2014 NIPS paper:

An Autoencoder Approach to Learning Bilingual Word Representations

By

Stanislas Lauly

Supervisor : Hugo Larochelle

UNIVERSITÉ DE SHERBROOKE

Work done with

- Sarath Chandar (*Indian Institute of Technology Madras, India*)

- Hugo Larochelle (*Université de Sherbrooke, Canada*)
- Mitesh M. Khapra (*IBM Research India*)
- Balaraman Ravindran (*Indian Institute of Technology Madras, India*)
- Vikas Raykar (*IBM Research India*)
- Amrita Saha (*IBM Research India*)

Outline

- Bilingual representation
  - We use it for cross-lingual classification task
- The models
  - Binary bag-of-words reconstruction training
  - Tree-based decoder training
- Cross-lingual classification task
- The data
- The results

Bilingual representation

- We want to learn text representation that is invariant to the language
  - Same text in two different languages
  - Same representation

- To do this we learn vectorial word representation

Bilingual representation: Cross-lingual classification task

- We want to classify documents for one language
- Labeled data available in an other language

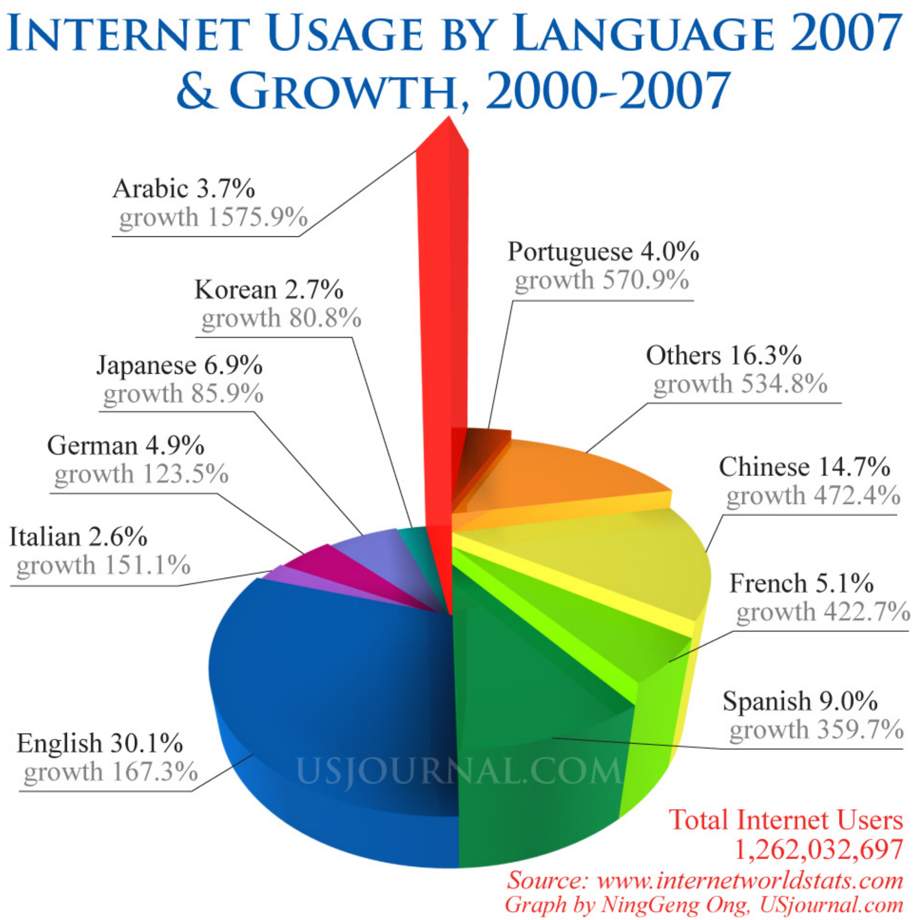
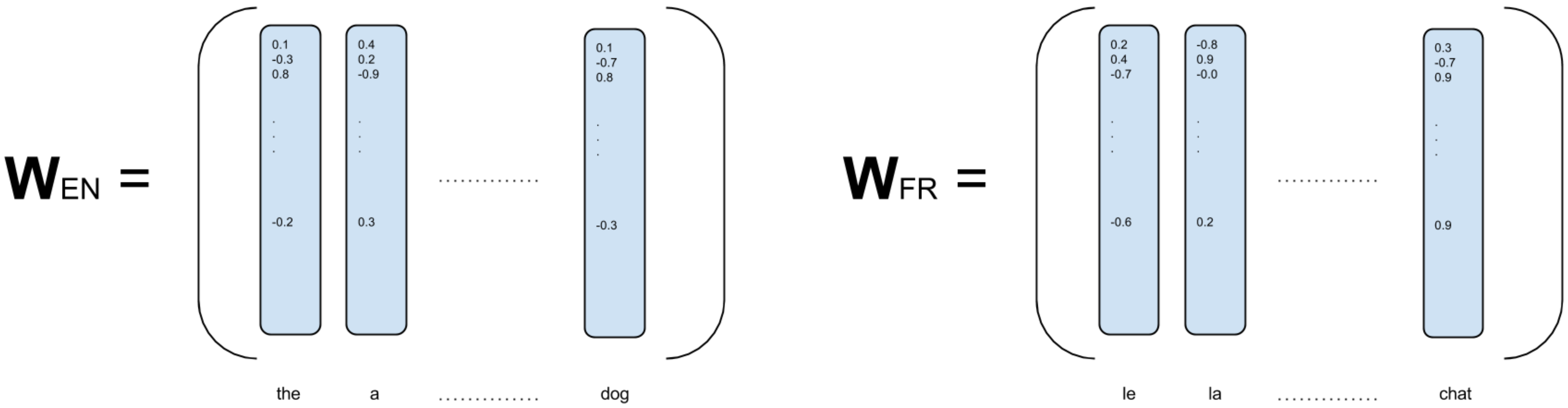


