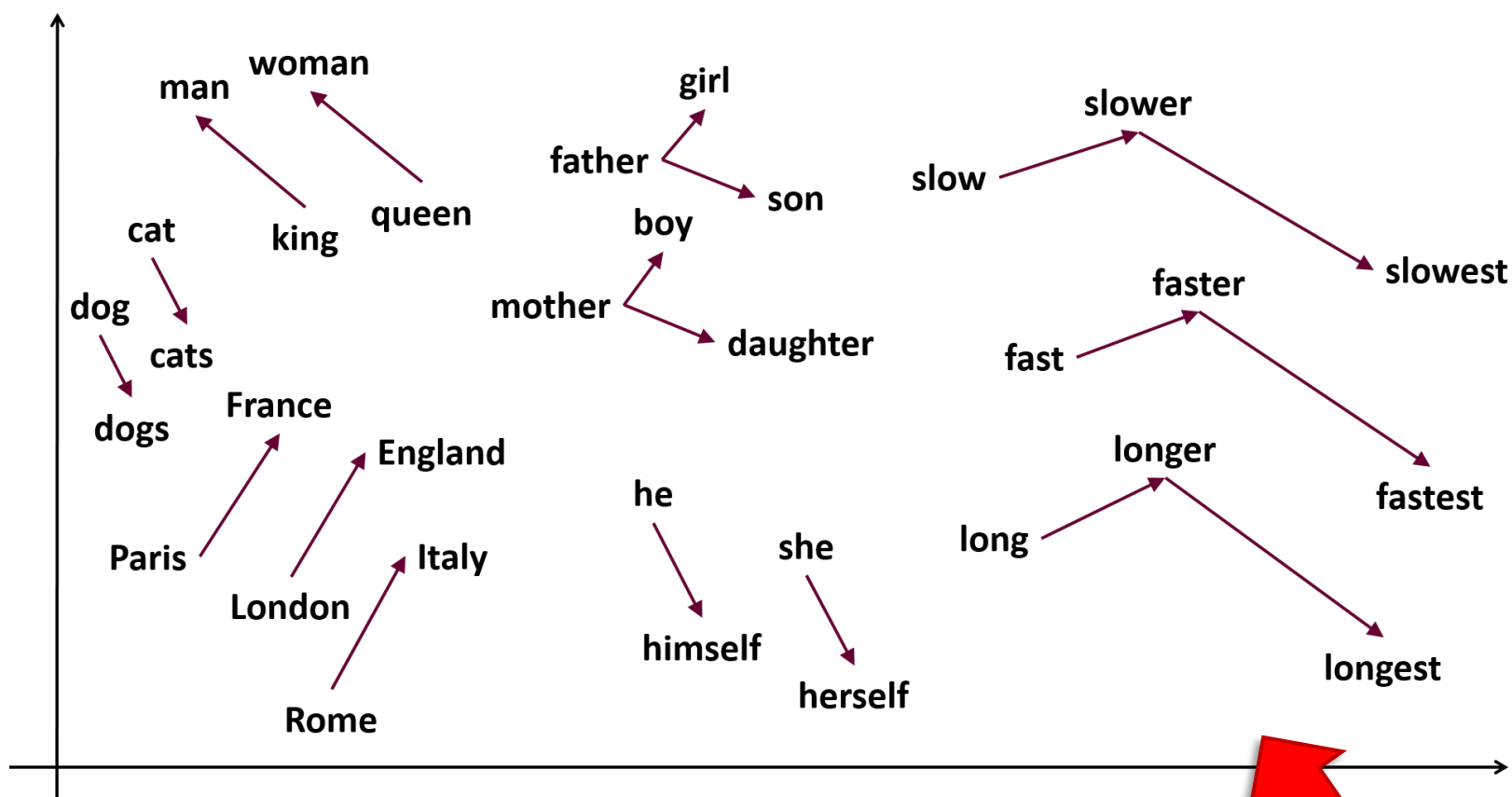
$$L_{u,v}(U, V) = -u^T v + \log \sum_{w \in \text{slovník}} \exp(u^T w)$$

- se v každém kroku vyhodnocuje pro celý slovník → velmi pomalé
- Suma představuje “záporná slova” w – která mají mít pro slovo u malou $P(w|u)$
- Není přitom nutné vyhodnocovat pro celý slovník, např. dvojici “soudruzi” a “labrador”
- Stačí 5-20 slov pro menší datasety, 2-5 pro velké
- Suma se tedy vyhodnotí místo celého slovníku jen pro několik málo slov, tj.

$$L_{u,v}(U, V) = -u^T v + \log \sum_{w \in \text{náhodný výběr}} \exp(u^T w)$$

- Výběr není úplně náhodný, závisí na frekvenci slov apod.
 - detaily v článku či např. zde: [Stanford CS224D \(2017\)](#)

word2vec: naučené vztahy mezi slovy



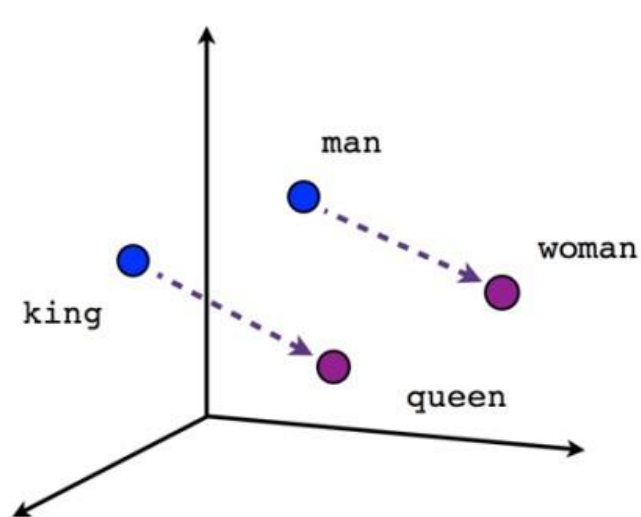
Příklady: France – Paris + London = England
king - man + woman = queen

...

obrázek: <http://www.samyzaf.com/ML/nlp/nlp.html>

každé slovo je vektor, zde 2D

word2vec: naučené vztahy mezi slovy

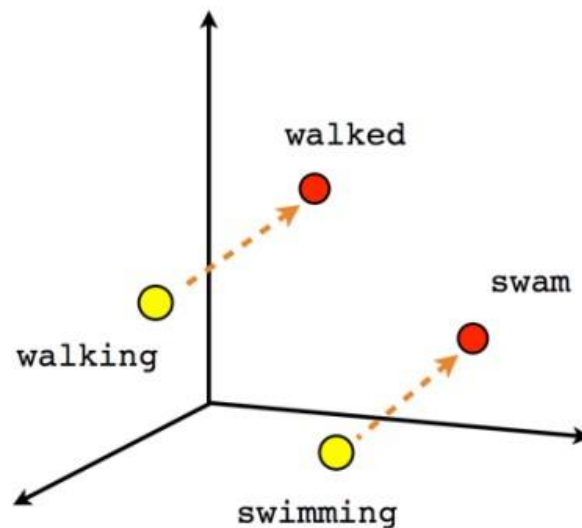


Male-Female

king – man + woman = queen



queen – king = woman – man

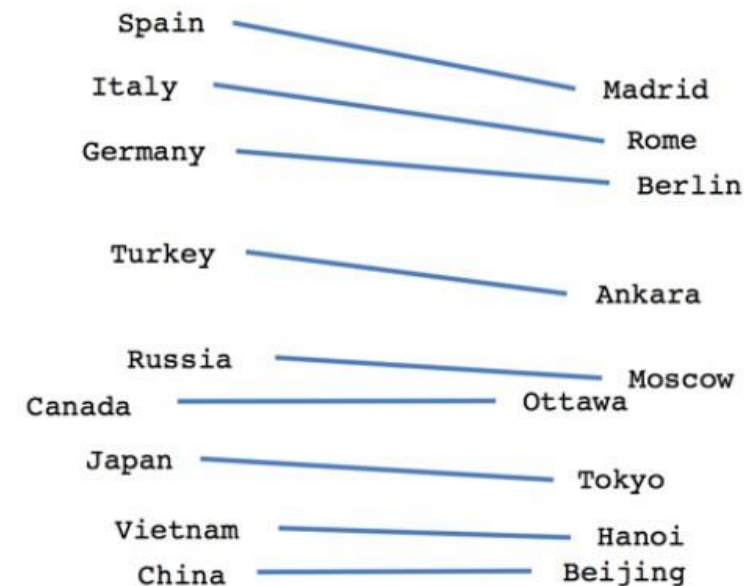


Verb tense

walking – swimming + swam = walked



walked – walking = swam – swimming



Country-Capital

podobně pak státy a města jsou od sebe vždy stejným směrem (liší se od sebe stejným vektorem)

obrázek: <https://www.analyticsvidhya.com/blog/2017/06/word-embeddings-count-word2vec/>

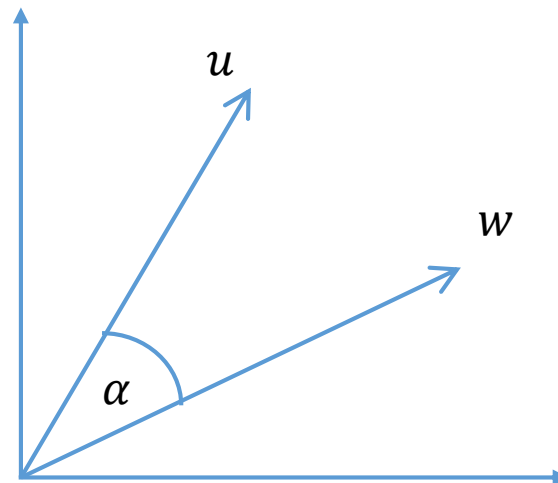
word2vec: nejbližší soused a kosinová vzdálenost

- Výsledkem operace s vektory je vektor

$$u = r - s + v$$

- Jakému slovu u odpovídá?
- Mezi všemi slovy $w \in$ slovník najdeme takový, který se mu nejvíce podobá
- Používá se kosinová vzdálenost

$$d(u, w) = \cos \alpha = \frac{u^T w}{\|u\| \cdot \|w\|}$$

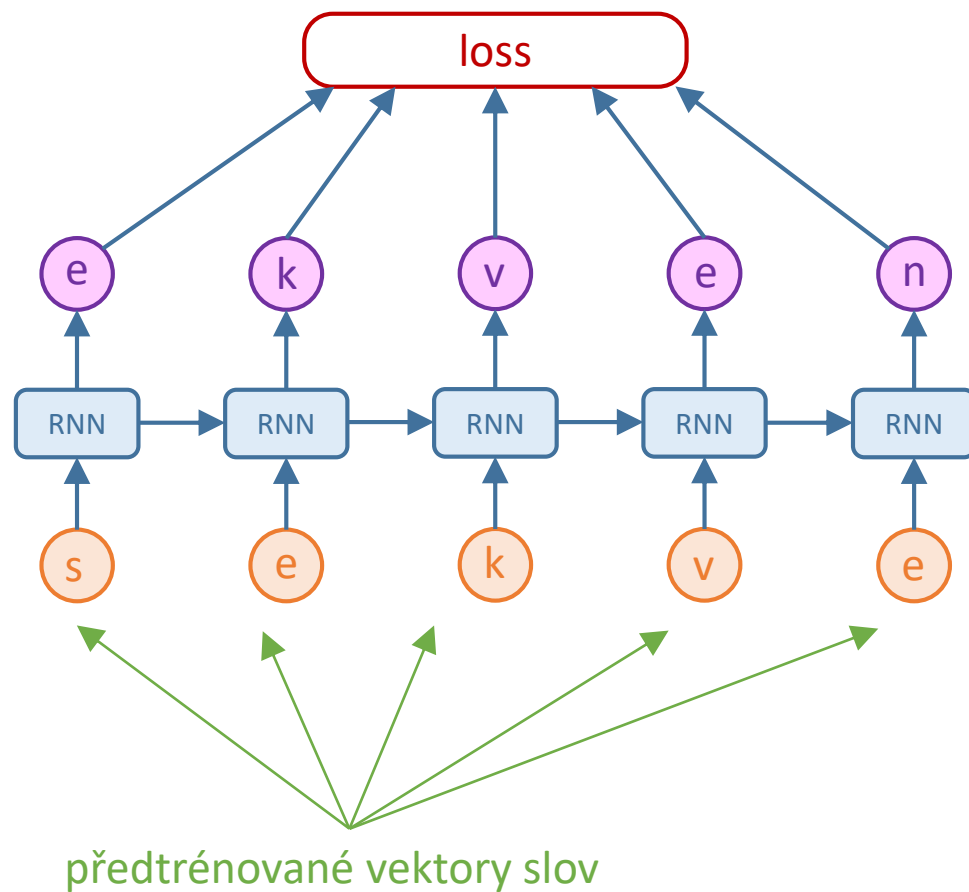


- Nezohledňuje velikost
- Zákonem velkých čísel bývají vektory ve vysokodim. prostorech na vrcholech hyperkvádrů (jedna ze souřadnic \gg ostatní) \rightarrow úhly vystihují lépe než vzdálenost

word2vec: poznámky

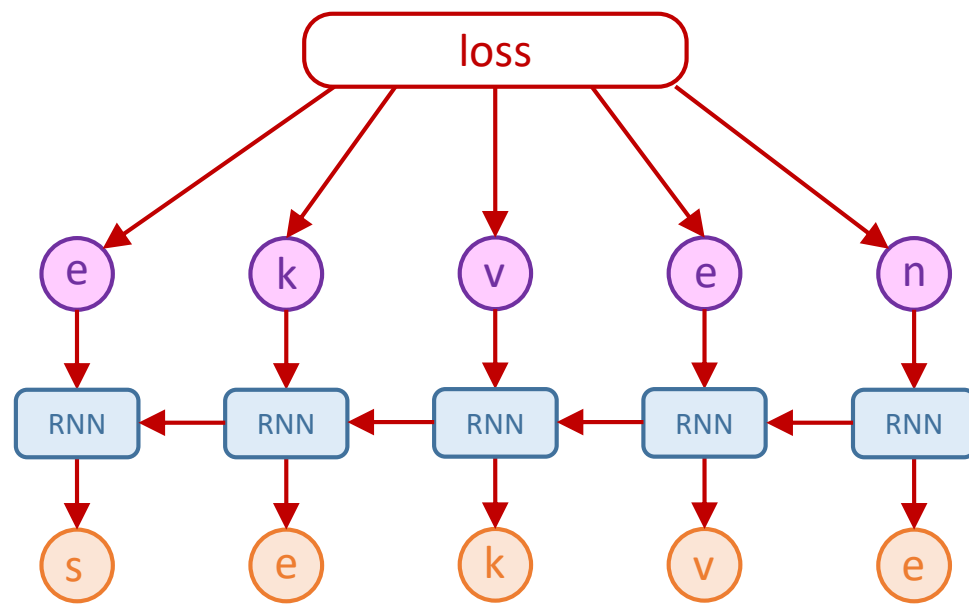
- V příkladech z praxe obvykle velikost okna = 5 (tj. 5 nalevo + 5 napravo)
- Dimenze vektorů 300
- Kromě záporného vzorkování se optimalizuje navíc omezováním příliš častých slov nebo vyhledáváním častých frází, které se pak považují za jedno slovo (např. “Dobrý den”)
- Pro různé jazyky existují předtrénované word-vektory na velkých korpusech podobně jako u konvolučních sítí → není nutné trénovat
 - např. zde: <https://fasttext.cc/docs/en/pretrained-vectors.html>
- Existují i další známé varianty:
 - CBOW namísto skip-gram: predikce prostředního slova ze sumy okolních
 - GloVe: kombinuje word2vec se singulárním rozkladem SVD
 - byt to autoři alt. metod rádi tvrdí, žádná z variant ve skutečnosti nefunguje lépe než word2vec

