## 李宏毅2023春机器学习课程

Lecture1

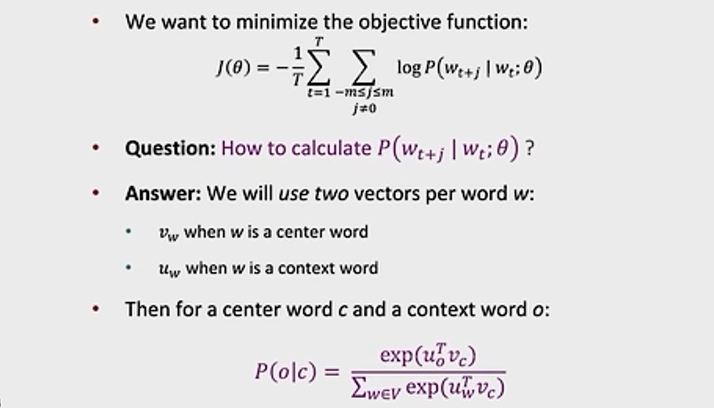
①传统的NLP任务中，我们把词看作是离散的表达符号，采用One-hot编码来对不同的词汇进行编码。但这样的表达使得单词之间没有关系，原本意思相近的词汇在表达之后并不相邻，并且会需要大量的维度来储存单词信息。

现代机器学习方法中，我们通过在实值向量中编码相似性来自然地表达词汇之间的语义相似性。分布语义学（Distributional Semantics）是自然语言处理领域中的一个分支，主要通过词语在大规模文本语料库中的分布特征来推断词语的含义。出现在相似语境中的词语往往具有相似的语义含义。通过分析词语在文本中的上下文信息，分布语义学可以生成词语的向量表示，用于计算词语之间的相似性和语义关系。

②词嵌入（Word Embedding）：是自然语言处理（NLP）中一种将词汇表中的单词或短语映射到高维空间中的向量表示的技术。这种技术使得计算机能够理解词与词之间的相似性和关系，进而能够处理和理解人类语言中的复杂性和多样性。

③Word2vec：一种词嵌入的技术。使用中心词来预测其上下文（即周围的词）。对于给定的中心词，模型会尝试最大化上下文单词出现在其周围窗口内的概率。

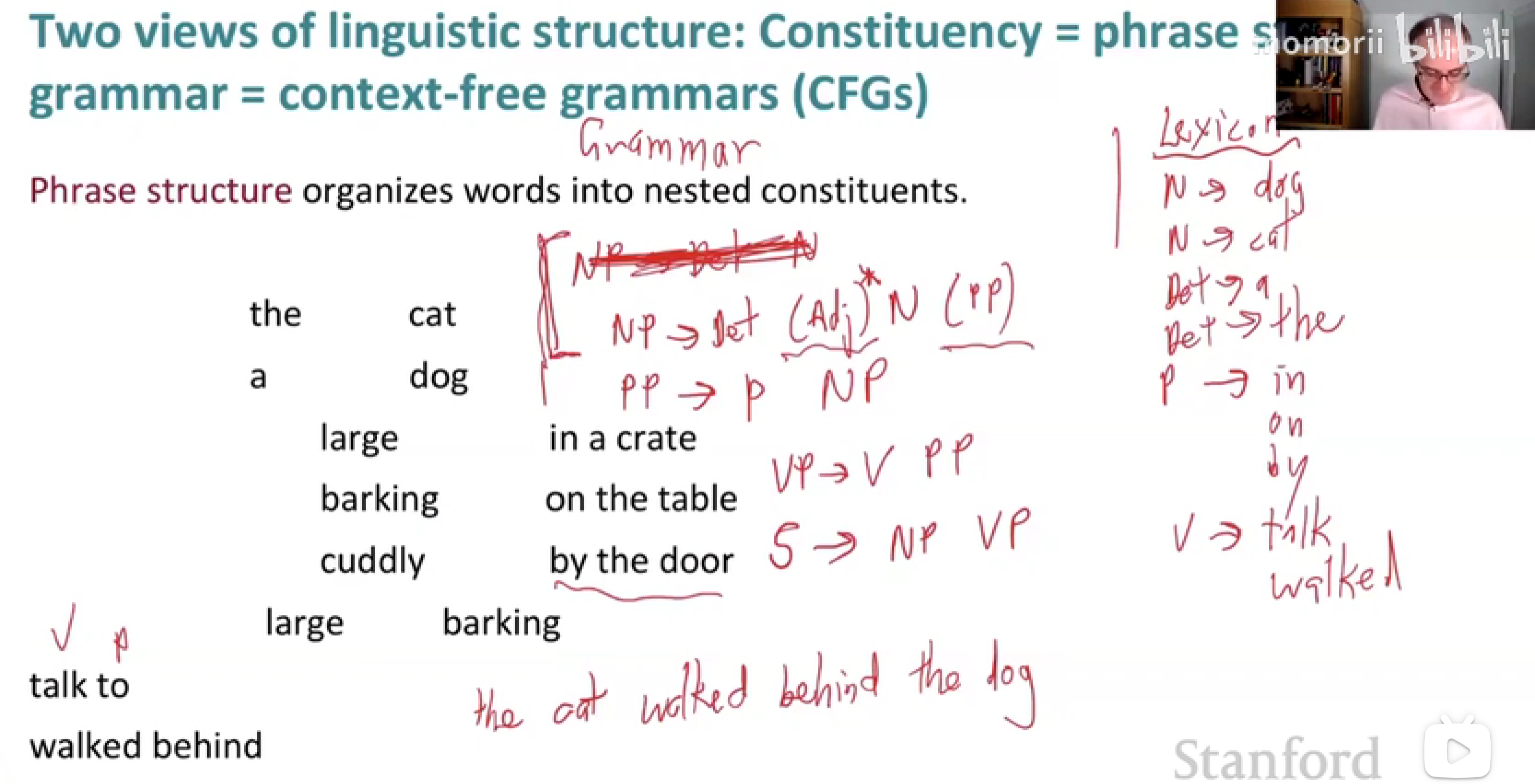
使用负对数似然损失函数来衡量模型预测与实际上下文词的差距。假设我们有一个训练样本，其中一个中心词和一个上下文词。模型的目标是最大化给定中心词下预测上下文词的概率。这个概率值用中心词和前后词的点积来表示，同时为了使得概率和为1，引入了softmax函数。通常使用随机梯度下降（SGD）或其变体来优化损失函数。SGD每次更新都会基于单个训练样本的梯度