Image from Sarath Chandar

Bilingual representation

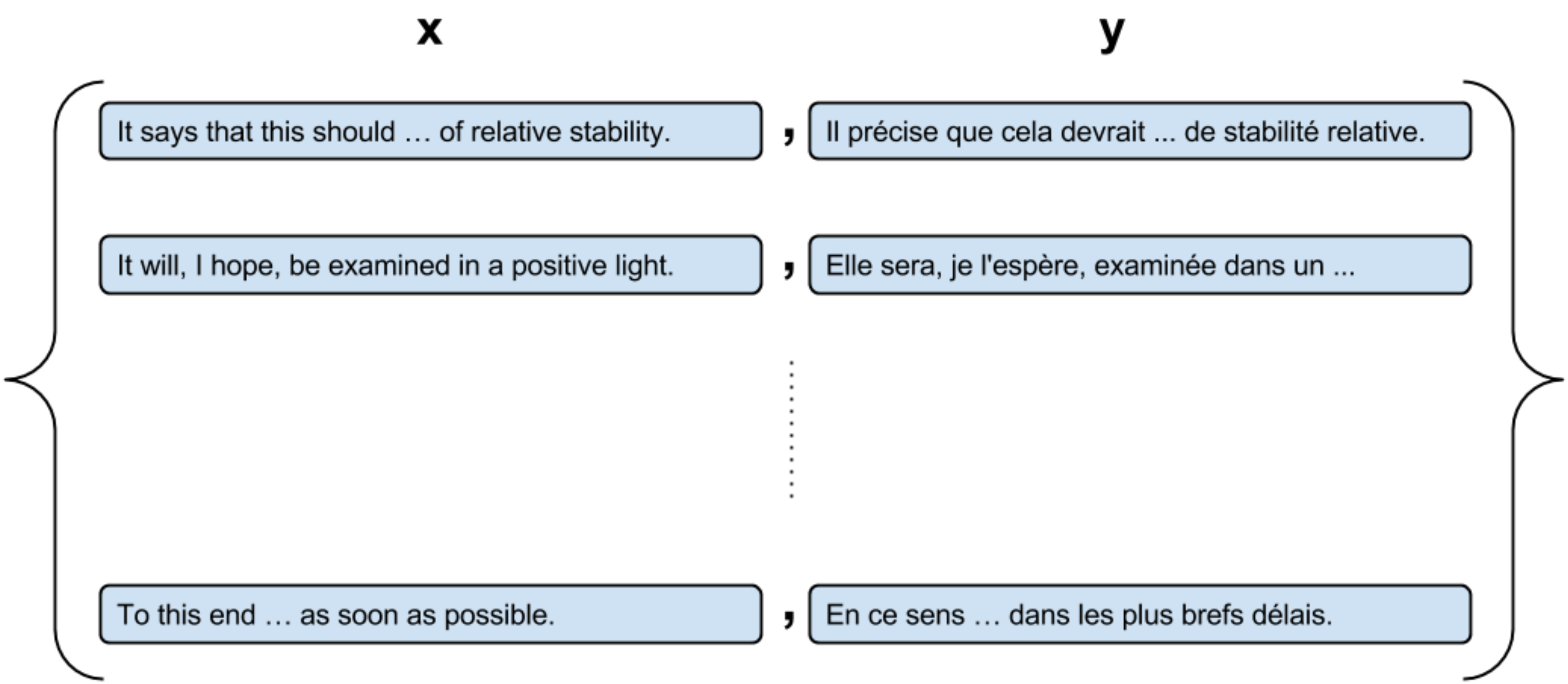
- Vectorial representation



- Text representation = sum of word vectors

Bilingual representation

- We use corpus of aligned sentences for languages **x** and **y**



The models

- We explore autoencoders to learn the vectorial representation of the words
- Two types of autoencoders
  - Predict a binary bag-of-words
  - Predict a multinomial representation of the text
- Word alignment is not needed
- We use one language to predict the other