word2vec: finetuning



- podobný princip jako u předtrénovaných konvolučních sítí
- stáhneme předtrénované vektory a použijeme ve vlastní úloze

word2vec: finetuning

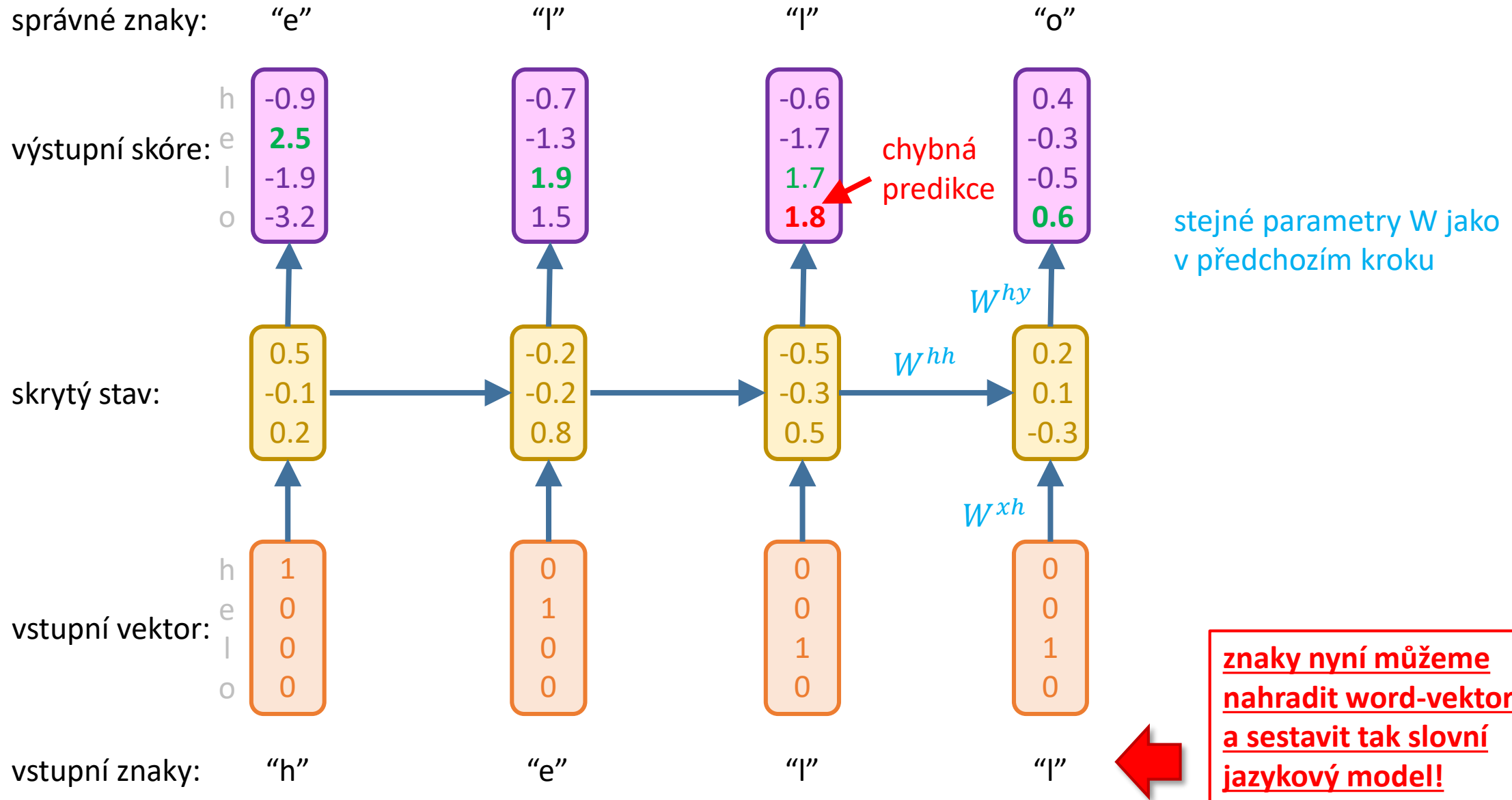


předtrénované vektory slov

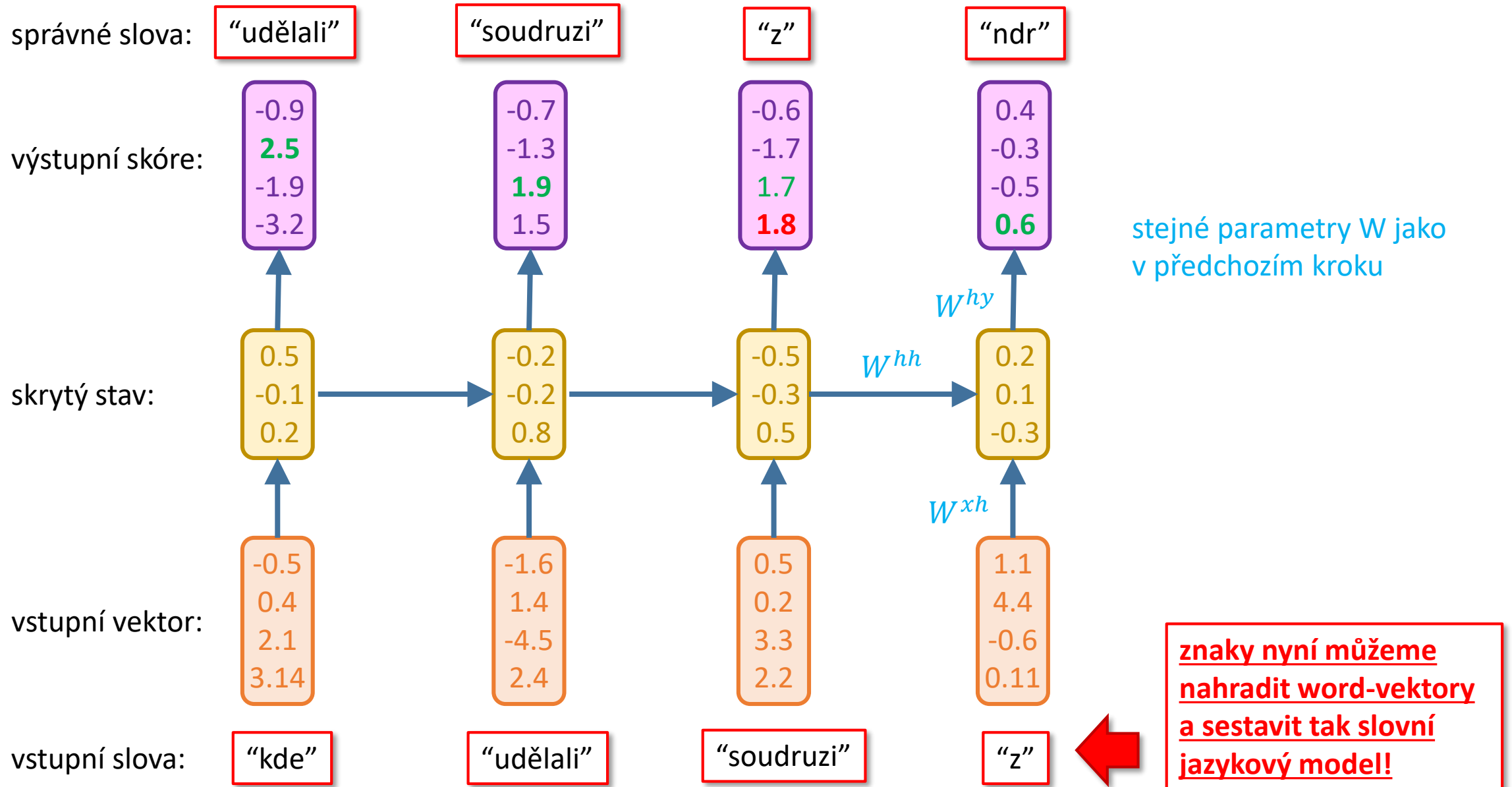
- podobný princip jako u předtrénovaných konvolutních sítí
- stáhneme předtrénované vektory a použijeme ve vlastní úloze
- můžeme i finetunit

gradient lze propagovat i do
vstupu →
síť se bude zároveň učit i
reprezentaci dat!

RNN znakový jazykový model



RNN znakový jazykový model



Analýza sentimentu

- Monitorování médií, kampaně, sledování značek, recenzí, telefonních rozhovorů, ...

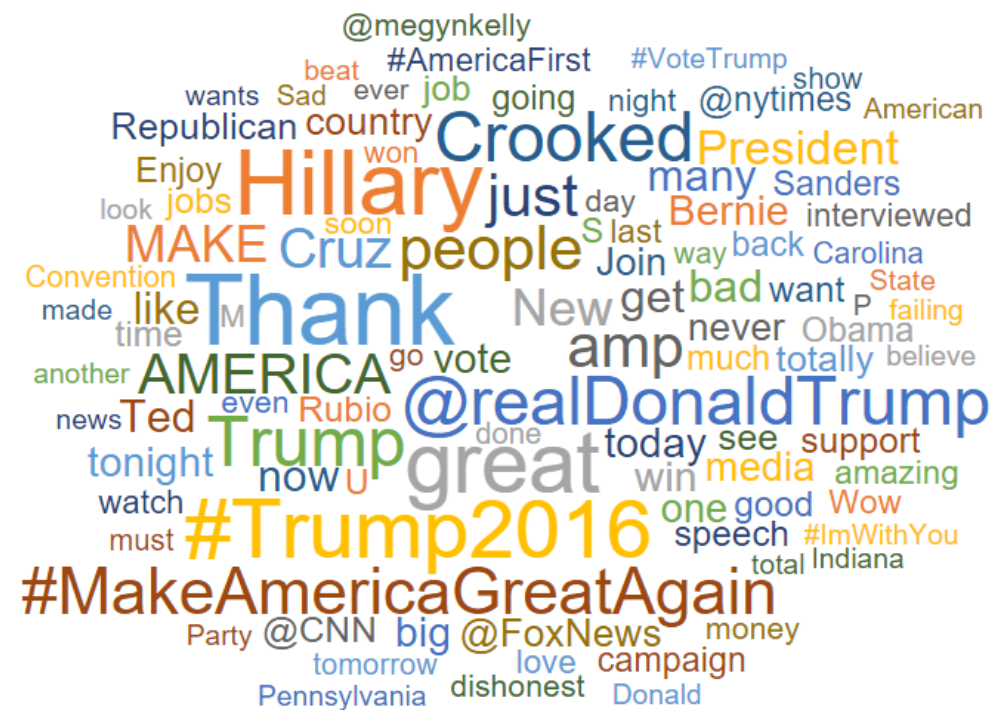
příklad:

film je plný akce, speciálních efektů a humoru, řemeslné zpracování na jedničku, děj sice dětinský, ale zábavný

vs

přemíra akce, speciálních efektů a humoru, kvalitní zpracování, sice vcelku zábavný, ale dětinský

zpříkladu je zřejmé, že pouhé počítání kladných a záporných přívlastků stačit nebude

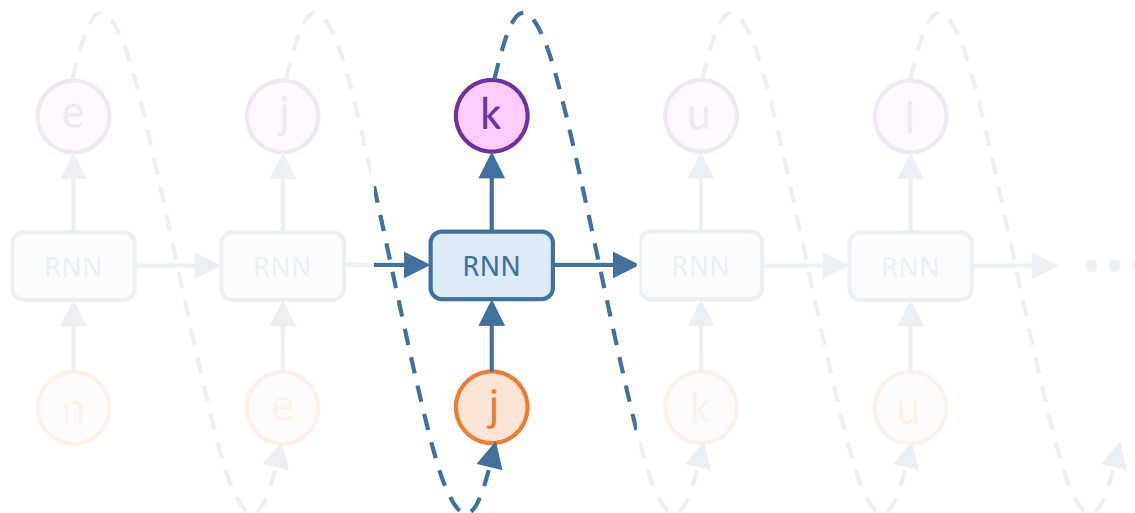


RNN jazykový model

v každém okamžiku t síť přijme jeden symbol a ihned predikuje následující

- vstup: jeden znak / slovo
- výstup: jeden znak / slovo

label síť generuje v každém kroku



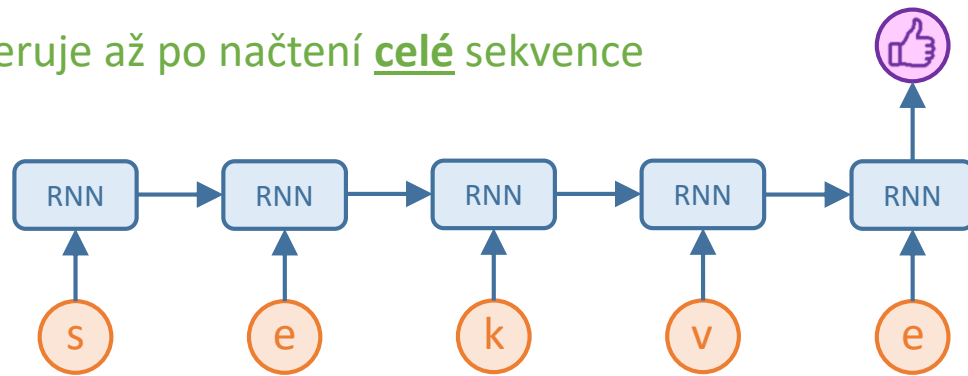
vstup : výstup \rightarrow 1 : 1

Klasifikace textu

např. analýza sentimentu: klasifikace **celého** textu do jedné z kategorií, např. pozitivní vs negativní komentář