在这样的词向量表示空间中，词向量之间的代数运算是有意义的。如king单词对应的词向量减去man加上women就能得到queen的大致词向量。

Lecture4

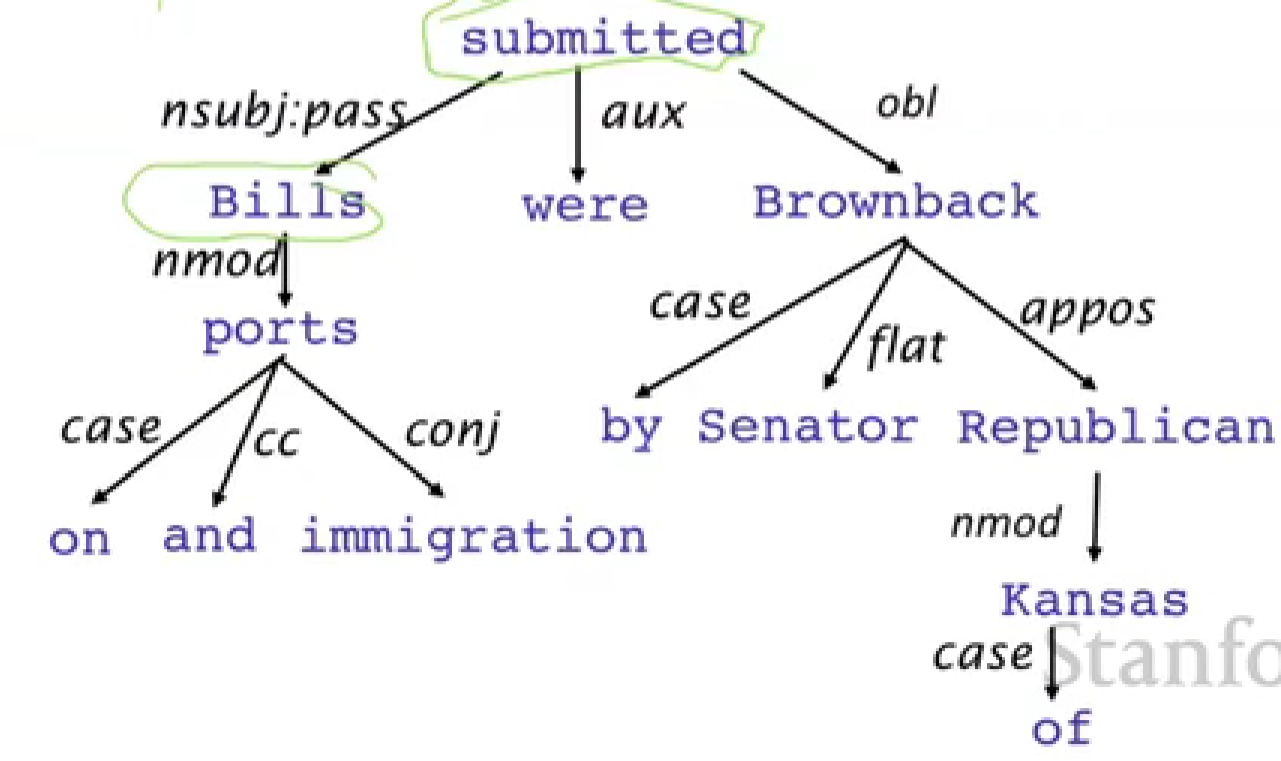
①语言是有语法的。对于词汇，我们已经形成了对应的词向量，但单词本身需要进行组合才能形成完成的表达。在机器学习中，常用CFGs（上下文无关文法）描述语言。CFGs由一组产生式规则组成，每个规则定义了如何从非终结符推导出字符串。下图中，笔记左侧列表示了从句子中提取的语法，右侧展示了对应的词性我们收集到的词汇。