Binary bag-of-words reconstruction training

- Use a binary bag-of-words representation $\mathbf{v}(\mathbf{x})$
- Encoder

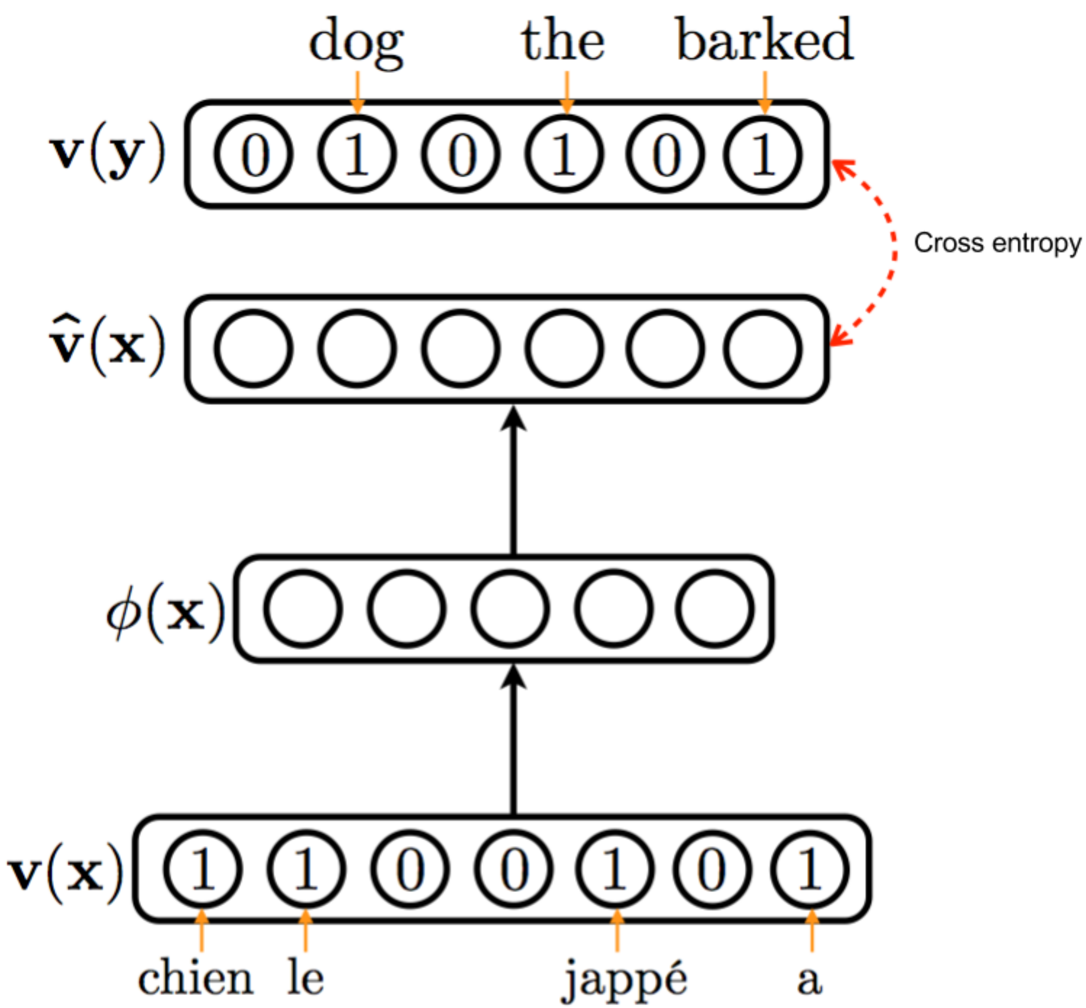
$$\mathbf{a}(\mathbf{x}) = \mathbf{c} + \mathbf{W}\mathbf{v}(\mathbf{x}), \;\; \phi(\mathbf{x}) = \mathbf{h}(\mathbf{a}(\mathbf{x}))$$

- Decoder

$$\widehat{\mathbf{v}}(\mathbf{x}) = \mathrm{sigm}(\mathbf{V}\phi(\mathbf{x}) + \mathbf{b})$$

- Cross entropy for the objective function

Binary bag-of-words reconstruction training



Tree-based decoder training

- Encoder

$$\mathbf{a}(\mathbf{x}) = \mathbf{c} + \sum_{i=1}^{|\mathbf{x}|} \mathbf{W}_{\cdot, x_i}, \quad \phi(\mathbf{x}) = \mathbf{h}(\mathbf{a}(\mathbf{x}))$$