- vstup: sekvence znaků / slov
- výstup: třída

label síť generuje až po načtení **celé** sekvence



celý text reprezentován jako vektor
(např. poslední stavový vektor)

vstup : výstup $\rightarrow \underline{M} : 1$

Sentiment analysis: state of the art

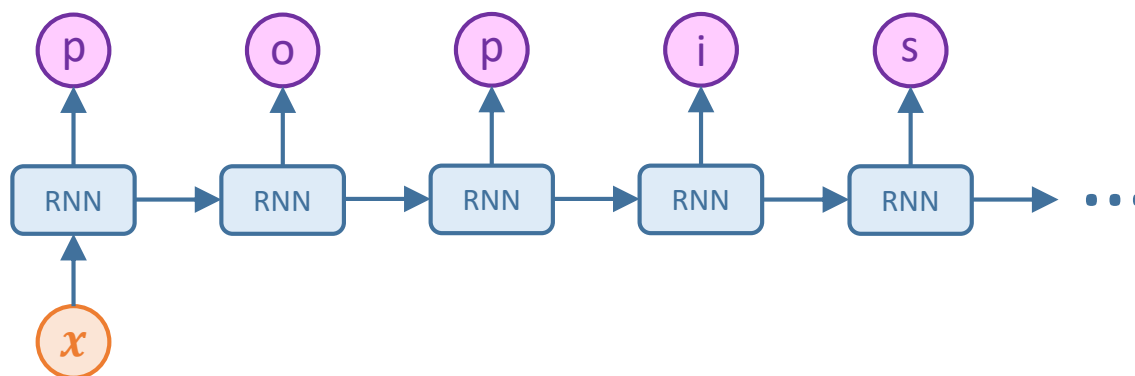


obrázek: <http://blog.paralleldots.com/data-science/breakthrough-research-papers-and-models-for-sentiment-analysis/>

Generování textu, tagování obrázků

síť převezme vstup pouze jednou jako inicializaci, poté v každém okamžiku t síť predikuje výstup (následující znak)

- vstup: jeden vektor, např. FC7 příznaky z VGG
- výstup: sekvence znaků / slov



síť převezme pouze jeden vstup, pak už jen generuje

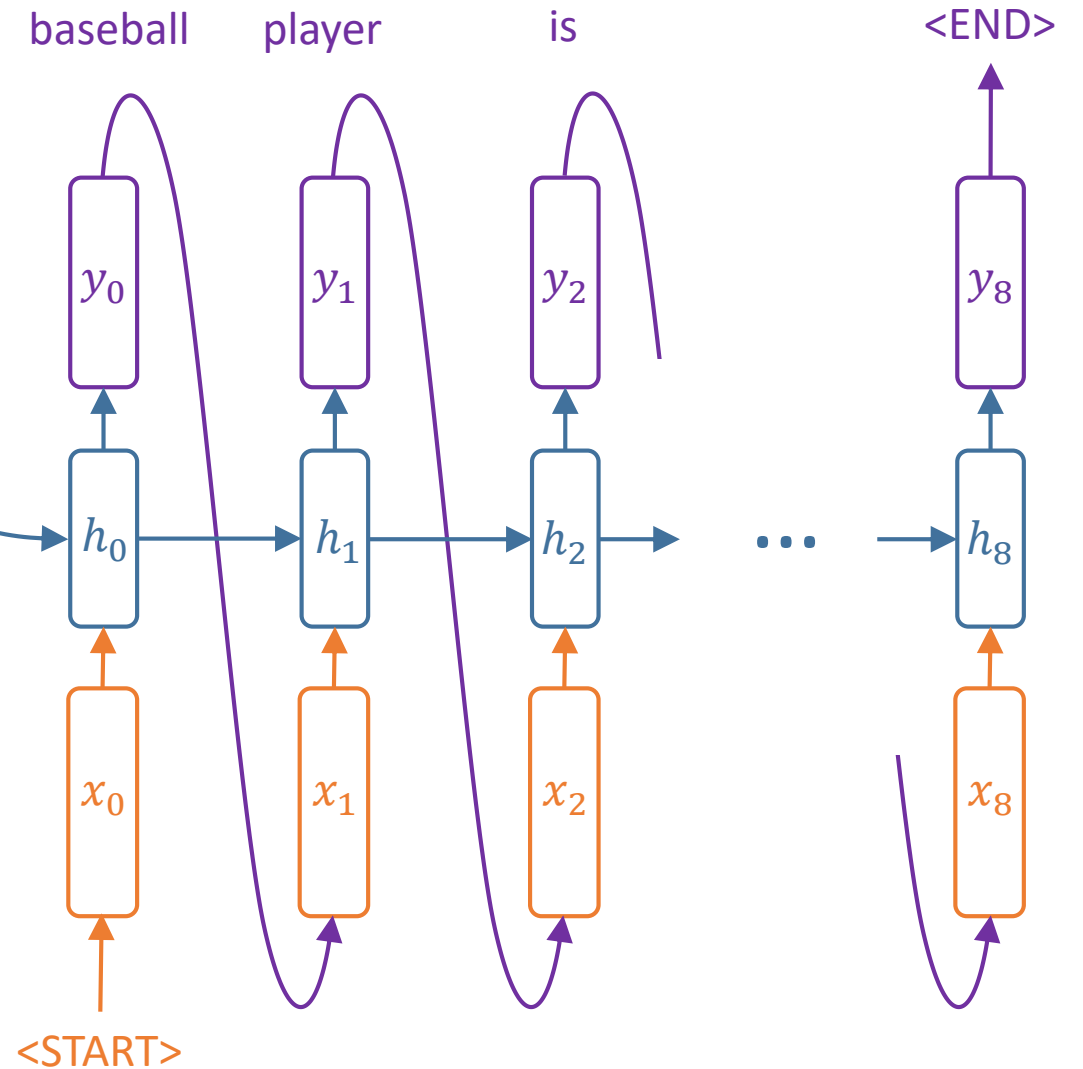
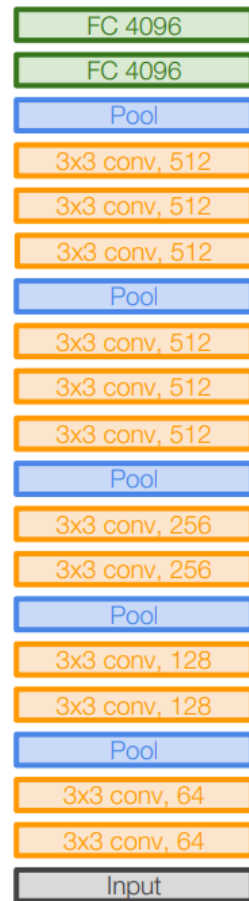
vstup : výstup $\rightarrow 1 : \underline{N}$

Tagování obrázků: VGG schéma



"baseball player is throwing ball
in game."

VGG-16



Tagování obrázků



"girl in pink dress is jumping in air."



"black and white dog jumps over bar."



"young girl in pink shirt is swinging on swing."



"man in blue wetsuit is surfing on wave."



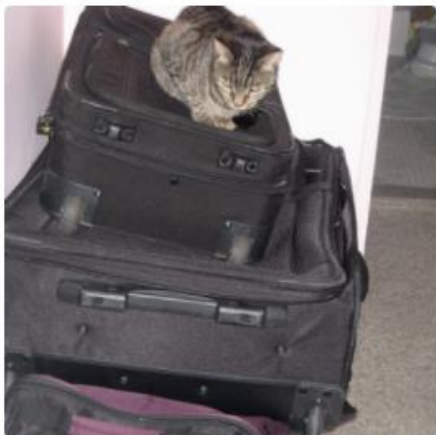
"little girl is eating piece of cake."



"baseball player is throwing ball in game."



"woman is holding bunch of bananas."



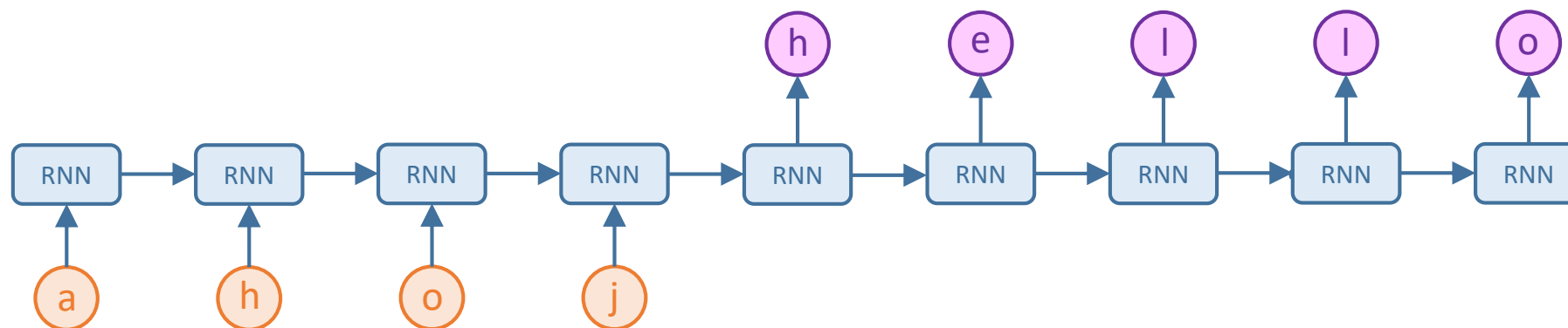
"black cat is sitting on top of suitcase."

obrázek: <https://cs.stanford.edu/people/karpathy/deepimagesent/>

Strojový překlad

síť nejprve “spolkne” celou vstupní sekvenci (větu), pak teprve začne generovat překlad

- vstup: sekvence znaků / slov
- výstup: sekvence znaků / slov

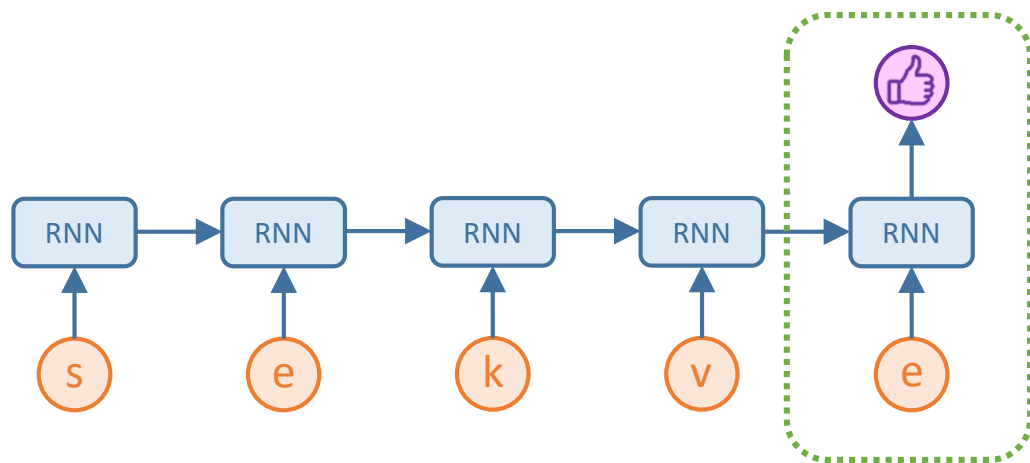


vstup : výstup $\rightarrow \underline{M} : \underline{N}$

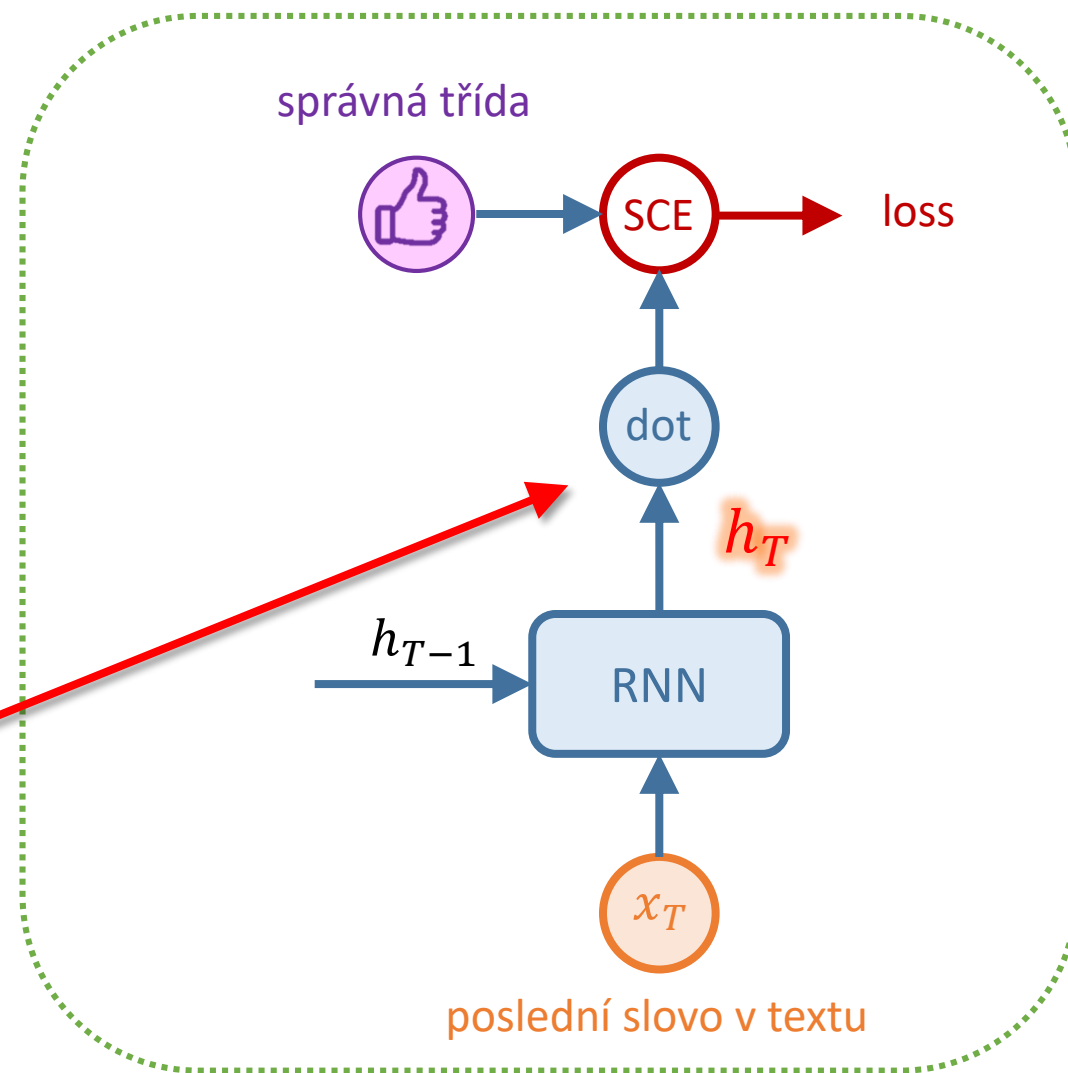
Klasifikace textu

např. analýza sentimentu: klasifikace celého textu do jedné z kategorií, např. pozitivní vs negativní komentář