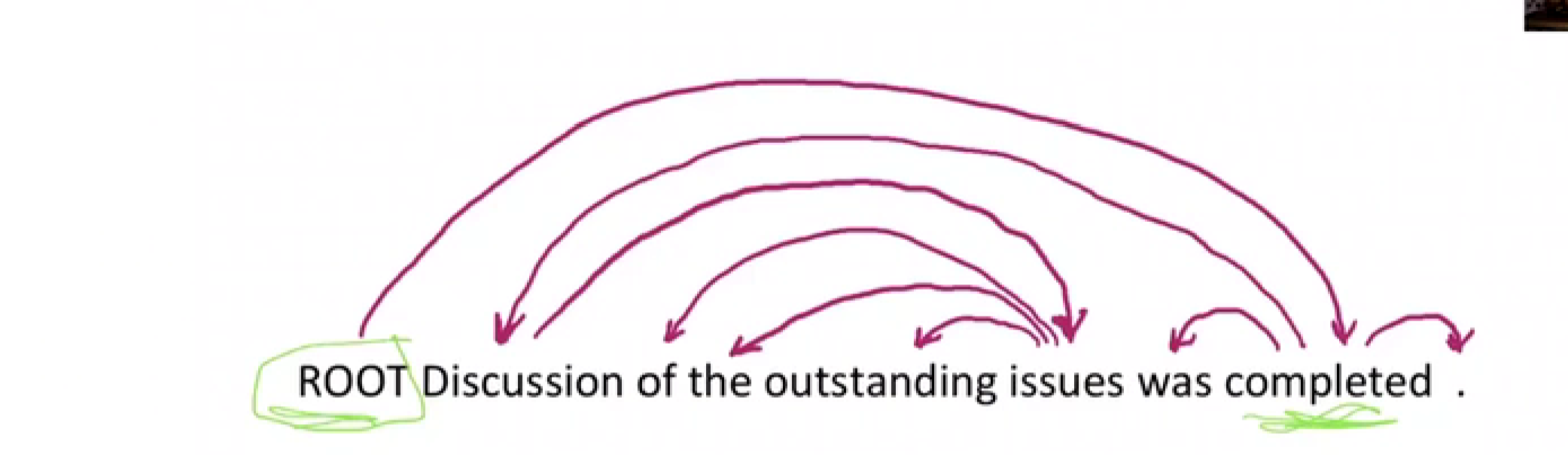
②依赖语法：是一种用于描述语言结构的语法理论，它通过分析句子中各个词之间的依赖关系来揭示句子的内部结构。与短语结构语法（Phrase Structure Grammar）不同，依赖语法不将句子视为由短语组成的层次结构，而是将句子视为一个由词和它们之间的依赖关系组成的网络。

在依赖语法中，每个词都被视为一个节点，节点之间的有向边表示词之间的依赖关系。通常，句子中有一个词被指定为根节点（通常是动词），其他词则通过依赖关系与根节点相连。依赖关系可以表示多种类型的语义和句法关系，如主语-谓语、宾语-动词、定语-名词等。

下图是描述一个句子依赖关系的树：



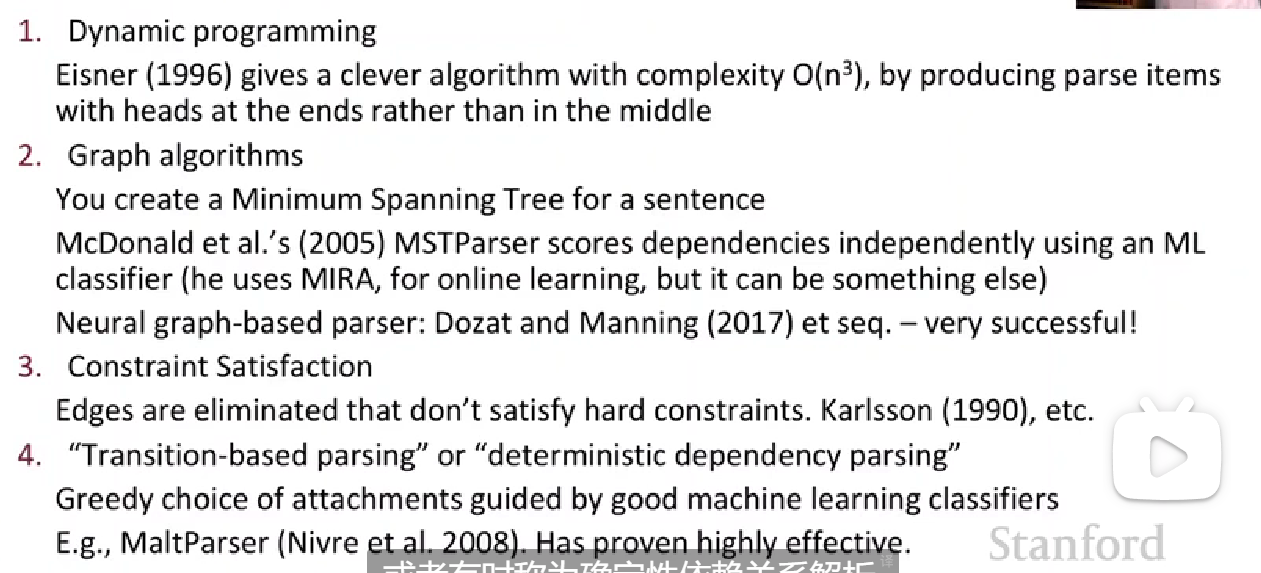
引入假词根：指一个词或短语被用作句子结构的核心，但它本身并不在语义上承担具体的意义。假词根在语法分析中被用来填补语法框架的需要，使得句子在结构上合乎语法规则，但并不一定传达具体的语义信息。