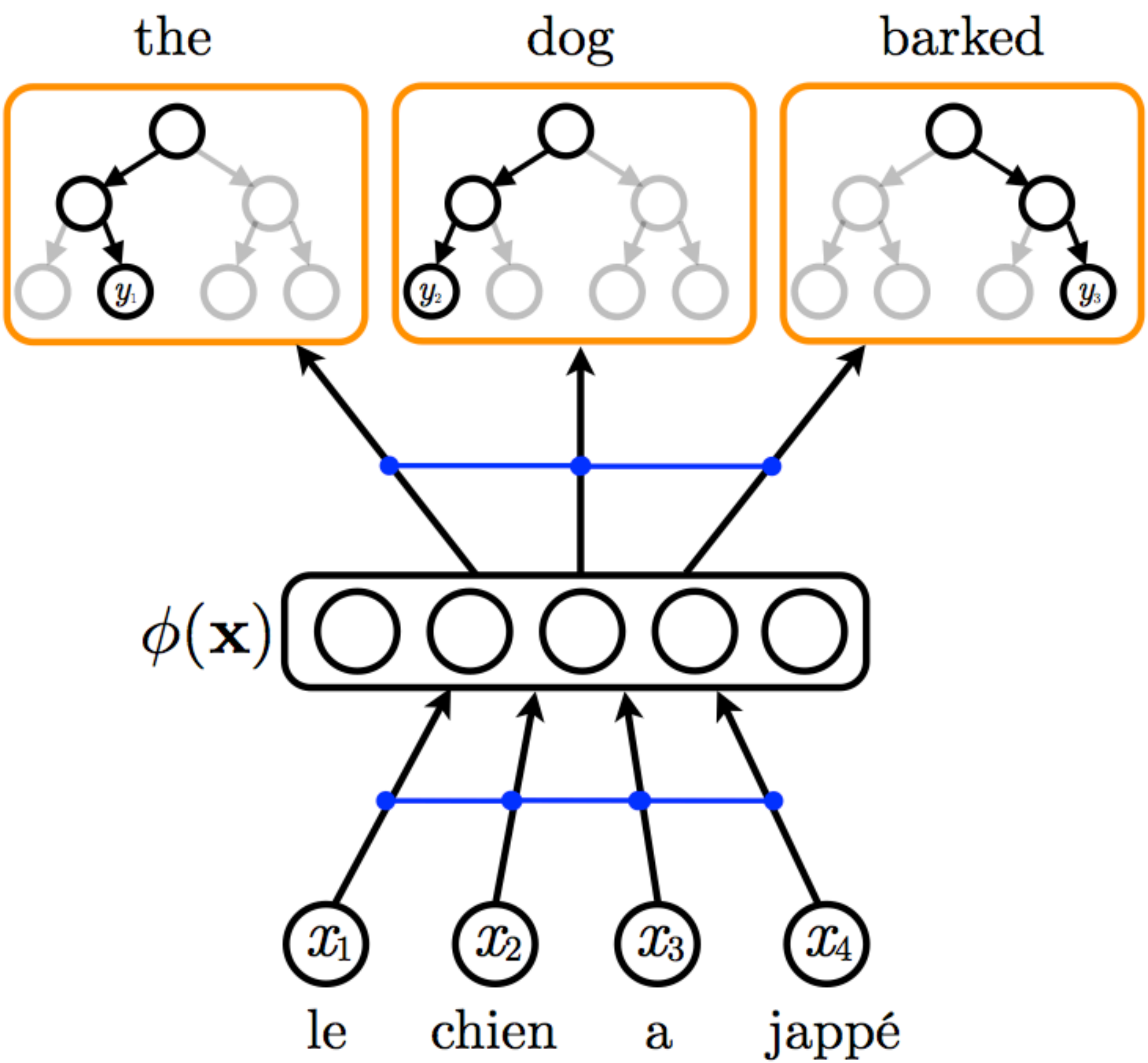
- Decoder: Probabilistic binary tree for computing

$$p(\widehat{x} = x_i | \phi(\mathbf{x}))$$

  - Each leaf of the tree is a word

- Minimize the negative log probability

Tree-based decoder training



The models

- Reconstruction
  - **x** to **x**
  - **y** to **y**
  - **x** to **y**
  - **y** to **x**
- We can take advantage of unaligned data

- Maximize the correlation between the dimensions of the embedding's learned between languages