- vstup: sekvence znaků / slov
- výstup: třída



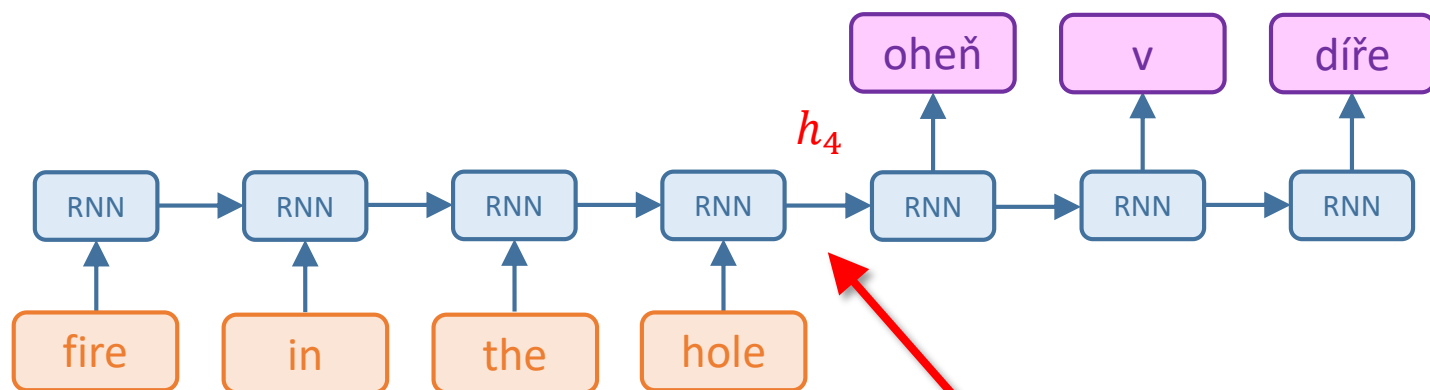
problém: stavový vektor h_T musí nějak obsahovat informaci z celé věty, protože dále už následuje pouze lineární klasifikace na základě tohoto vektoru



Strojový překlad

síť nejprve “spolkne” celou vstupní sekvenci (větu), pak teprve začne generovat překlad

- vstup: sekvence znaků / slov
- výstup: sekvence znaků / slov

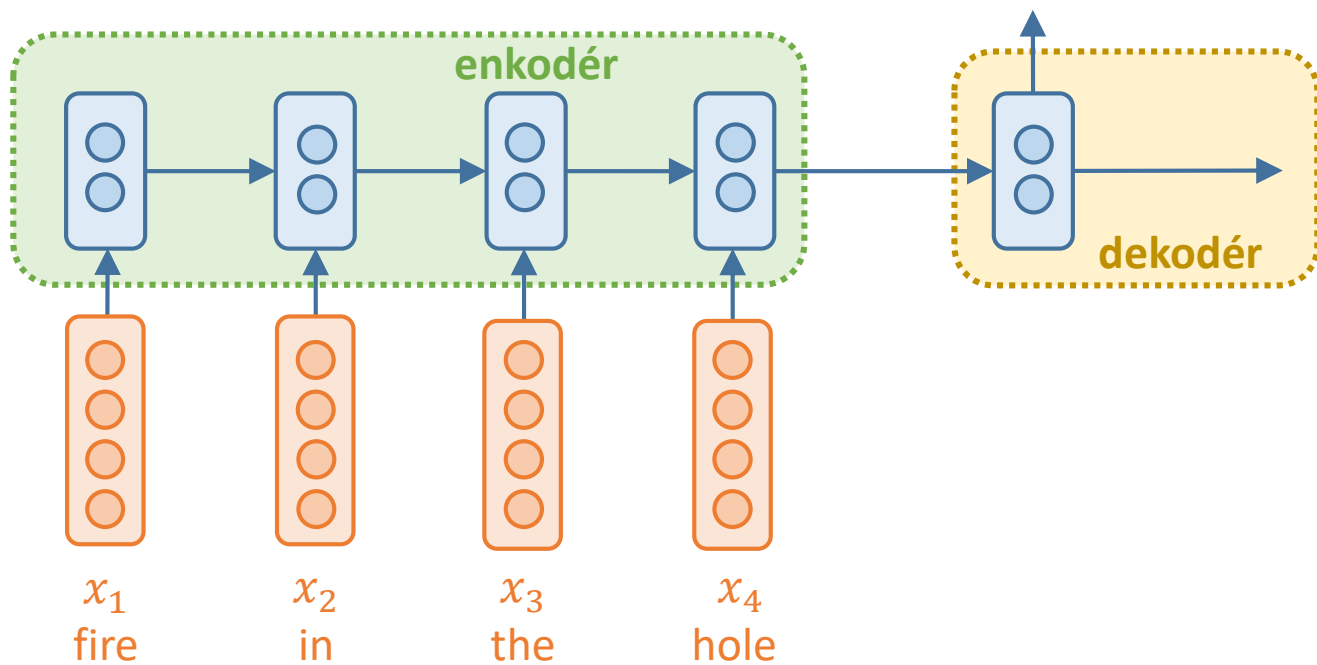


podobný problém i zde: na to, aby síť vygenerovala celý překlad, jí musí stačit skrytý stavový vektor h_4 , slova “fire” a “oheň” jsou však od sebe vzdálena potenciálně neomezeně → problém s gradientem

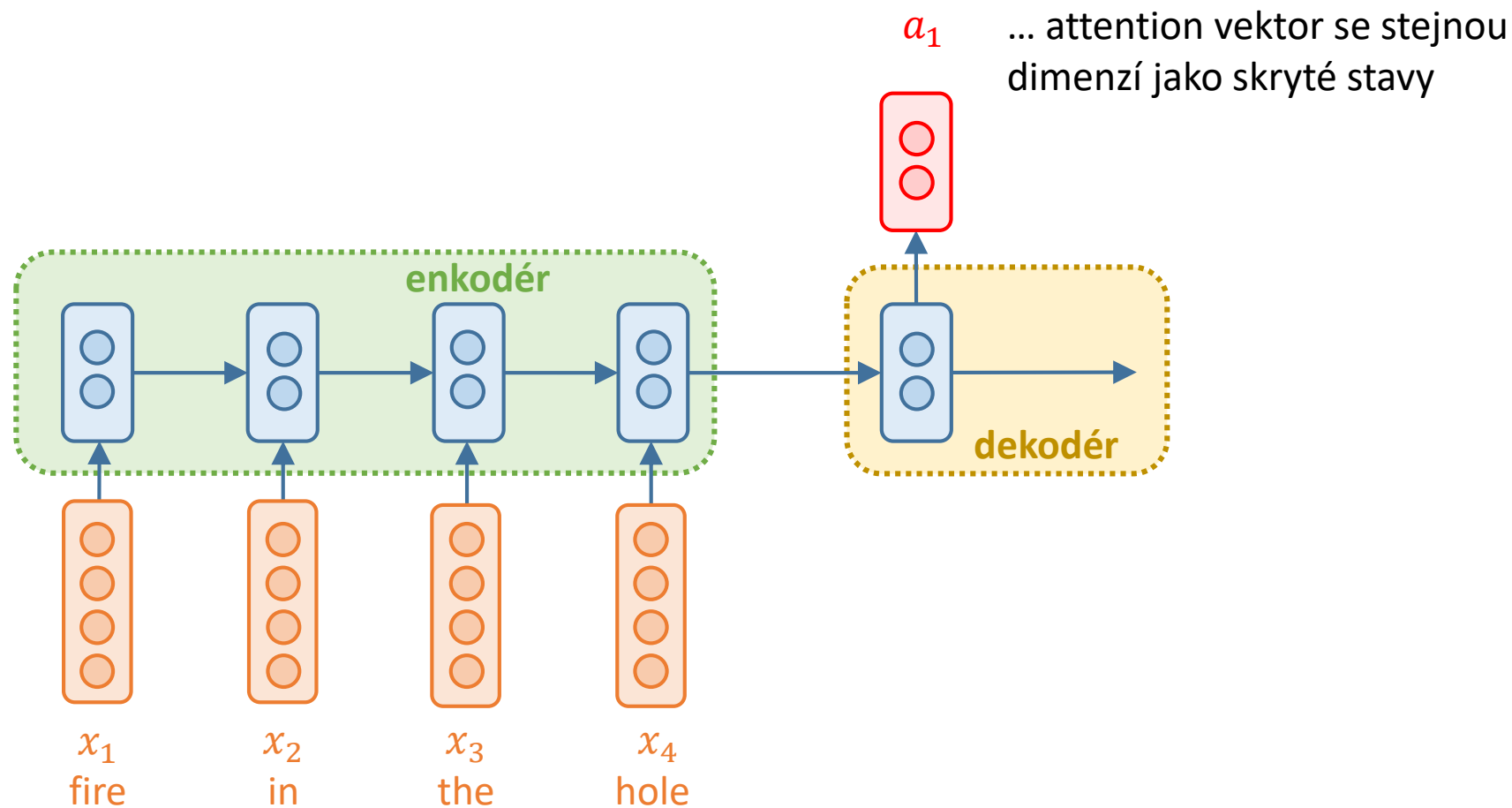
Attention

- Problémy:
 - stavový vektor h_t musí nějak obsahovat celou větu, protože dale už následuje pouze lineární klasifikace na základě tohoto vektoru → sentence embedding
 - nezávisle na délce vstupní věty, embedding vektor má vždy stejnou dimenzi
 - navíc řešíme potenciálně velmi dlouhé závislosti, na které je v praxi i LSTM krátká
- Pokusy o řešení:
 - především v oblasti strojového překladu zadat vstupní sekvenci v obráceném pořadí
 - nechat projít vstupní sekvenci rekurentní sítí 2x
- Řešení:
 - mechanismus paměti/soustředěnosti (attention)
 - zapamatovat a při rozhodování váhově zohlednit všechny skryté stavy, ne jen ten poslední

