Distributed representations and language processing

Liangjie Cao

11 May 2018

Today I learn Distributed representations and language processing. The paper says Deep learning theory shows that deep nets have two different exponential advantages over classic learning algorithms that do not use distributed representations. Both of these advantages arise from the power of composition and depend on the underlying data-generating distribution having an appropriate componential structure. [1] Firstly, learning distributed representations enable generalization to new combinations of the values of learned features beyond those seen during training. Secondly, composing layers of representation in a deep net brings the potential for another exponential advantage.

The paper gives a concrete example to us. It takes the content of local text as input, training multilayer neural network to predict the next word in the sentence. Each word in the content is represented as a vector of one of n points in the network. That is to say, there is one value of 1 in each component and the rest is all 0. Each word in the context is presented to the network as a one-of-N vector, that is, one component has a value of 1 and the rest are 0. In the first layer, each word creates a different pattern of activations, or word vectors (Fig. 1). The network learns word vectors that contain many active components each of which can be interpreted as a separate feature of the word, as was first demonstrated in the context of learning distributed representations for symbols.

The professors say the issue of representation lies at the heart of the debate between the logic-inspired and the neural-network-

inspired paradigms for cognition. Then Before the introduction of neural language models71, the standard approach to statistical modelling of language did not exploit distributed representations: it was based on counting frequencies of occurrences of short symbol sequences of length up to N (called N-grams). Typical N-grams(Figure. 2) model can be seen at the Table 2 and Table 1 called Bi-gram model. N-grams treat each word as an atomic unit, so they cannot generalize across semantically related sequences of words, whereas neural language models can because they associate each word with a vector of real valued features, and semantically related words end up close to each other in that vector space. What an amazing work process.

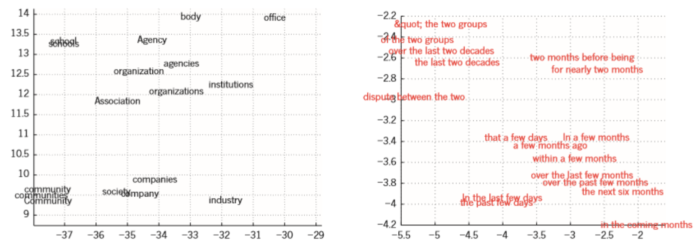

Figure 1: Model

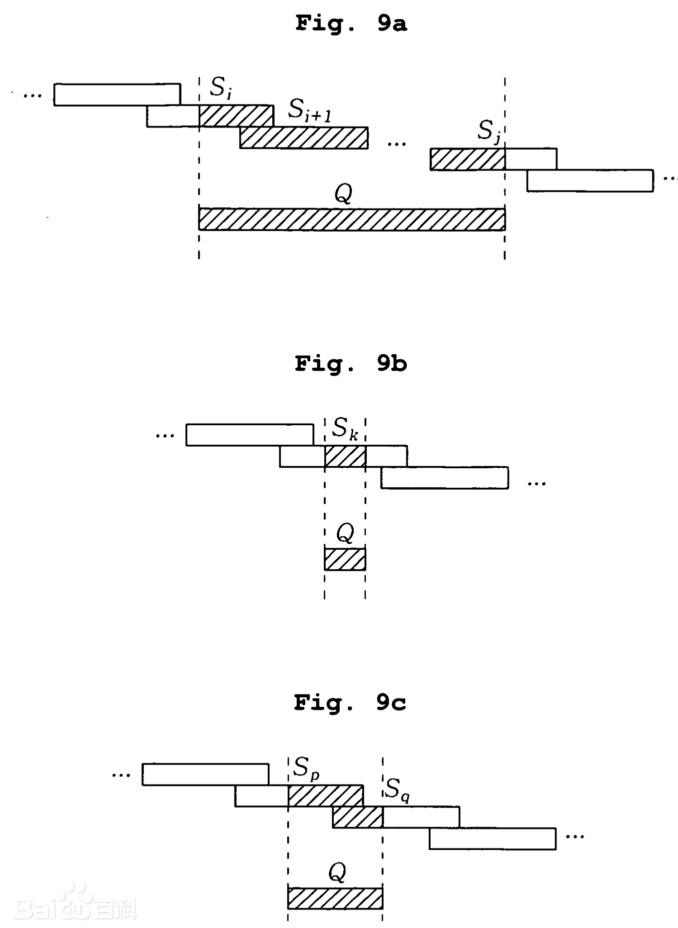

Figure 2: N-gram model

I	3437
want	1215
to	3256
eat	938
Chinese	213
food	1506
lunch	459

Table 1: **words and frequency**

	I	want	to	eat	Chinese	food	lunch
I	8	1087	0	13	0	0	0
want	3	0	786	0	6	8	6
to	3	0	10	860	3	0	12
eat	0	0	2	0	19	2	52
Chinese	2	0	0	0	0	120	1
food	19	0	17	0	0	0	0
lunch	4	0	0	0	0	1	0

Table 2: **Word sequence frequency**

References

[1] Yann LeCun *etal.* Deep learning. *Nature*, 521(28):9, 2015.