Cross-lingual classification task

- We use the bilingual representation learned

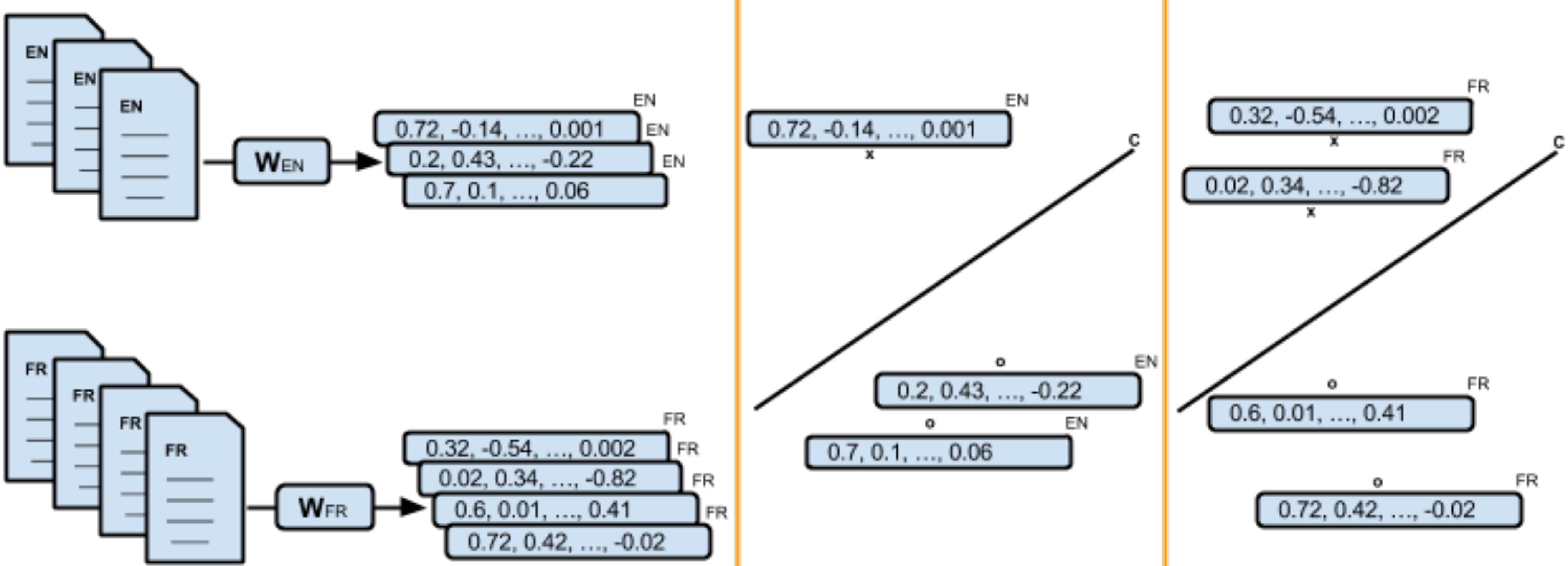
- We want to classify documents for one language
- Labeled data available in an other language
  - Learn a classifier with one language
  - Apply that classifier on the other language

Classification task

The model create a bilingual representation for documents in both languages

We learn a classifier on the representations of one language

We apply that classifier on the representations of the other language



Data

- Europarl
  - Transcript of the European parliament debate translated in multiple languages
  - Dataset for learning the bilingual representation
  - Each example is a pair (**x**, **y**) of the same sentence in two different language

- Data: Reuters RCV1/RCV2
  - Dataset for learning the classifier
  - Different documents labeled for each language

Results

- Cross-lingual classification accuracy for 3 different pairs of languages, with 1000 labeled examples.

	EN → DE	DE → EN	EN → FR	FR → EN	EN → ES	ES → EN
BAE-tr	81.8	60.1	70.4	61.8	**59.4**	60.4
BAE-cr	**91.8**	**74.2**	**84.6**	**74.2**	49.0	**64.4**
Klementiev et al.	77.6	71.1	74.5	61.9	31.3	63.0
MT	68.1	67.4	76.3	71.1	52.0	58.4
Majority Class	46.8	46.8	22.5	25.0	15.3	22.2

Results

- Cross-lingual classification accuracy for 1000 labeled examples (with 500K in training set).

	EN -> DE	DE -> EN
Hermann and Blunsom	83.7%	71.4%
BAE-cr	**87.9%**	**76.7%**

Results

- Example English words along with the closest words both in English (EN) and German (DE), using the Euclidean distance between the embeddings learned by BAE-cr.

Word	Lang	Nearest neighbors	Word	Lang	Nearest neighbors
january	EN	january, march, october	oil	EN	oil, supply, supplies, gas, fuel
	DE	januar, märz, oktober		DE	öl, boden, befindet, gerät, erdöl
president	EN	president, i, mr, presidents	microsoft	EN	microsoft, cds, insider, ibm
	DE	präsident, präsidentin		DE	microsoft, cds, warner
said	EN	said, told, say, believe	market	EN	market, markets, single
	DE	gesagt, sagte, sehr, heute		DE	markt, marktes, märkte

NADE Language Model

- Combine two approach
  - DocNADE (A Neural Autoregressive Topic Model)
  - Neural language model