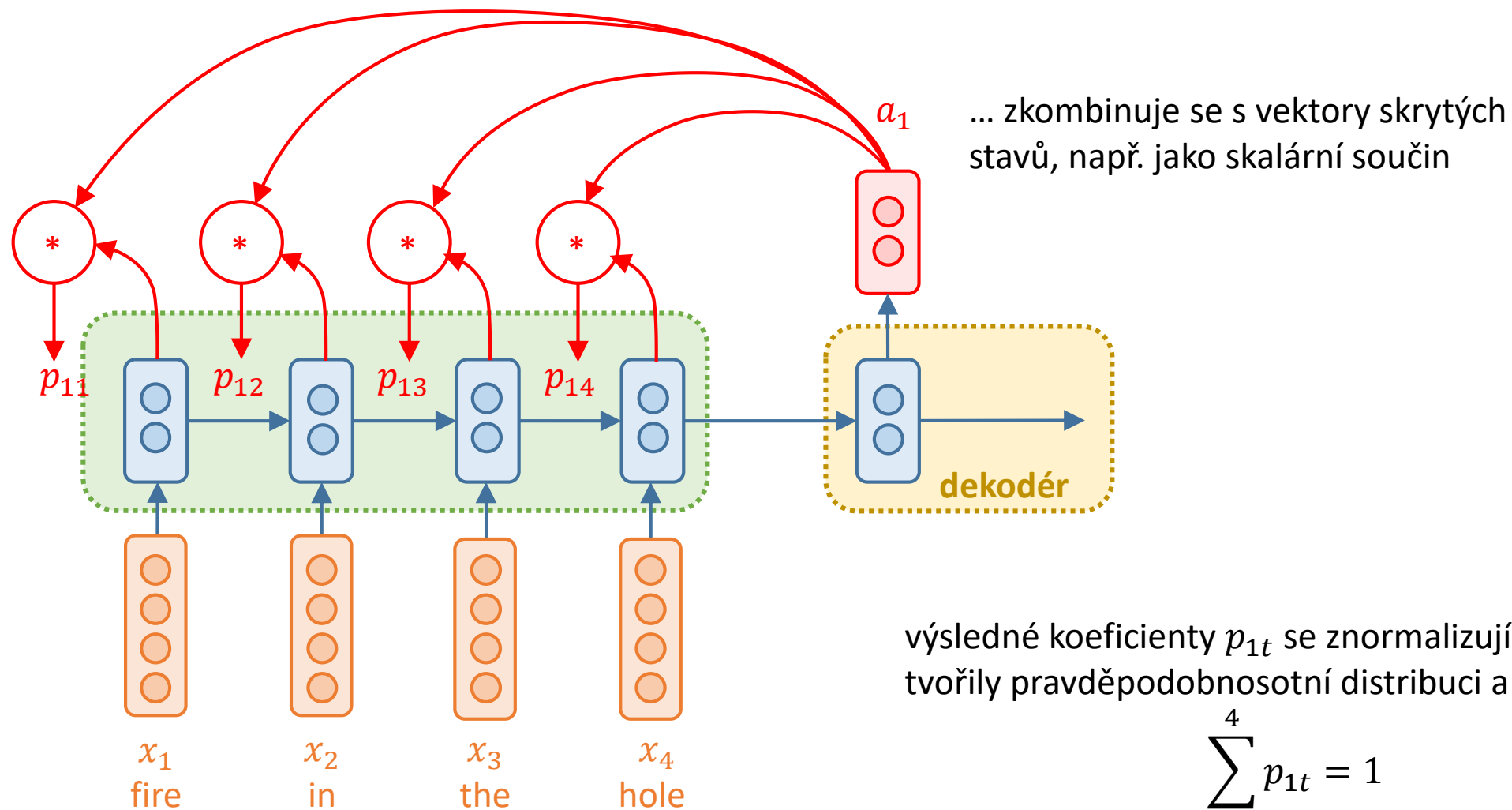
Princip attention pro strojový překlad



Princip attention pro strojový překlad

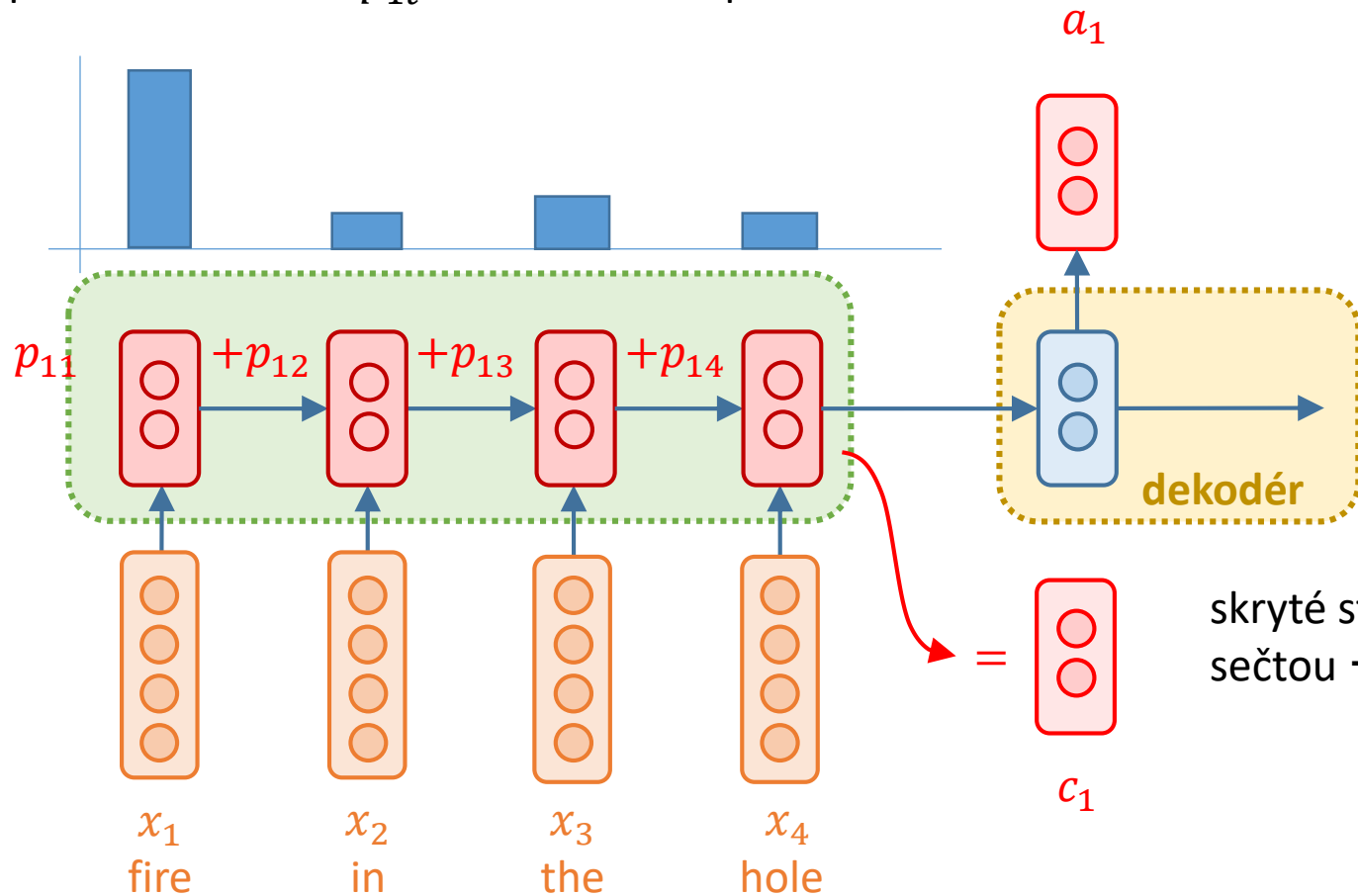


Princip attention pro strojový překlad

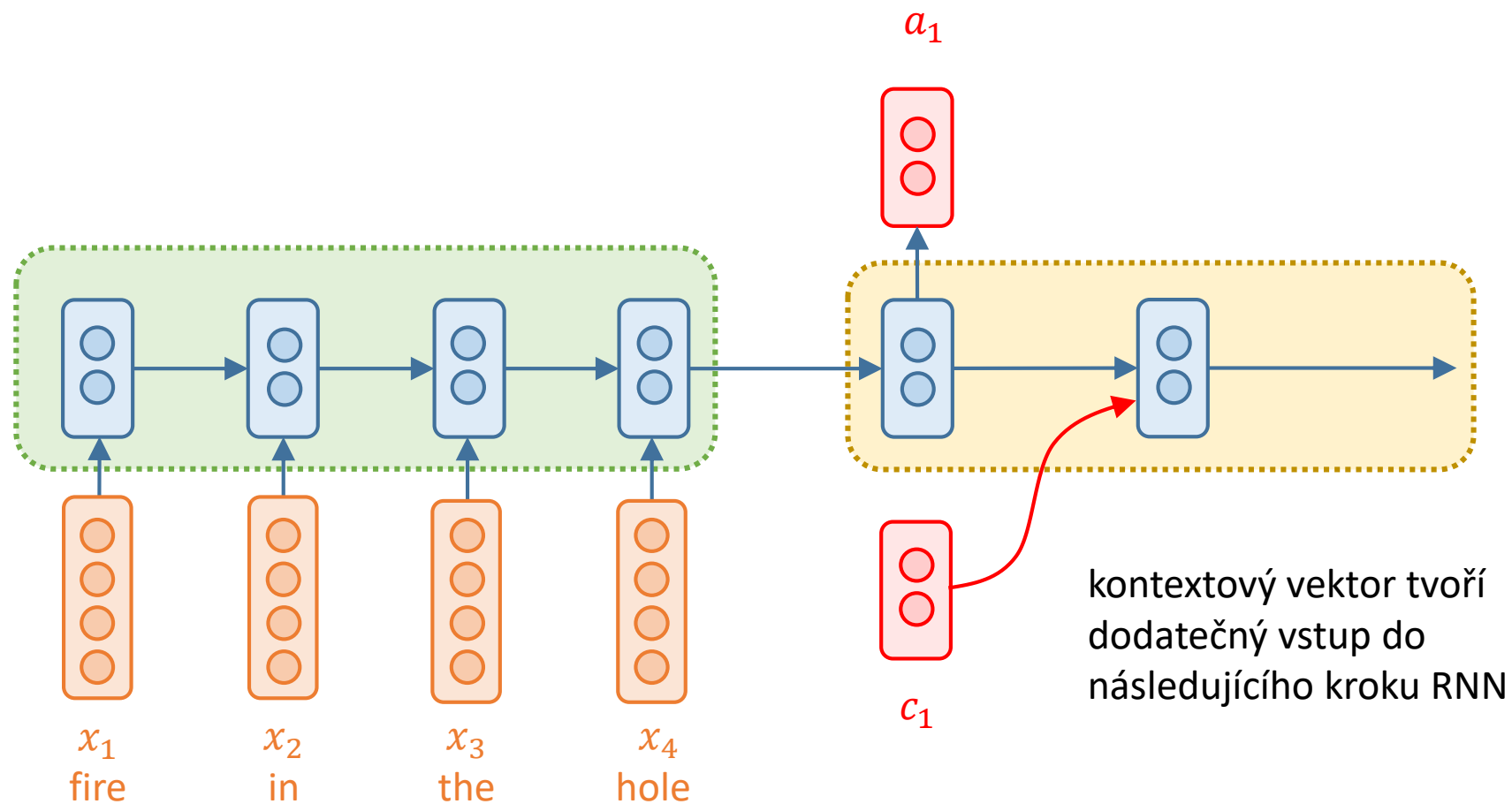


Princip attention pro strojový překlad

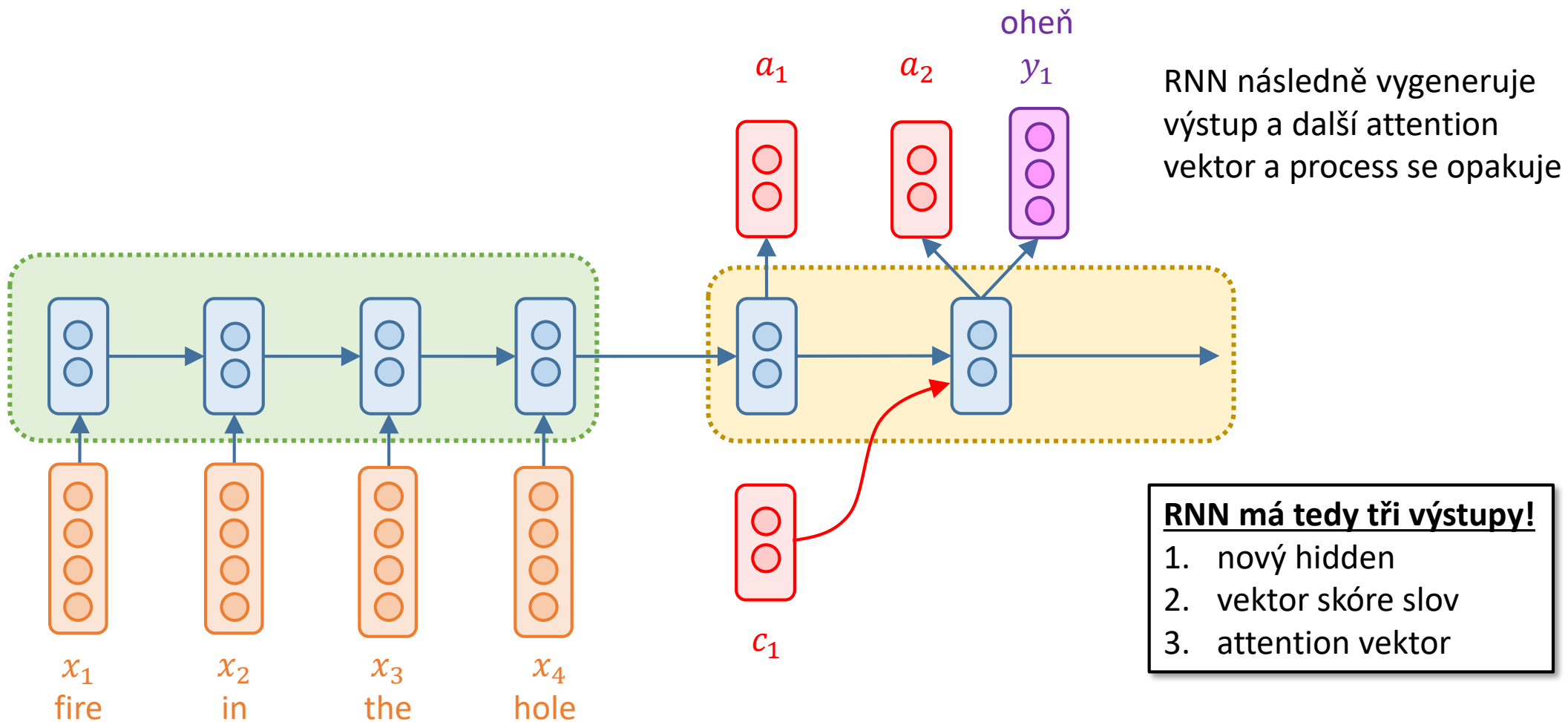
tam, kde jsou si attention vektor a_1 a skrytý stav h_t podobné \rightarrow velké p_{1t} = “soustředění pozornosti”