③Treebank：

Treebank（树库）是一种语言学和计算语言学研究中常用的语料库形式。它包含了经过语法分析和标注的句子集合，通常以语法树的形式展示句子的结构和成分关系。

④Dependency Parsing：如何构建依赖关系解释器：

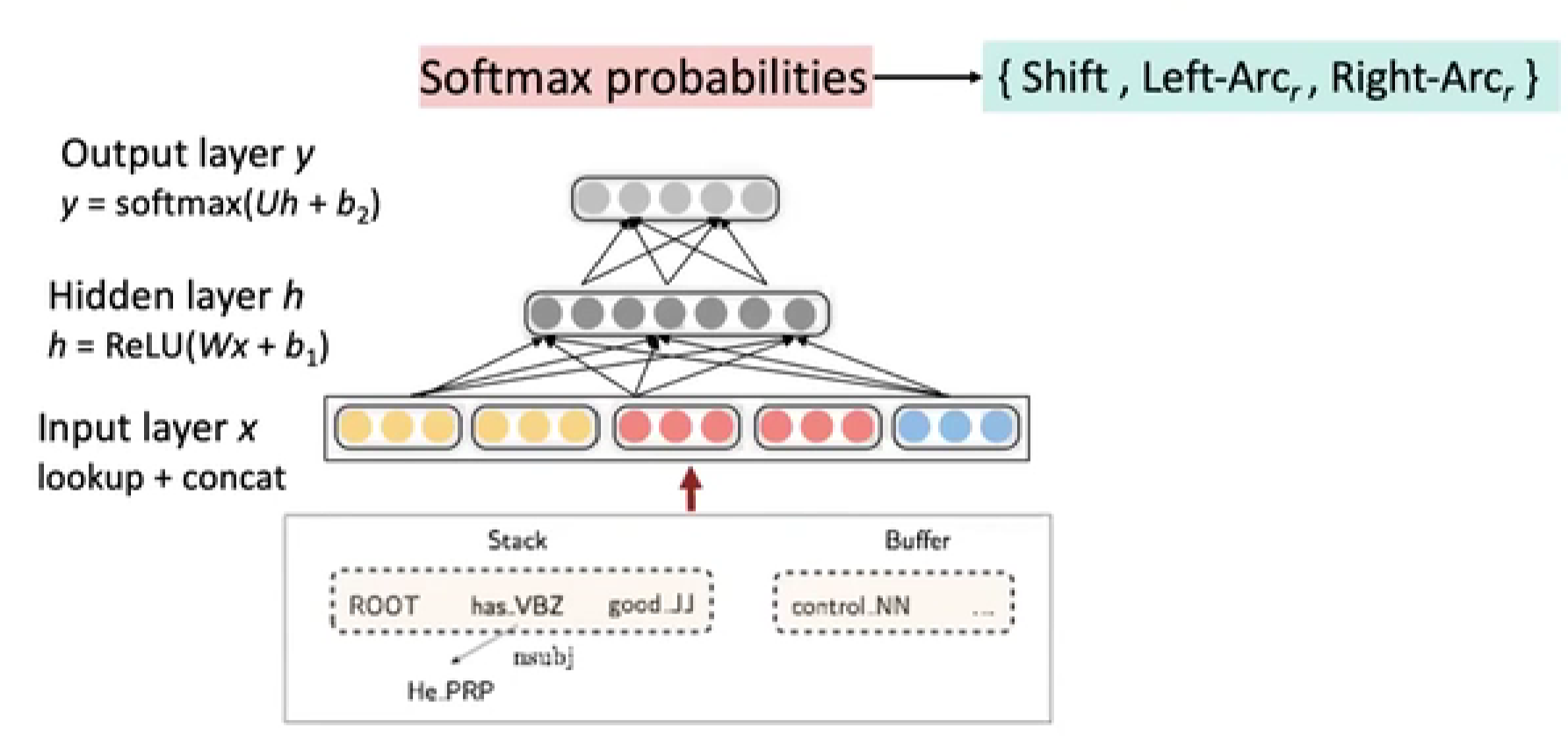


其中4为基于转换的句法分析方式。通过一系列的状态转移操作来逐步构建句子的依存句法树。在这个过程中会进行如下三种操作：

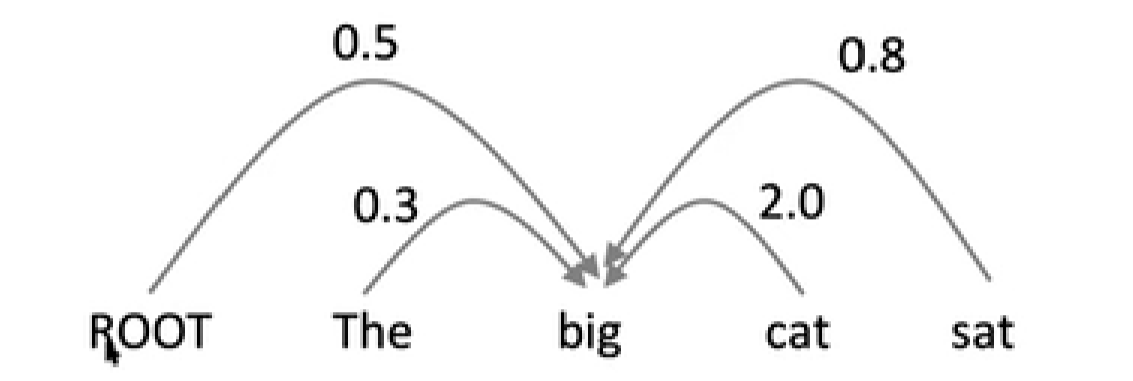
1. **SHIFT**：将队列中的第一个词移动到栈顶。
2. **LEFT-ARC**：在栈顶的两个词之间建立一条从左到右的边，并指定依赖关系的类型。然后，将栈顶的第二个词从栈中弹出。