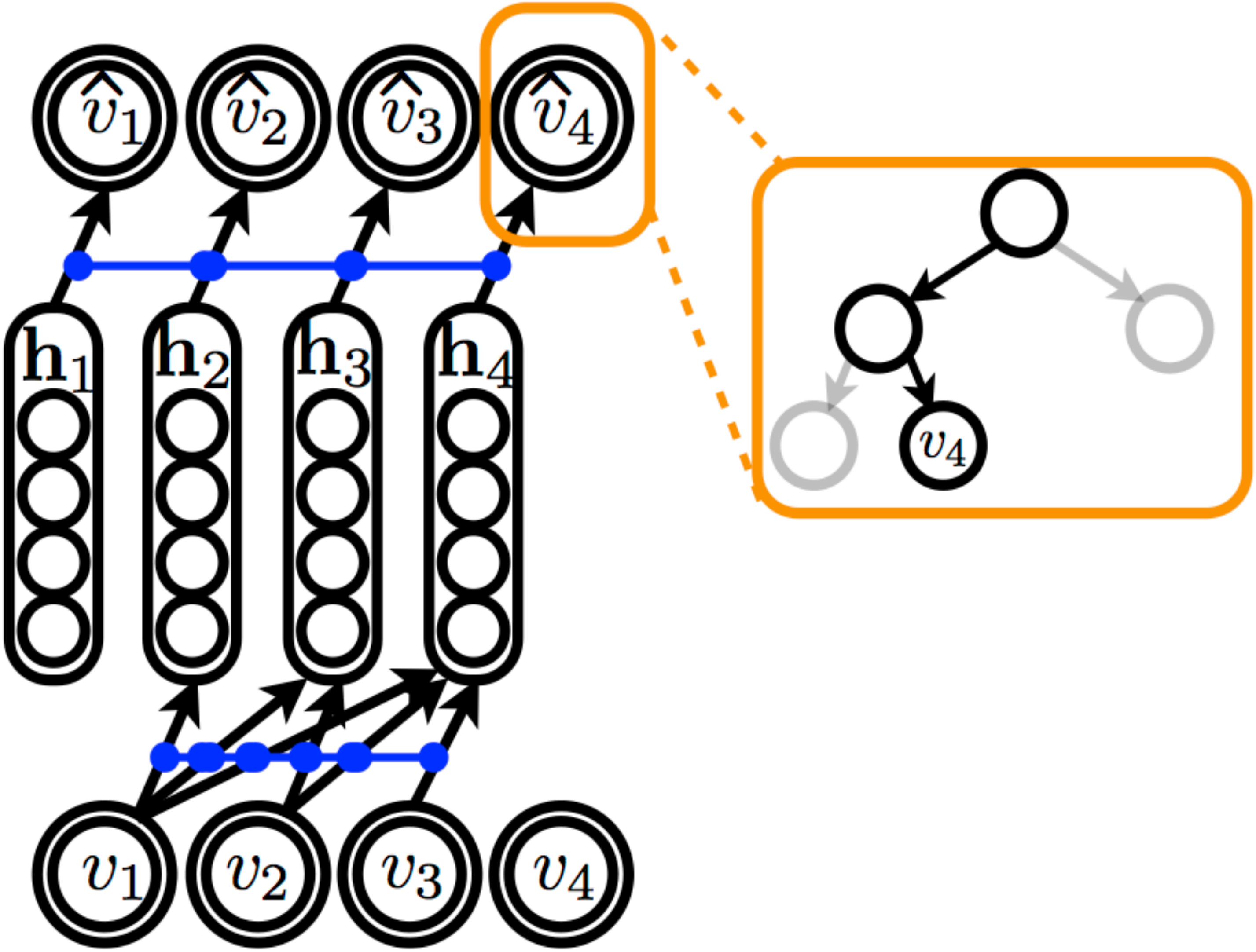
DocNADE (A Neural Autoregressive Topic Model)

- Generative model.
- Multinomial observations $\mathbf{v} \in \{1, 2, \ldots, N\}^D$.
- Extract a representation $\mathbf{h}_i$ of all previous words $\mathbf{v}_{<i}$ for each v_i .