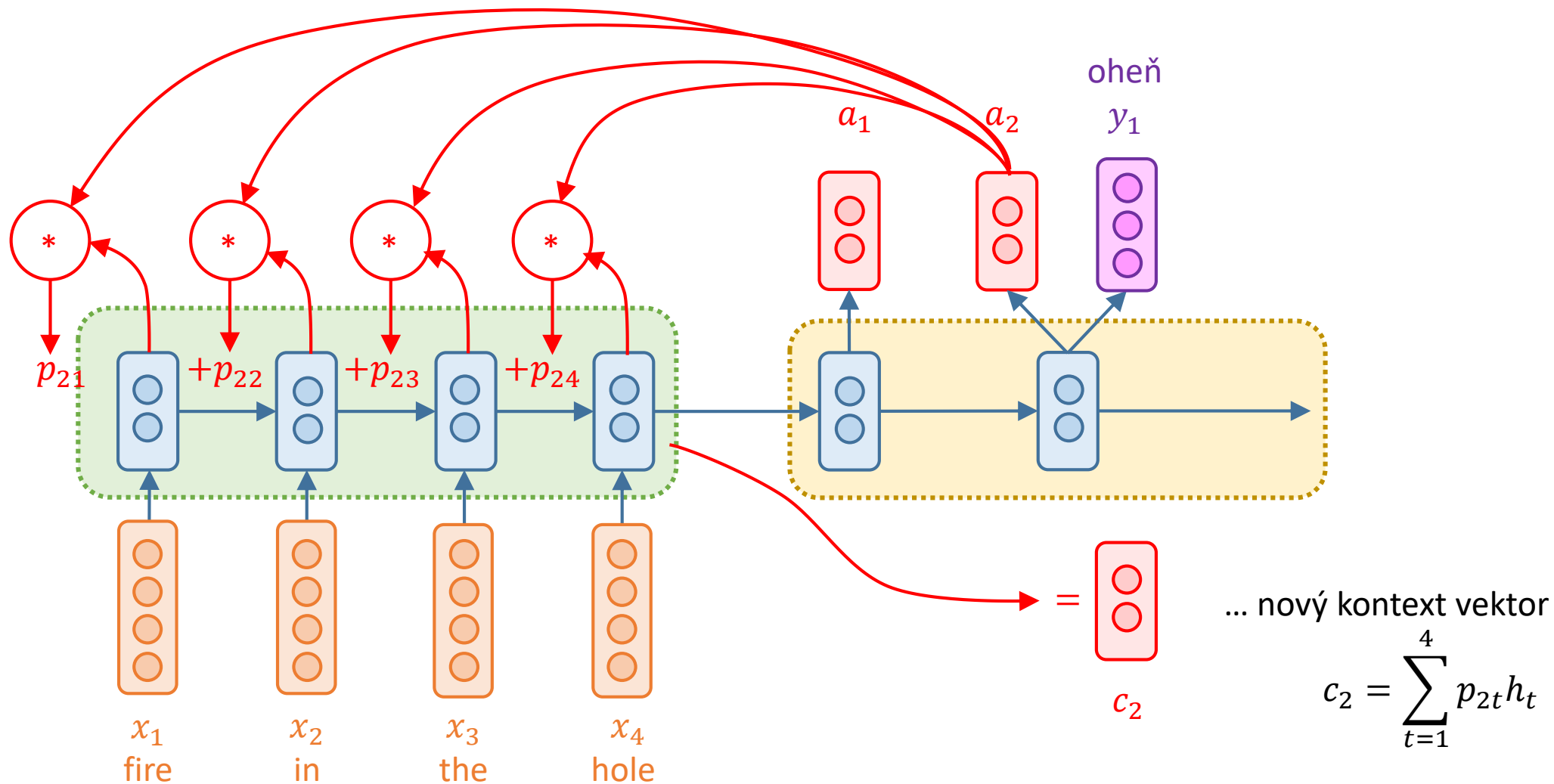
Princip attention pro strojový překlad



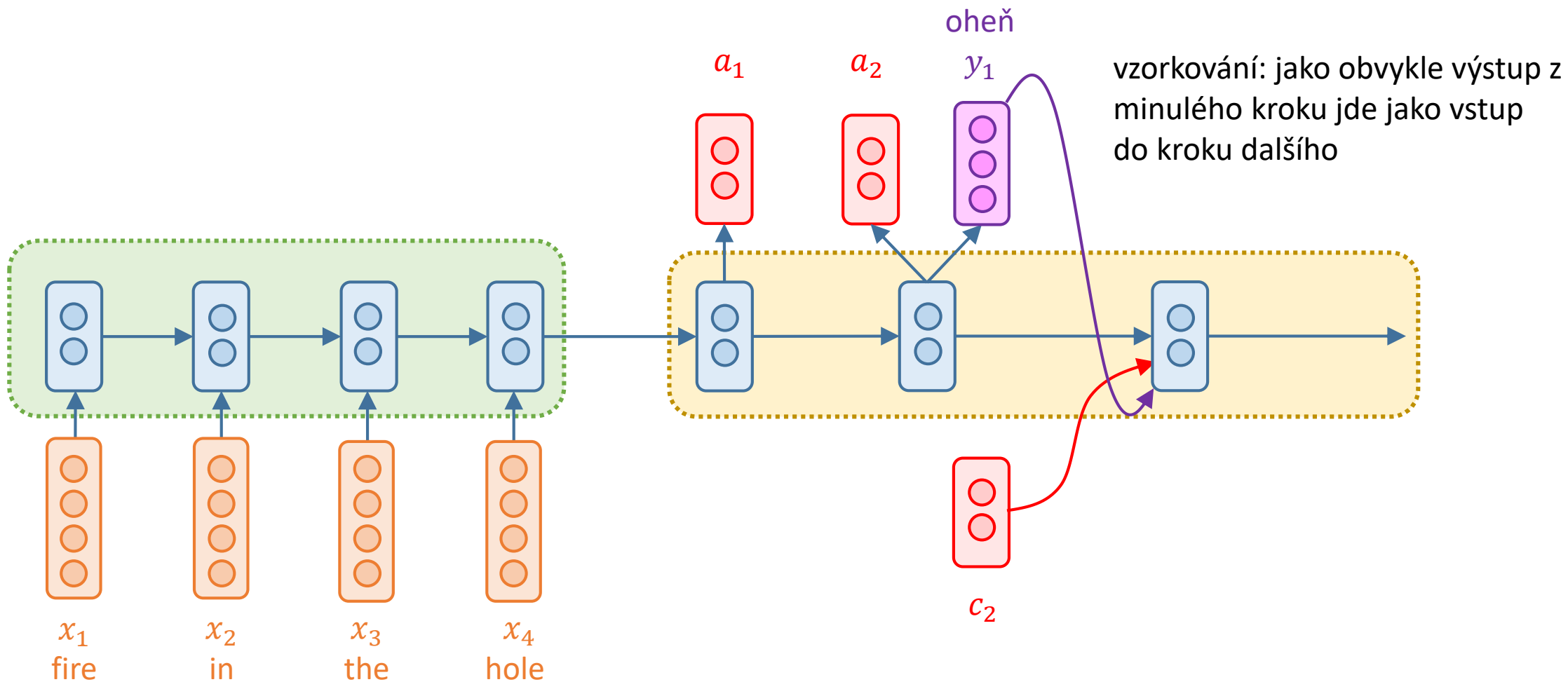
Princip attention pro strojový překlad



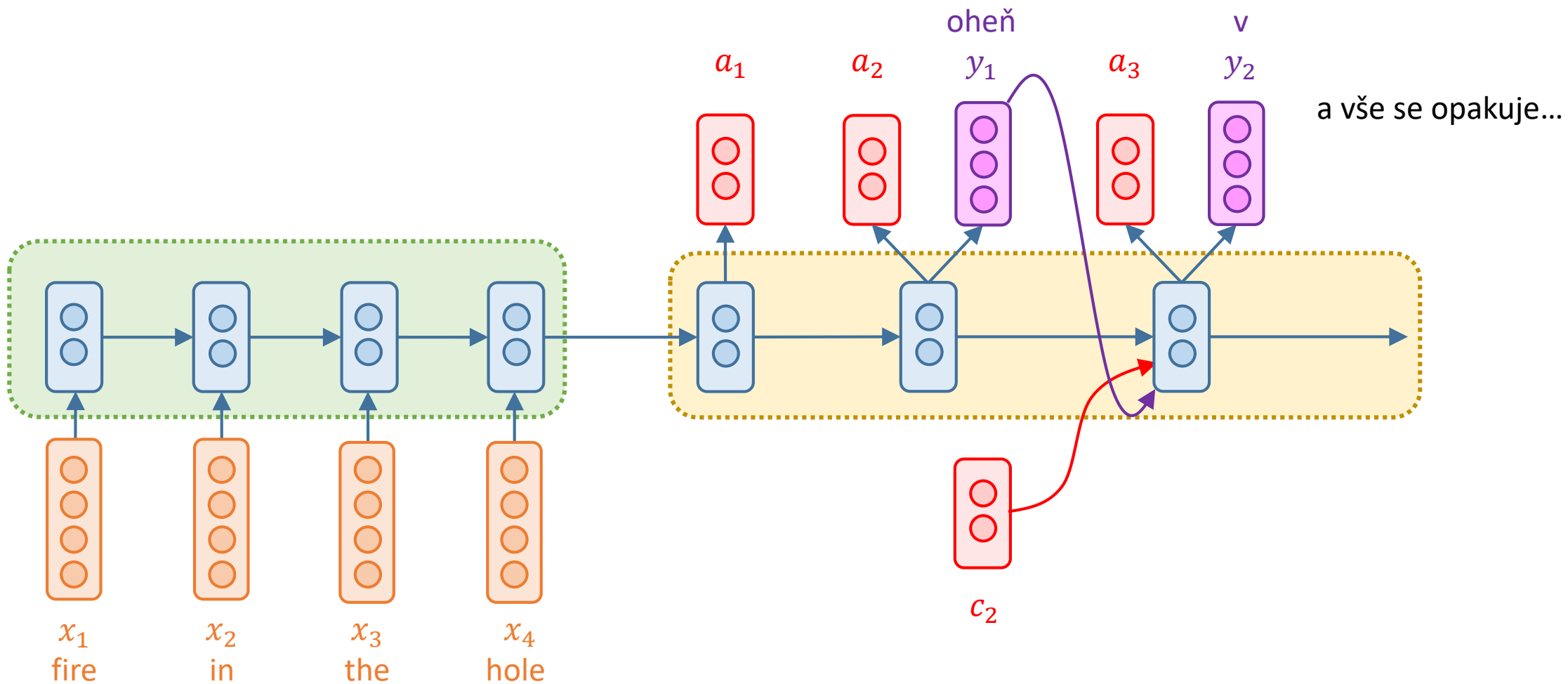
Princip attention pro strojový překlad



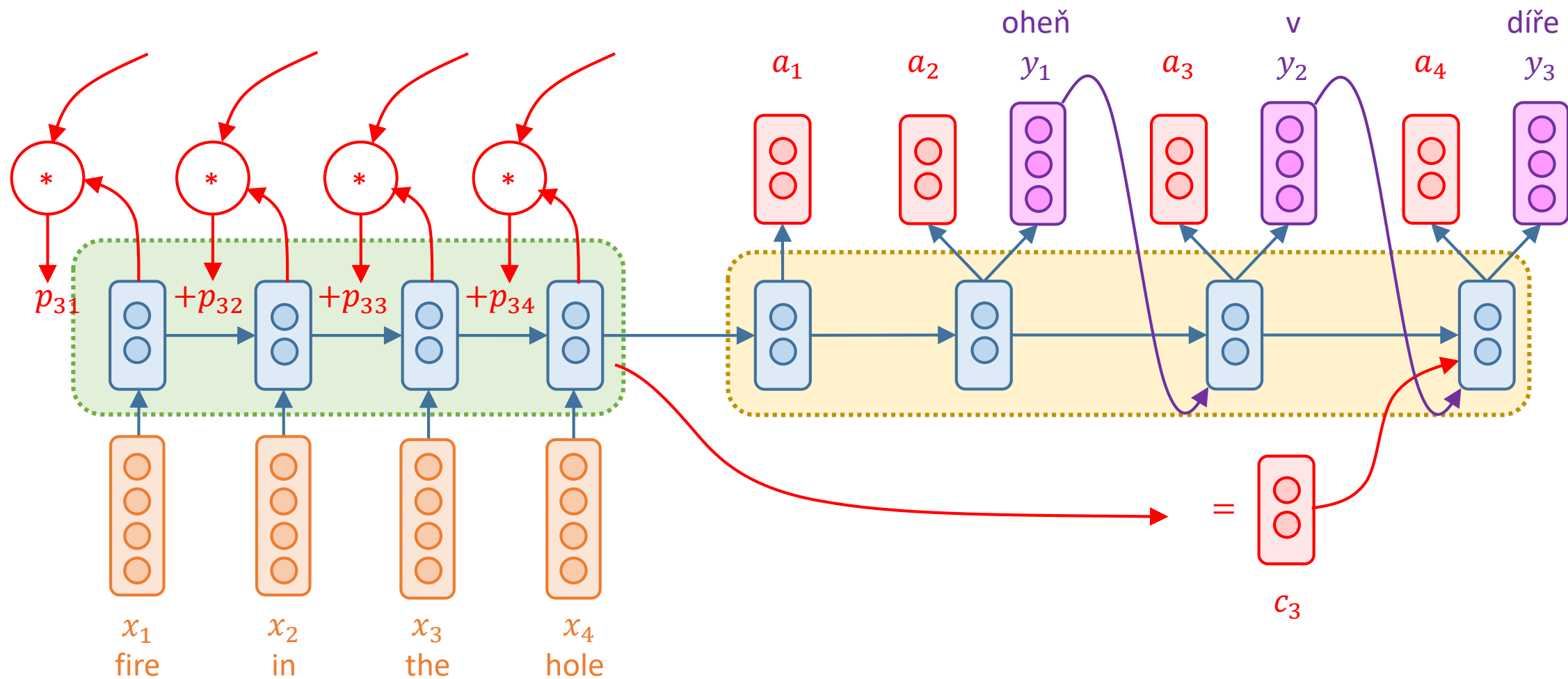
Princip attention pro strojový překlad



Princip attention pro strojový překlad

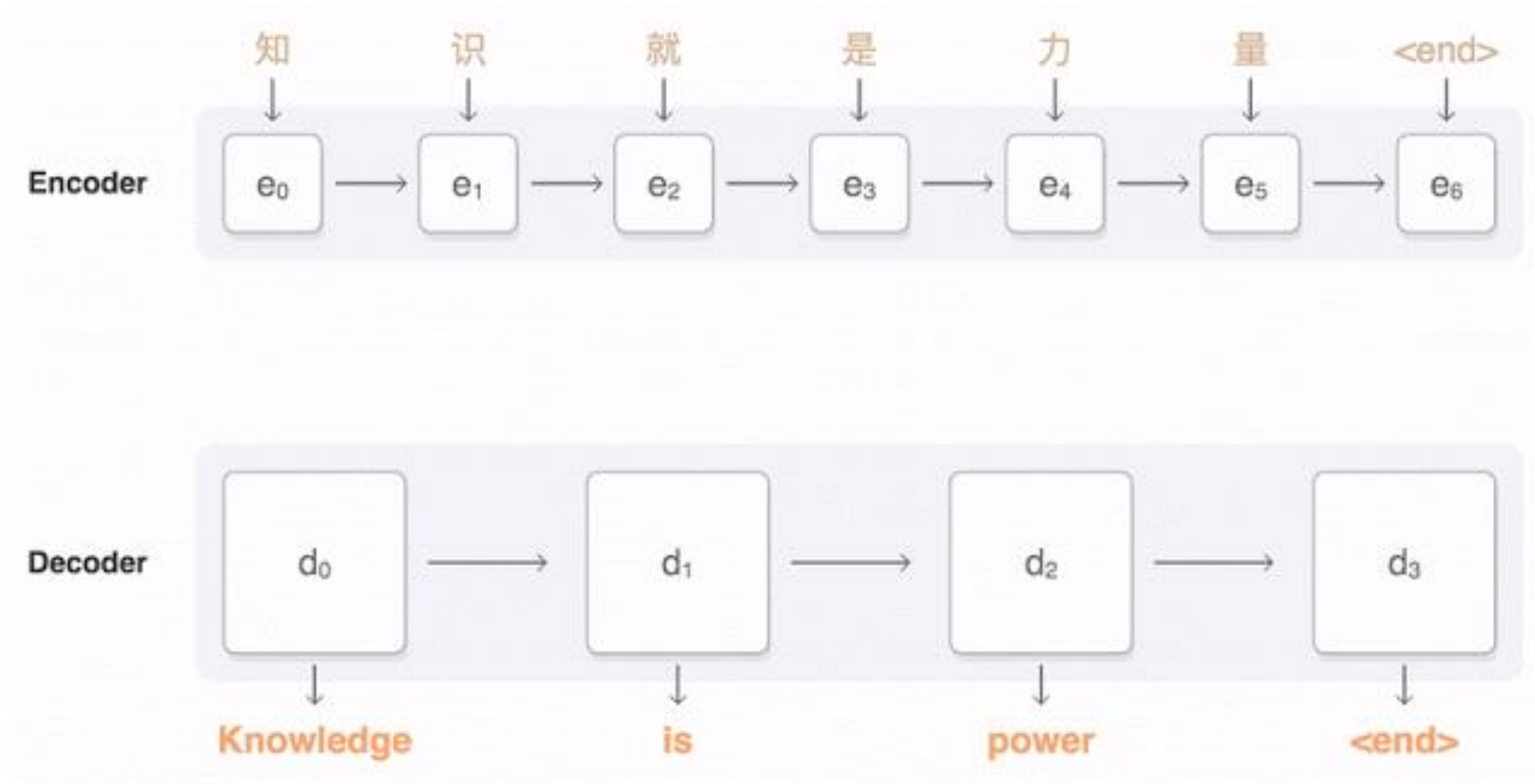


Princip attention pro strojový překlad



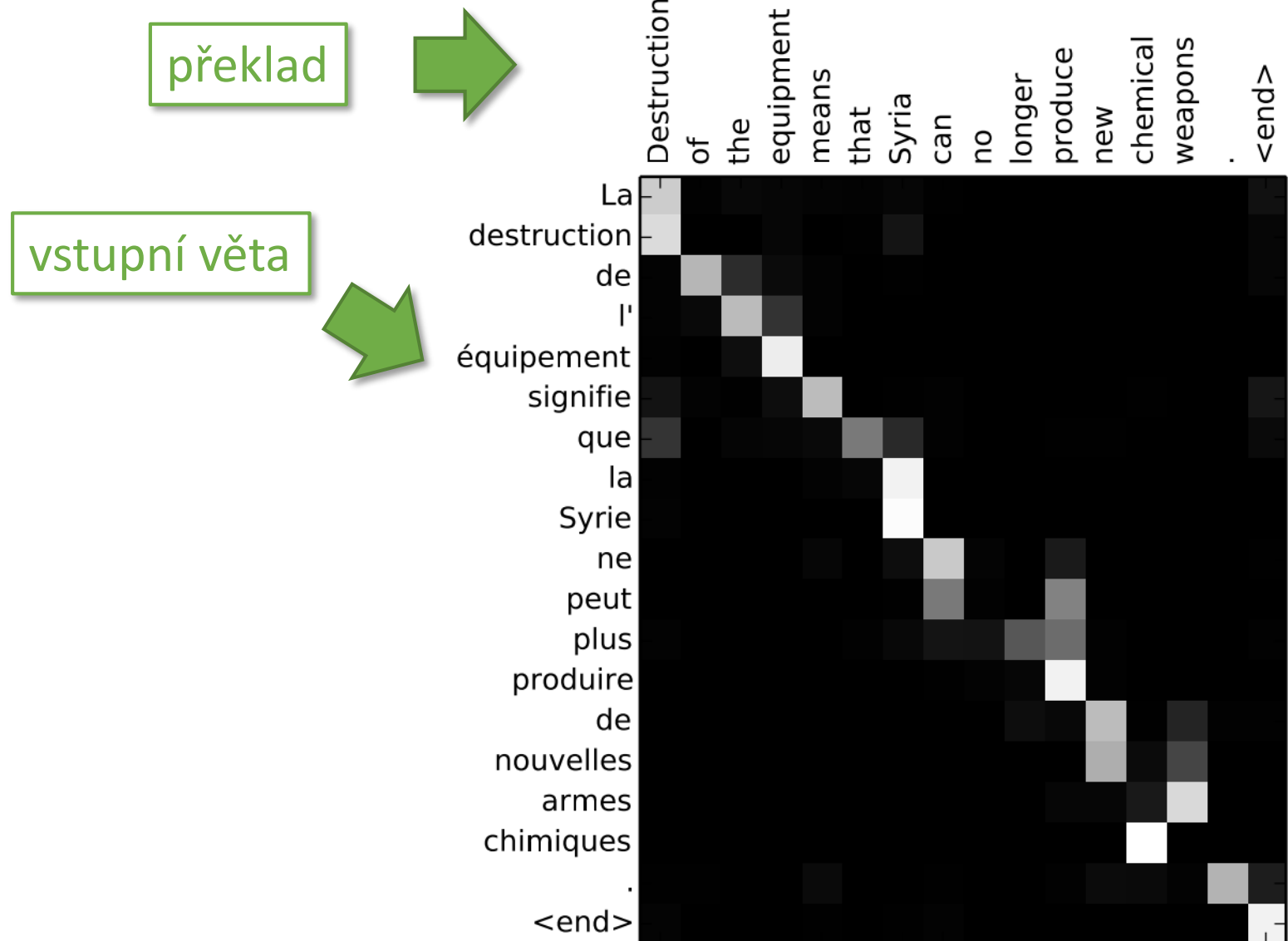
Attention pro strojový překlad: animace

(ne)průhlednost čar znázorňuje váhu = výsledné koeficienty p_t



animace: <https://research.googleblog.com/2016/09/a-neural-network-for-machine.html>

Vizualizace attention vah pro strojový překlad



- vysoké váhy znázorňují, na která slova překládací síť zaměřuje pozornost
- na mechanismus lze nahlížet také jako na “soft” asociativní paměť
- attention vektor a je query, kontext vektor c pak navracená hodnota
- s rozdílem, že c je složeninou celé paměti danou vahami p

*Attention is all you need

Vaswani et al: Attention is all you need