3. **RIGHT-ARC**：在栈顶的两个词之间建立一条从右到左的边，并指定依赖关系的类型。然后，将栈顶的第一个词从栈中弹出。

在以前，我们并不知道应该做哪一种操作，复杂度是指数级的。但当我们引入了机器学习以后，我们可以训练一个预测器，用于预测下一步应该进行的操作是怎样的，这样当预测器有较高的正确率时，我们的算法复杂度就变为了线性的。

①Neural Dependency Parser Model：通过神经网络预测构建语法依赖关系中基于转换的语法分析方式的下一步动作是什么。一个简单的可用于此任务的神经网络架构如下所示：



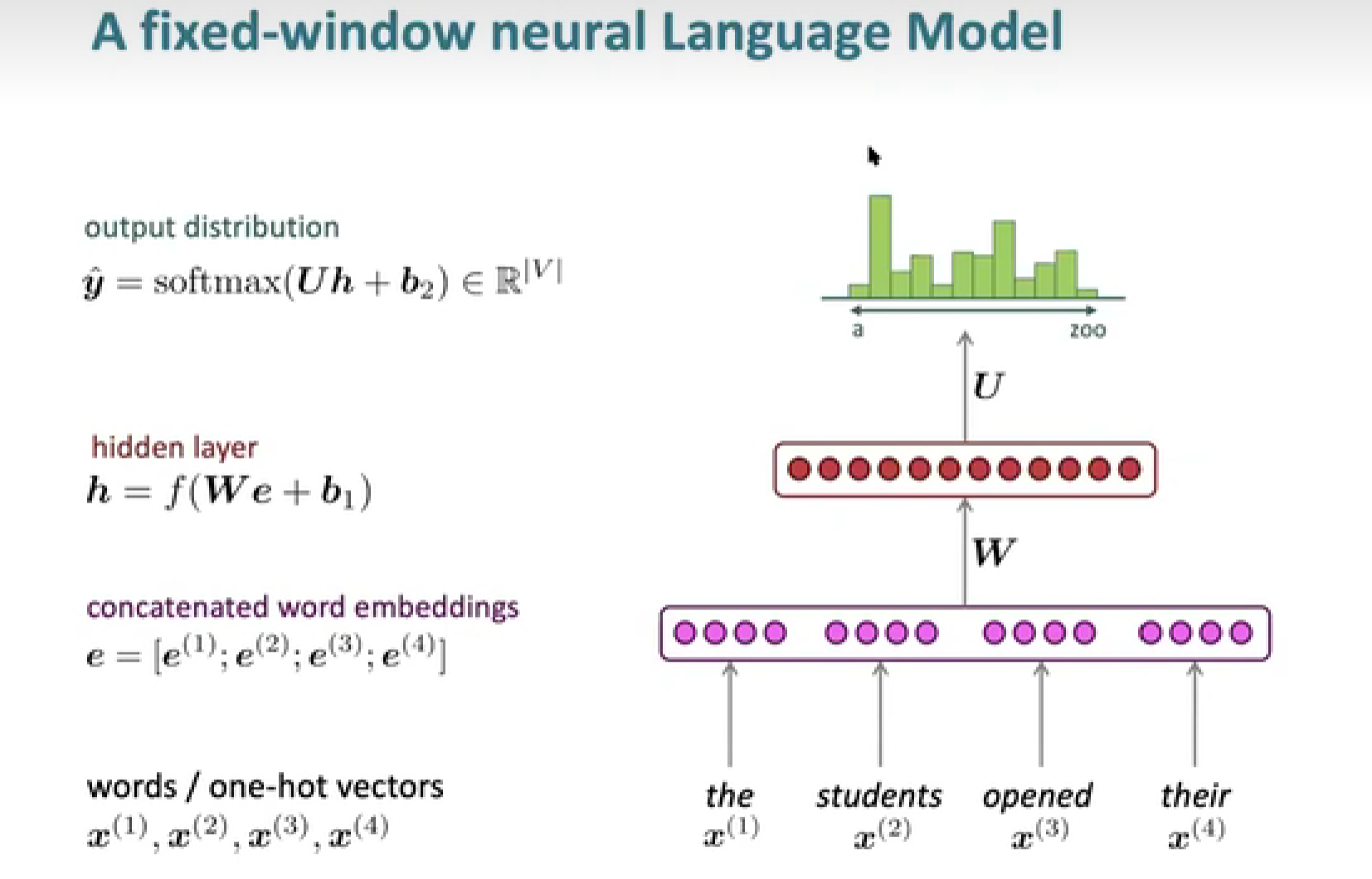
②Graph-based dependency parsers:



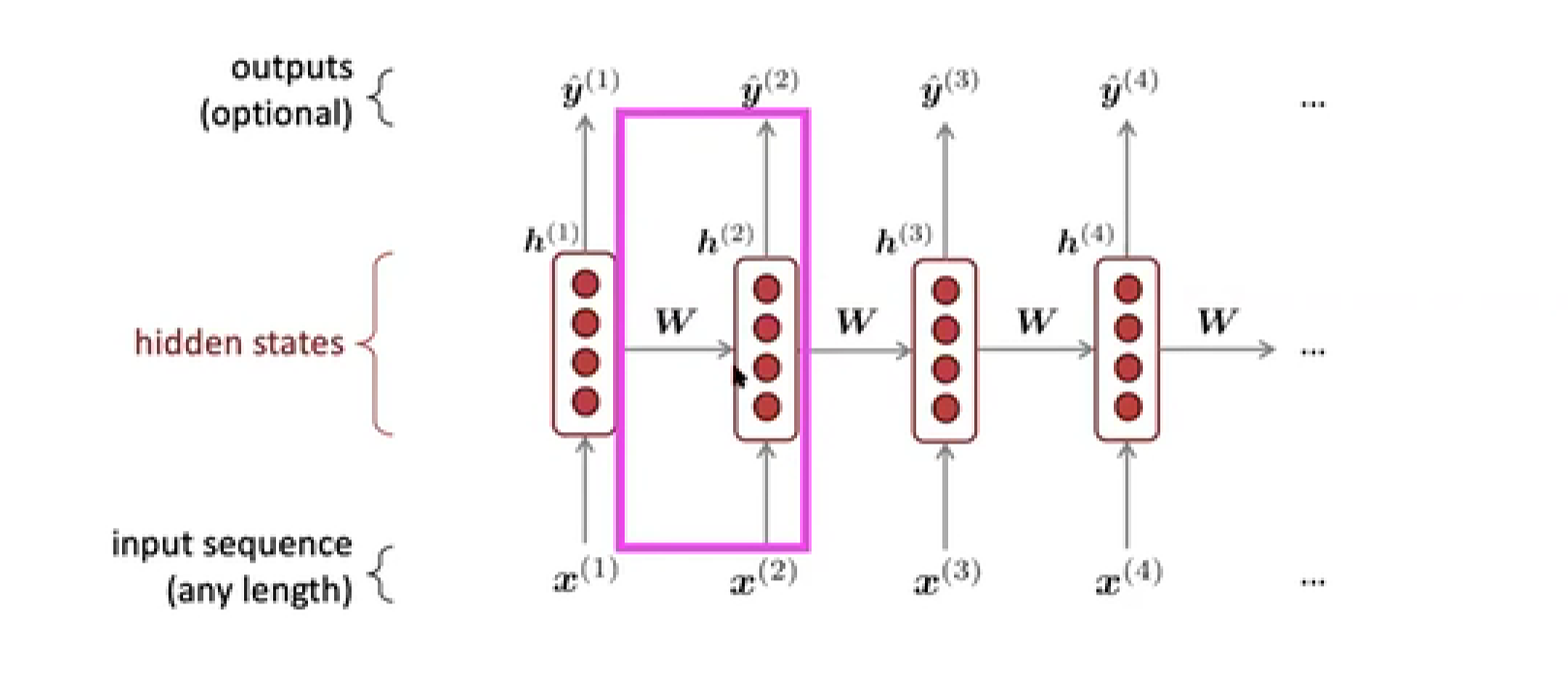
对每个可能的依赖都计算一个分数。

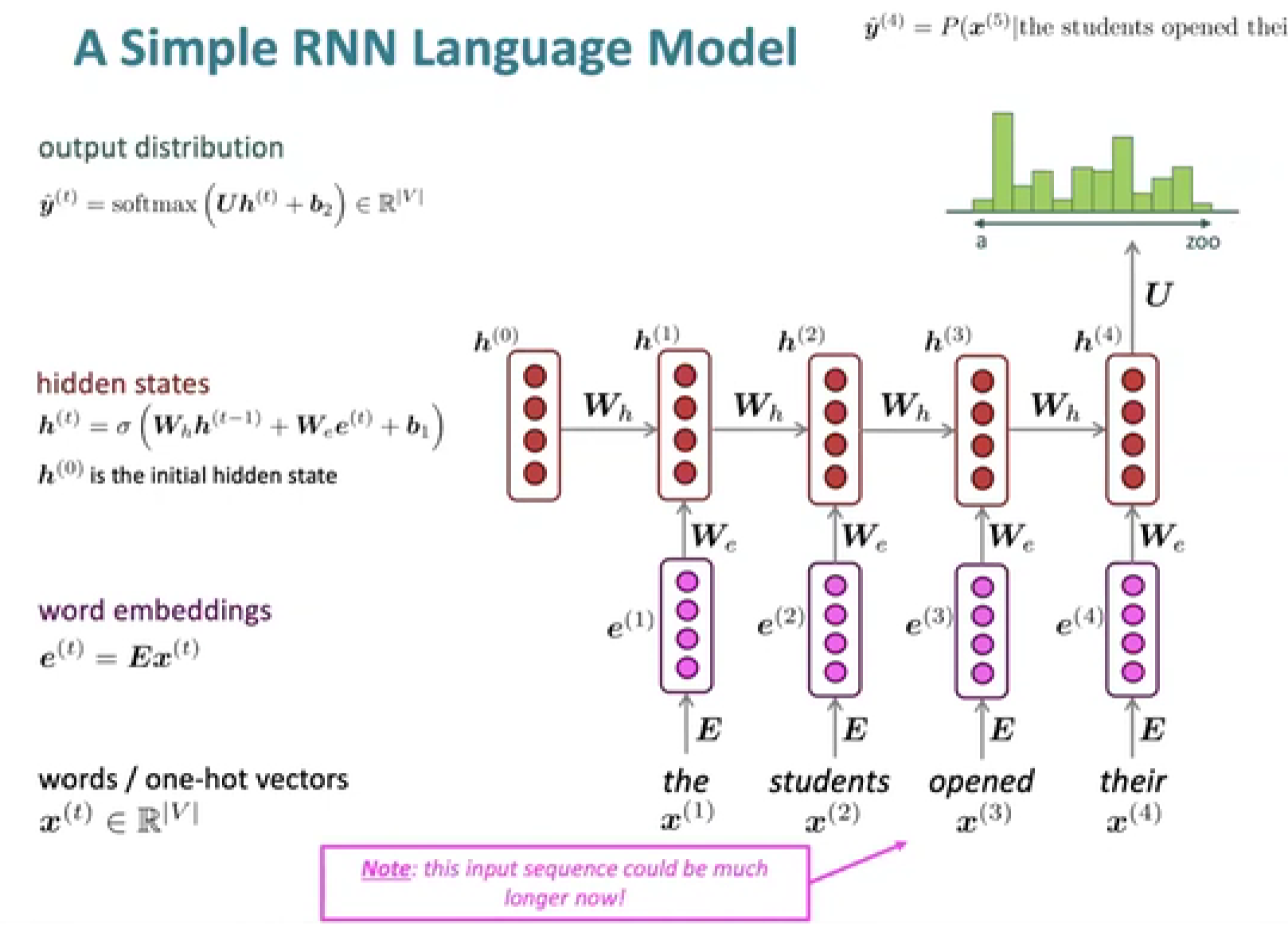
③Language model：语言模型，指的是预测下一个单词的任务。即做“词语接龙”。是根据给定上下文来预测下一个词或一句话的概率的模型。n-gram language model（n元语言模型）是一种简单而常用的统计语言模型。n-gram模型基于一个假设，即一个词的出现只依赖于前面的n-1个词。因此，n-gram模型将文本中的词序列表示为连续的n个词的序列。