$$\mathbf{h}_i(\mathbf{v}_{<i}) = sigm(\mathbf{c} + \sum_{k<i} W_{:,v_k})$$

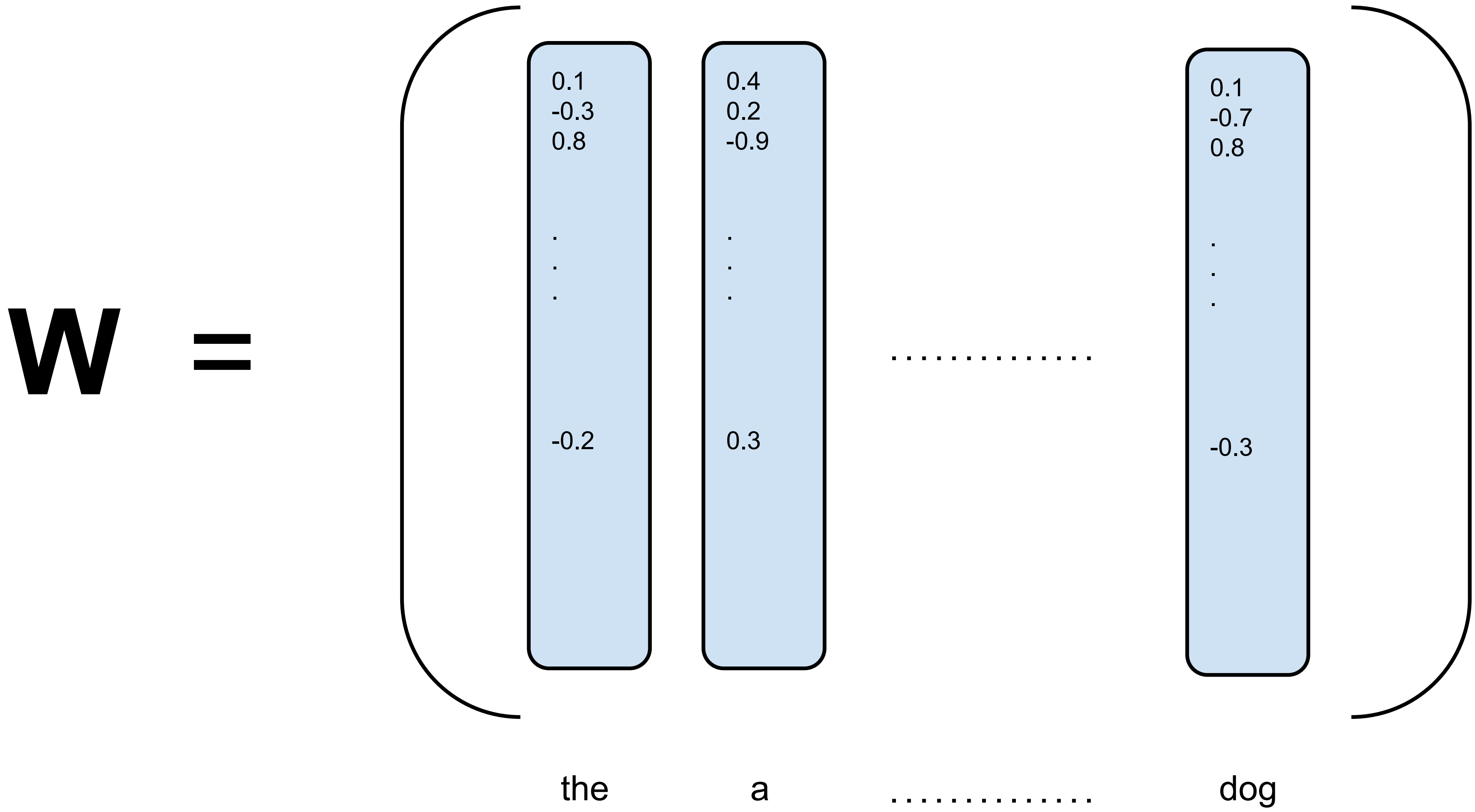
$$\mathbf{h}_{i+1}(\mathbf{v}_{<i+1}) = sigm(W_{:,v_i} + \mathbf{c} + \sum_{k<i} W_{:,v_k})$$