- autoři úplně vynechali rekurentní sítě, vše řeší pouze attention!
- state of the art výsledky

Table 2: The Transformer achieves better BLEU scores than previous state-of-the-art models on the English-to-German and English-to-French newstest2014 tests at a fraction of the training cost.

Model	BLEU		Training Cost (FLOPs)	
	EN-DE	EN-FR	EN-DE	EN-FR
ByteNet [18]	23.75			
Deep-Att + PosUnk [39]		39.2		$1.0 \cdot 10^{20}$
GNMT + RL [38]	24.6	39.92	$2.3 \cdot 10^{19}$	$1.4 \cdot 10^{20}$
ConvS2S [9]	25.16	40.46	$9.6 \cdot 10^{18}$	$1.5 \cdot 10^{20}$
MoE [32]	26.03	40.56	$2.0 \cdot 10^{19}$	$1.2 \cdot 10^{20}$
Deep-Att + PosUnk Ensemble [39]		40.4		$8.0 \cdot 10^{20}$
GNMT + RL Ensemble [38]	26.30	41.16	$1.8 \cdot 10^{20}$	$1.1 \cdot 10^{21}$
ConvS2S Ensemble [9]	26.36	41.29	$7.7 \cdot 10^{19}$	$1.2 \cdot 10^{21}$
Transformer (base model)	27.3	38.1	$3.3 \cdot 10^{18}$	
Transformer (big)	28.4	41.8	$2.3 \cdot 10^{19}$	

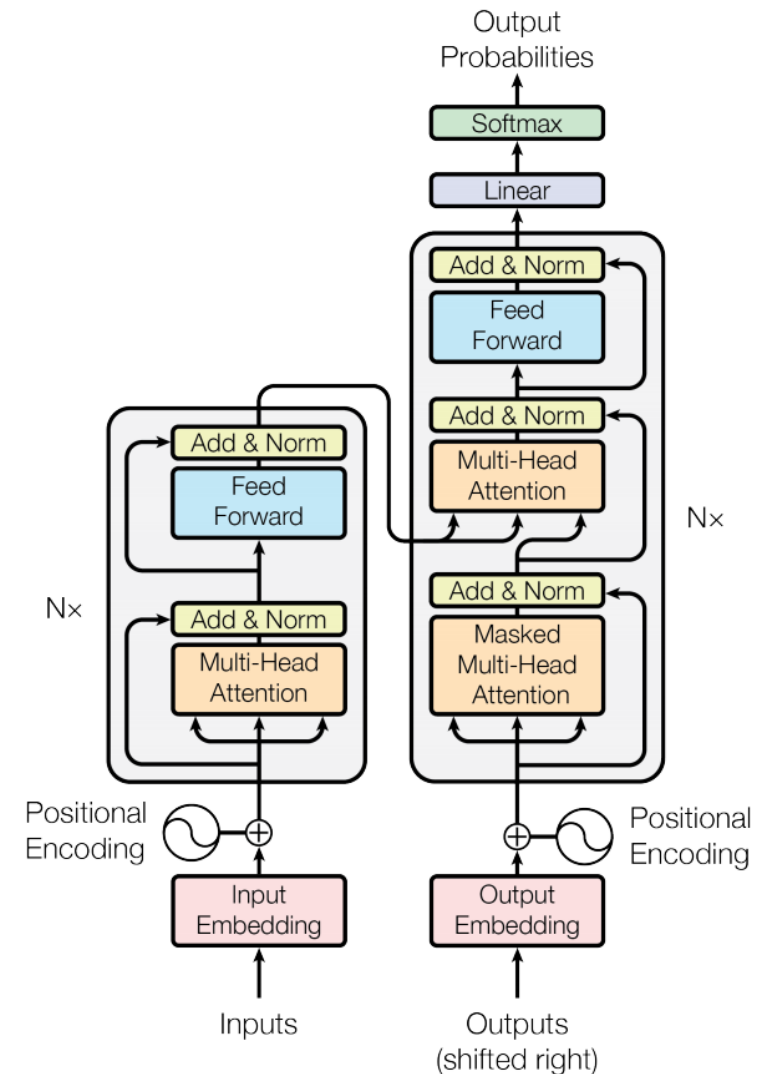
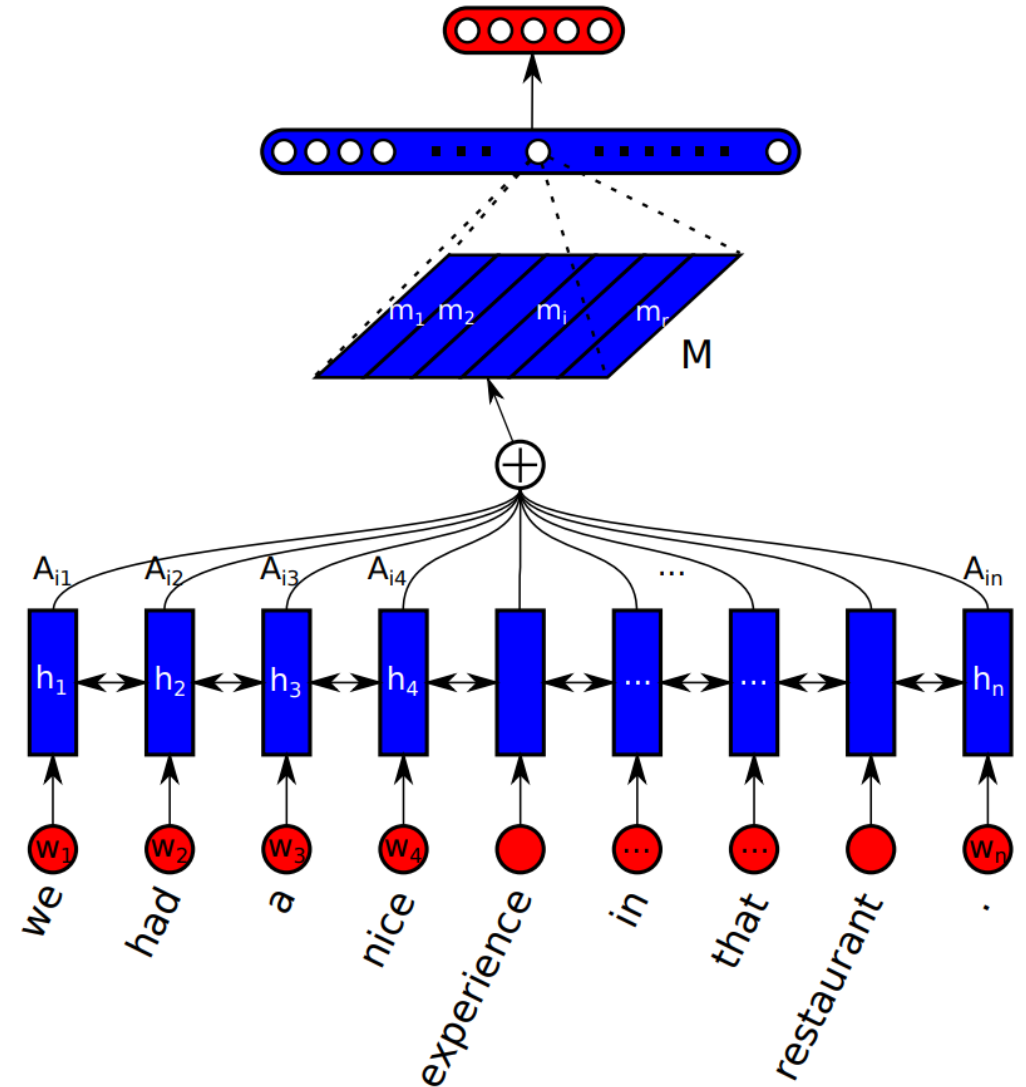


Figure 1: The Transformer - model architecture.