通过统计给定文本语料库中每个n-gram序列的出现频率，并计算条件概率，这些概率用于估计下一个词在给定前面的n-1个词的情况下出现的概率。基于马尔可夫假设提出的这个模型。

相较于早期的n元语言模型，我们现在采用神经网络来预测下一个单词的概率。采用同样的思想，我们可以设置输入为单词前的数个词，舍弃前置的词汇。示意图如下所示：

这个模型的问题仍然在与我们舍弃了很多词语，导致前置的语境丧失。

④递归神经网络(RNN):





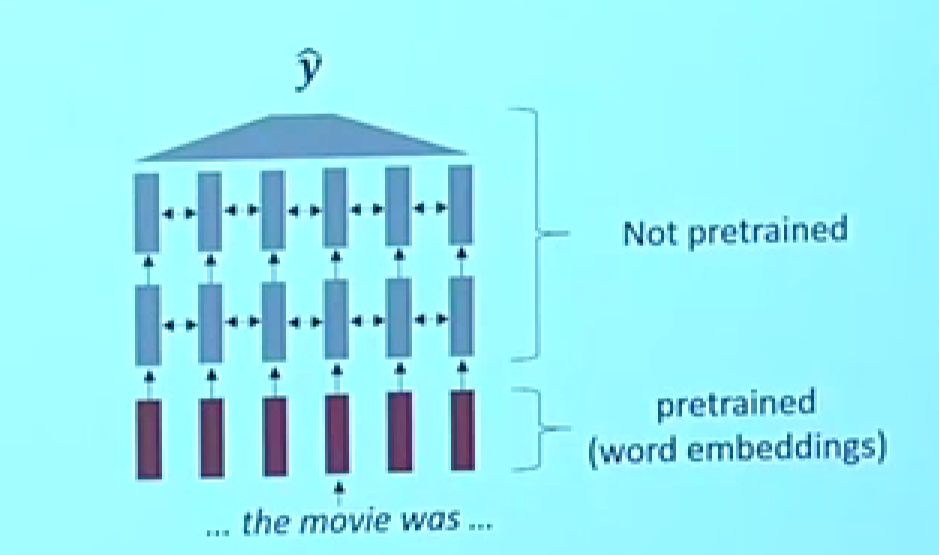
优点：可以接受任意长度的输入；参数共享，在不同的时间步上共享相同的参数；RNN可以灵活地处理不同长度的输入序列，而不需要对输入数据进行复杂的预处理。

