

主题模型

Machine Learning Engineer

机器学习工程师

讲师：加号

目录

CONTENTS

01

主题模型理论

02

主题模型算法

03

实战案例



01

主题模型理论

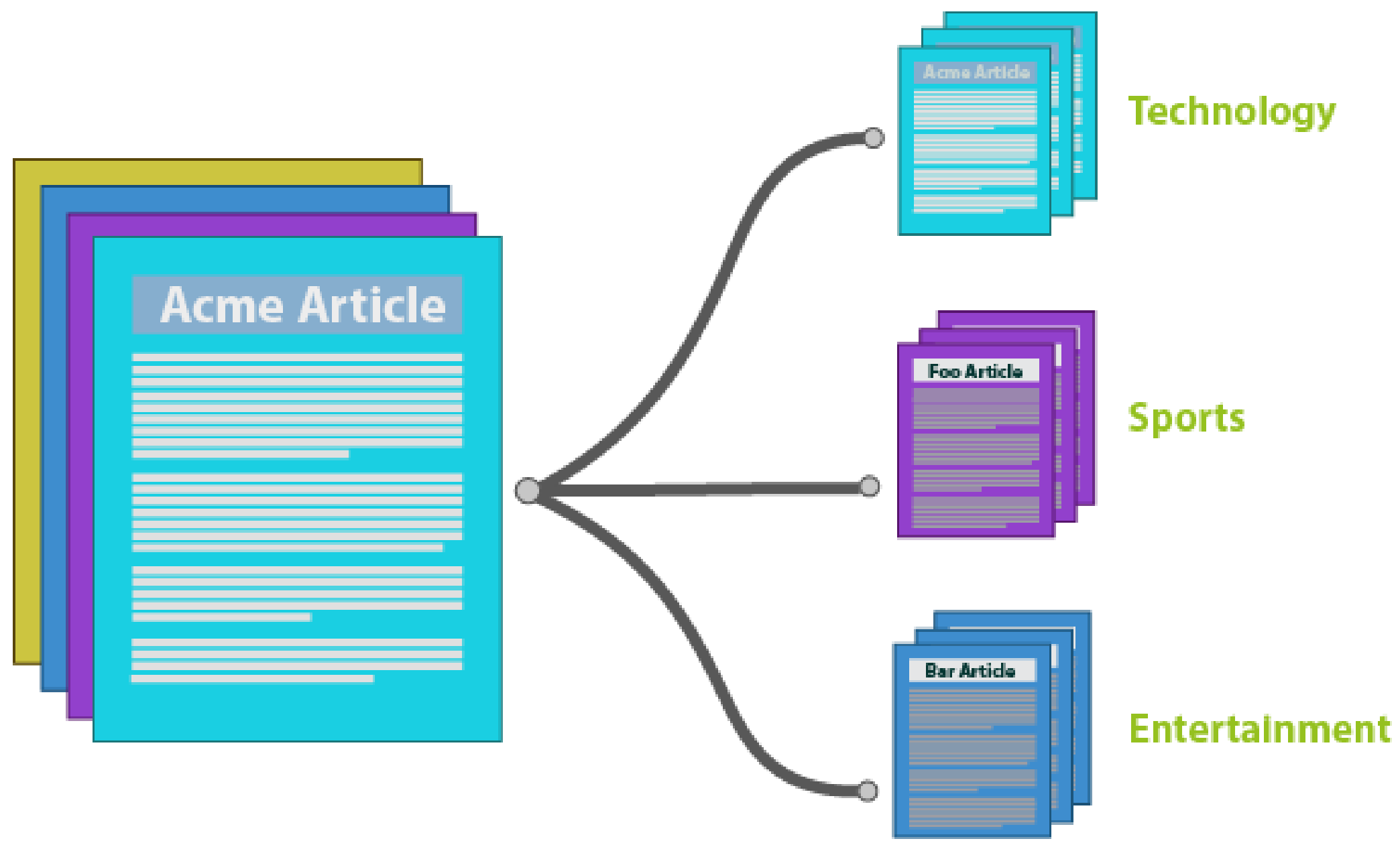
1.1

直观版

1.2

标准版

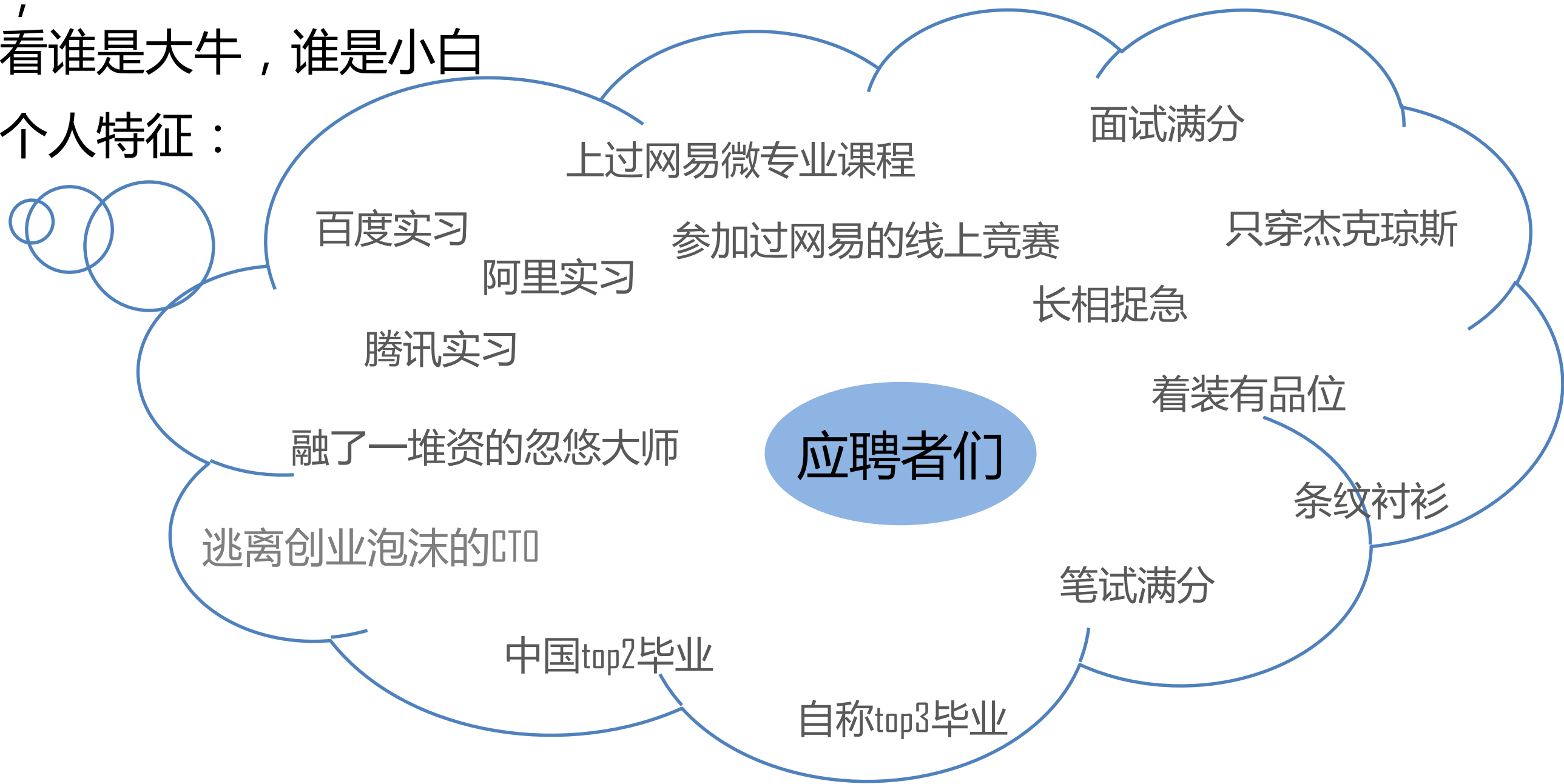
什么是主题模型？



什么是主题模型？

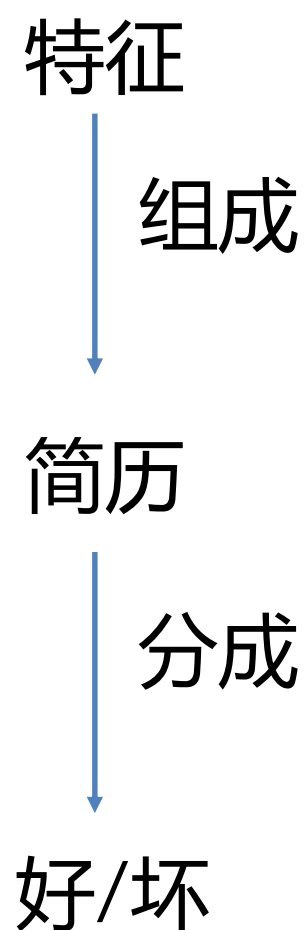
1. 直观版：

假设某企业想要招聘一个工程师，
他们收到了一把的简历，
他们想直接通过简历来看谁是大牛，谁是小白
简历里通常会包含这些个人特征：



1. 直观版：

以下三个要素，构成了这家企业的人力总监判断的基础：



这位人力总监要做的事是：

- 拿出一份份简历
 - 记录下每份简历包含的特征
- 然而，他并不知道，这一切代表着什么

于是他开始猜

拿起一份简历A，

- 他看到里面说A参加过网易机器学习工程师的微专业
- 他就猜这位童鞋的水平应该很高，八成是个好工程师
- 但他又看到A学历只是小学毕业，心里又有了两成的担忧

他又看到B

又看到C

等等。。。

1. 直观版：

当然，这个猜，只是猜，没有任何证据可以证实。

但是这位人力总监是久经职场的老司机，他通过经验统计来调整自己的猜想：

- 选一份『张三』的简历，选一个特征『条纹衬衫』
- 为什么『张三』有可能喜欢穿『条纹衬衫』？也许是因为穿条纹衬衫是优秀程序员的信仰。
- 也就是说，越多的优秀程序员穿『条纹衬衫』，越让人力总监猜想『张三』的其他个人特征也符合优秀程序员的喜好，并且『张三』本人穿『条纹衬衫』是一个优秀的程序员自我修养的体现。
- 继续猜，继续拿『张三』和『条纹衬衫』两个元素。人力总监转念一想，也有可能爱穿条纹的都是小白。
- 于是他按照上面的逻辑，再看看『张三』穿『条纹衬衫』是『小白』的可能性有多少。
- 把所有的简历和所有的特征都做个组合，都来猜一下是小白还是大牛。