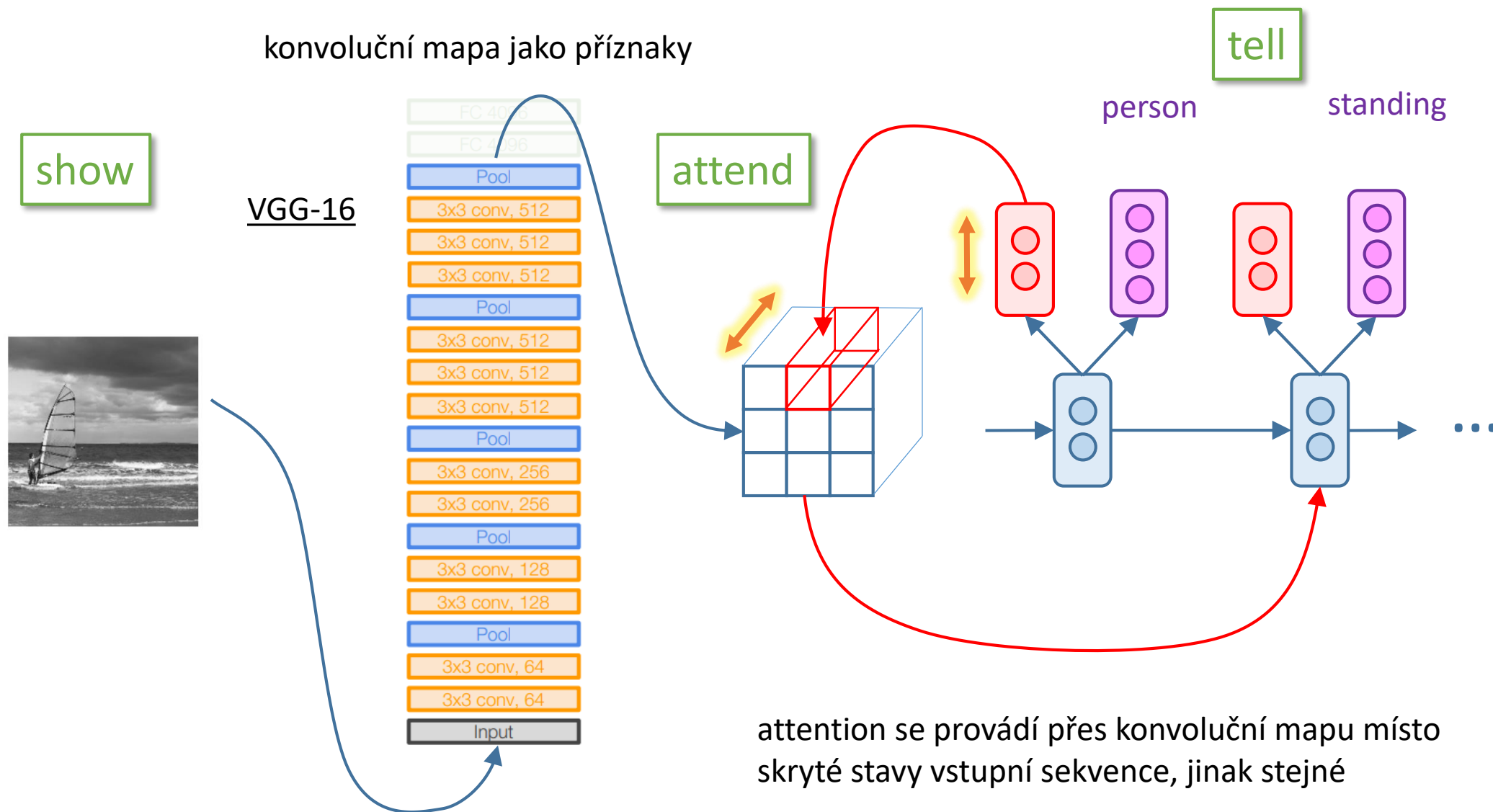
*Self attention

- Pro modely pouze s jedním výstupem se základní verze attention příliš nehodí
- Věta lze namísto vektoru reprezentovat jako matici (více vektorů) o fixní velikosti
- Kritérium navíc, které zajišťuje, aby každý sloupec M zachycoval něco jiného
- Klasifikuje se až výsledná matice



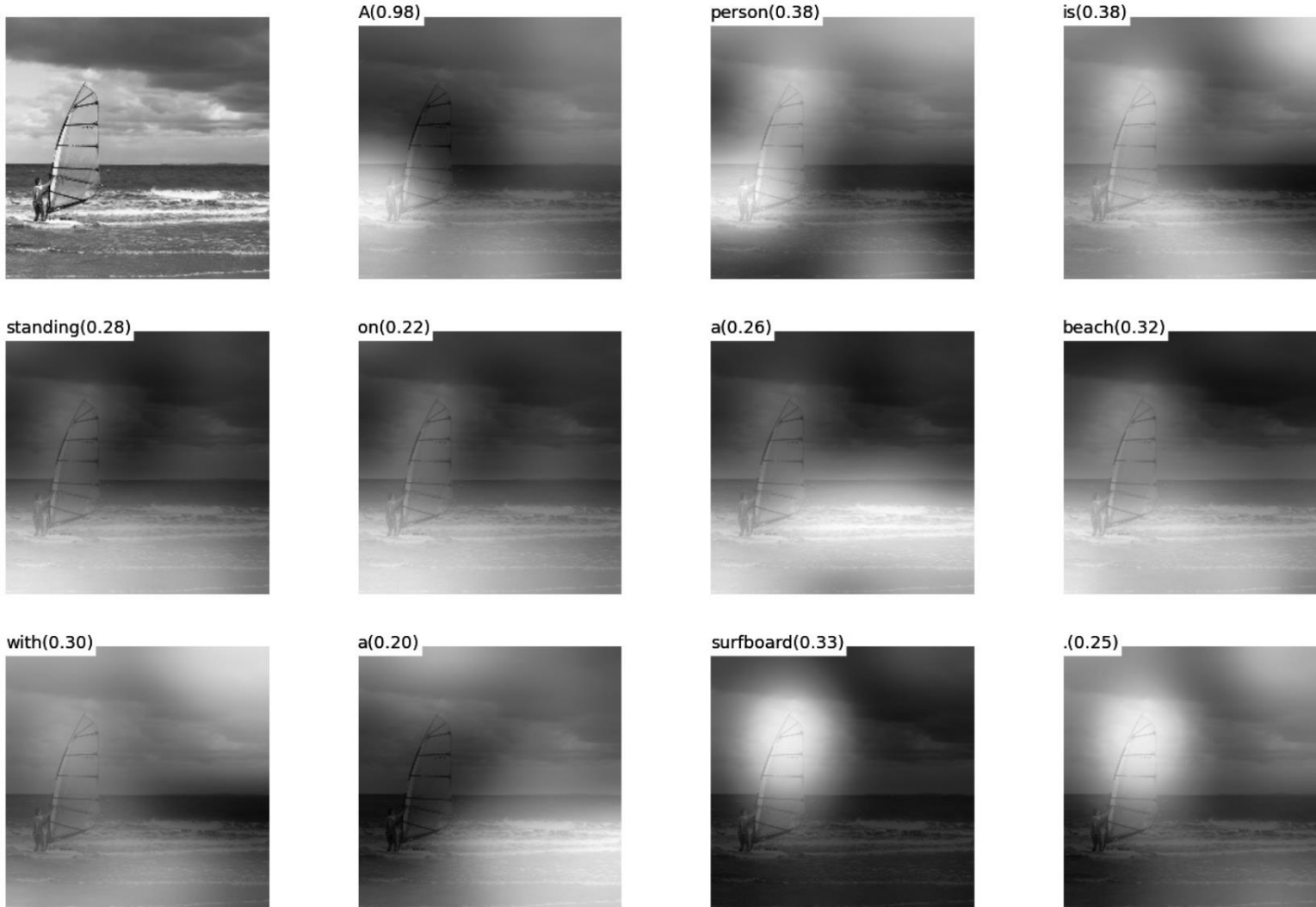
obrázek: [Lin et al: A Structured Self-attentive Sentence Embedding](#)

Attention pro obrazová data: show, attend and tell



např. článek: [Xu et al.: Show, Attend and Tell: Neural Image Caption Generation with Visual Attention](#)

Vizualizace attention vah pro obrazová data



- váhy jsou přes konvoluční mapy
- menší rozměr než vstupní obrázek
- z vrchní vrstvy zpětně dopočteno do obrázku
- obrázky znázorňují, na jaké části obrázku se síť při generování jednotlivých slov soustředí

(b) A person is standing on a beach with a surfboard.

Automatické odezírání ze rtů: listen/watch, attend and spell

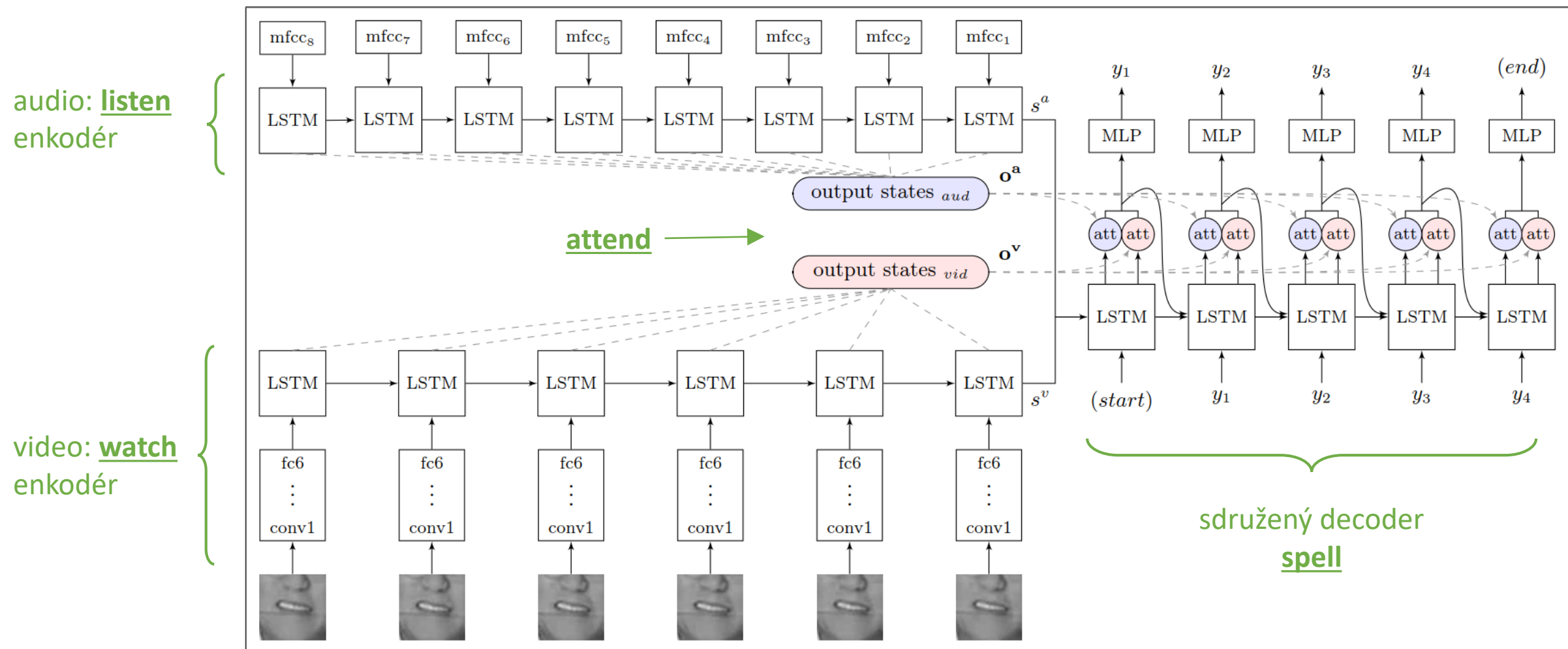


Figure 1. Watch, Listen, Attend and Spell architecture. At each time step, the decoder outputs a character y_i , as well as two attention vectors. The attention vectors are used to select the appropriate period of the input visual and audio sequences.

Automatické odezírání ze rtů

člověk expert!

trénovací data

Channel	Series name	# hours	# sent.
BBC 1 HD	News [†]	1,584	50,493
BBC 1 HD	Breakfast	1,997	29,862
BBC 1 HD	Newsnight	590	17,004
BBC 2 HD	World News	194	3,504
BBC 2 HD	Question Time	323	11,695
BBC 4 HD	World Today	272	5,558
All		4,960	118,116

Table 1. Video statistics. The number of hours of the original BBC video; the number of sentences with full facetrack. [†]BBC News at 1, 6 and 10.

výsledky: CER, WER ... chybovost (méně = lépe)

Method	SNR	CER	WER	BLEU [†]
Lips only				
Professional [‡]	-	58.7%	73.8%	23.8
WAS	-	59.9%	76.5%	35.6
WAS+CL	-	47.1%	61.1%	46.9
WAS+CL+SS	-	44.2%	59.2%	48.3
WAS+CL+SS+BS	-	42.1%	53.2%	53.8
Audio only				
LAS+CL+SS+BS	clean	16.2%	26.9%	76.7
LAS+CL+SS+BS	10dB	33.7%	48.8%	58.6
LAS+CL+SS+BS	0dB	59.0%	74.5%	38.6
Audio and lips				
WLAS+CL+SS+BS	clean	13.3%	22.8%	79.9
WLAS+CL+SS+BS	10dB	22.8%	35.1%	69.6
WLAS+CL+SS+BS	0dB	35.8%	50.8%	57.5

Table 5. Performance on the LRS test set. **WAS**: Watch, Attend and Spell; **LAS**: Listen, Attend and Spell; **WLAS**: Watch, Listen, Attend and Spell; **CL**: Curriculum Learning; **SS**: Scheduled Sampling; **BS**: Beam Search. [†]Unigram BLEU with brevity penalty. [‡]Excluding samples that the lip reader declined to annotate. Including these, the CER rises to 78.9% and the WER to 87.6%.

Automatické odezírání ze rtů

příklady rozpoznaných vět **pouze z videa**

MANY MORE PEOPLE WHO WERE INVOLVED IN THE ATTACKS
CLOSE TO THE EUROPEAN COMMISSION'S MAIN BUILDING
WEST WALES AND THE SOUTH WEST AS WELL AS WESTERN SCOTLAND
WE KNOW THERE WILL BE HUNDREDS OF JOURNAL- ISTS HERE AS WELL
ACCORDING TO PROVISIONAL FIGURES FROM THE ELECTORAL COMMISSION
THAT'S THE LOWEST FIGURE FOR EIGHT YEARS
MANCHESTER FOOTBALL CORRESPONDENT FOR THE DAILY MIRROR
LAYING THE GROUNDS FOR A POSSIBLE SECOND REFERENDUM
ACCORDING TO THE LATEST FIGURES FROM THE OF- FICE FOR NATIONAL STATISTICS
IT COMES AFTER A DAMNING REPORT BY THE HEALTH WATCHDOG

Table 6. Examples of unseen sentences that WAS correctly predicts (lips only).

Shrnutí

- slova vhodné reprezentovat jako husté vektory
- nejrozšířenější metoda skipgram word2vec
- podobně jako u konv. sítí lze využít předtrénované word-vektory a příp. finetunit

aplikace a odpovídající architektura	
jazykový model	1:1
klasifikace sekvence, analýza sentimentu	M:1
popis obrázků	1:N
strojový překlad, klasifikace na úrovni snímků, chatbot	M:N

- pro fungování především M:1 a M:N systémů velmi důležitý attention mechanismus