缺点：由于需要保留每个时间步的隐藏状态，RNN的计算成本仍然相对较高；由于梯度消失/梯度爆炸的问题，很难学习到长距离的信息。

Lecture5

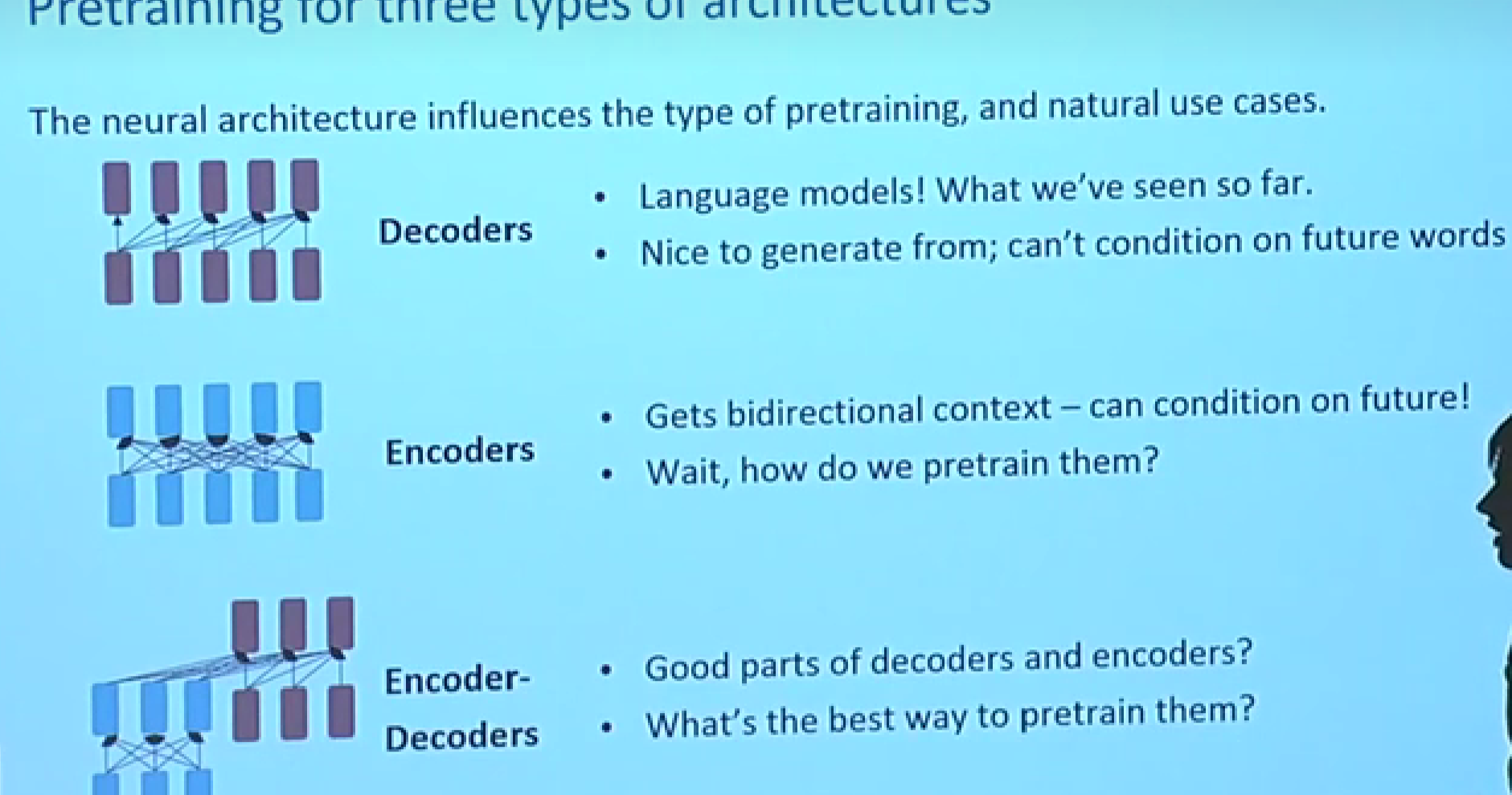
①子词模型：子词模型是自然语言处理和机器学习任务中使用的一种方法，用于将单词分解为更小单位，称为子词。这种方法有助于解决语言中复杂形态，帮助模型泛化到单词的不同形式上。子词模型通常使用字节对编码（Byte-Pair Encoding，BPE）或者词块（WordPiece）等算法生成。这些算法通过迭代地合并语料库中最频繁出现的字符或标记对，创建子词单元的词汇表。

②传统的Woerd2Vec方法将词转化为词嵌入，作为模型的输入，但这样并没有考虑某些词汇在不同语境下的不同含义，因此在训练时可以采用LSTM、Transformer等技术来联系上下文的语境。相当于我们有一些预训练的参数，同时也有一些随机初始化的未训练参数。



隐藏了部分模型的输入，并使模型对这部分被隐藏的输入进行重构。

③预训练模型的三种结构：



其中encoder使用掩码语言模型作为预训练任务。在预训练阶段，模型会随机掩盖输入文本中的一部分单词，并尝试根据上下文预测这些被掩盖的单词。常见的模型代表为bert。