DocNADE

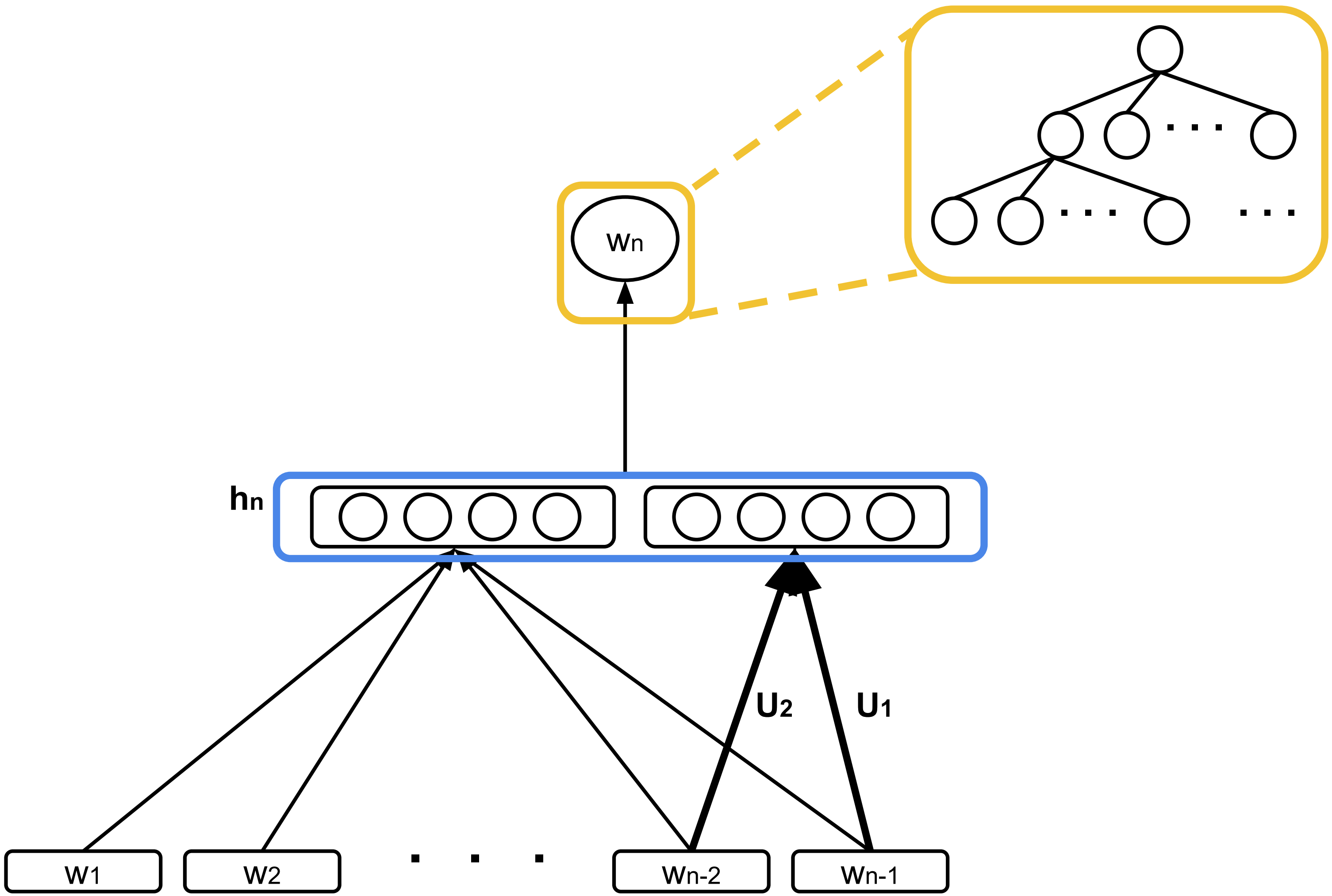


$\hat{v}_1$ $\hat{v}_2$ $\hat{v}_3$ $\hat{v}_4$
$\mathbf{h}_1$ $\mathbf{h}_2$ $\mathbf{h}_3$ $\mathbf{h}_4$
v_1 v_2 v_3 v_4
v_4

Vectorial representation



NADE Language Model



Preliminary results

- Perplexity on AP news dataset

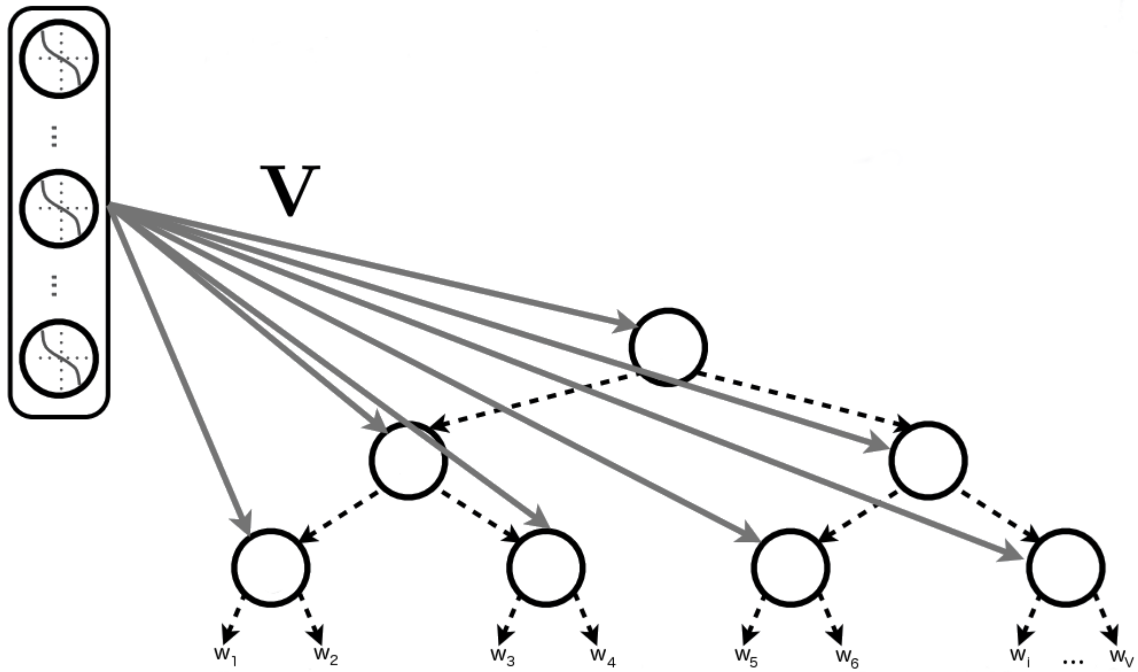
	Perplexity
Mnih & Hinton	112.1
NADE language model	113.8

Thank you !

Question time !

Tree-based decoder training

- Each leaf of the tree is a word
- $\mathbf{l}(w)$ is the sequence or nodes for the word w
- $\boldsymbol{\pi}(w)$ is the sequence of left/right choice
- $\mathrm{l}(w)_1$ is the root $\pi(w)_1$ is its left/right choice



Tree-based decoder training

- $V_{l(w)_m,:}$ is the binary logistic regression for the node $l(w)_m$

$$p(w \mid \boldsymbol{h}) = \prod_{m=1}^{|\boldsymbol{\pi}(w)|} p(\pi(w)_m \mid \boldsymbol{h})$$

$$p(\pi(w)_m = 1 \mid \boldsymbol{h}) = sigm(b_{l(w)_m} + V_{l(w)_m,:}\boldsymbol{h})$$