1. 直观版：

久经沙场之后，老司机人力总监掌握了如下信息：

对于是不是优秀程序员的分类，它通过人头统计大概有了数。

这让他以后看到新简历的时候，一眼就知道他是不是个优秀程序员。

对于每个特征 C ，他也能说出大概百分之多少的人拥有特征 C 可以说明他们是优秀的程序员。

总结成公式就是：

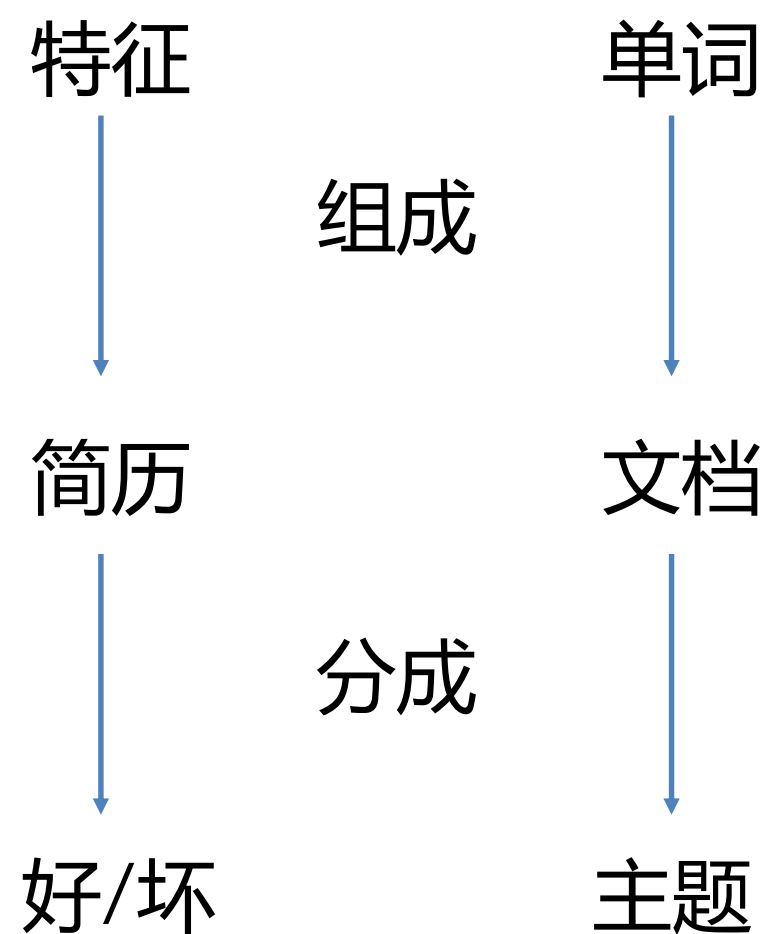
$$P(\text{优秀程序员} | \text{特征}, \text{简历}) = \frac{\text{此特征在优秀程序员之中出现的次数}}{\text{优秀程序员拥有的所有特征}} \times \text{此简历中属于优秀程序员特征的个数}$$

什么是主题模型？

1. 直观版：

例子与理论的关系：

以上，就是我们用现实的例子模拟的LDA模型来区分简历好坏。
在文本的主题分类中，我们的例子和实际之间的联系如下：



什么是LDA？

Latent Dirichlet Allocation:

是一种无监督的贝叶斯模型

是一种主题模型，它可以将文档集中每篇文档的主题按照概率分布的形式给出。同时它是一种无监督学习算法，在训练时不需要手工标注的训练集，需要的仅仅是文档集以及指定主题的数量 k 即可。此外LDA的另一个优点则是，对于每一个主题均可找出一些词语来描述它。

是一种典型的词袋模型，即它认为一篇文档是由一组词构成的一个集合，词与词之间没有顺序以及先后的关系。一篇文档可以包含多个主题，文档中每一个词都由其中的一个主题生成。

— wikipedia

什么是主题模型？

什么是贝叶斯模型？

理论

$$P(\theta, y) = P(\theta)P(y | \theta)$$

$$P(\theta | y) = \frac{P(\theta, y)}{P(y)} = \frac{P(y | \theta)P(\theta)}{P(y)}$$

模型

- 用概率作为『可信度』
- 每次看到新数据，就更新『可信度』
- 需要一个模型来解释数据的生成

什么是主题模型？

先验，后验与似然

$$P(\text{好工程师} | \text{简历}) = P(\text{好工程师}) P(\text{简历} | \text{好工程师}) / P(\text{简历})$$

后验

先验

似然

应聘者概率模型

简历数据生成模型

2. 标准版

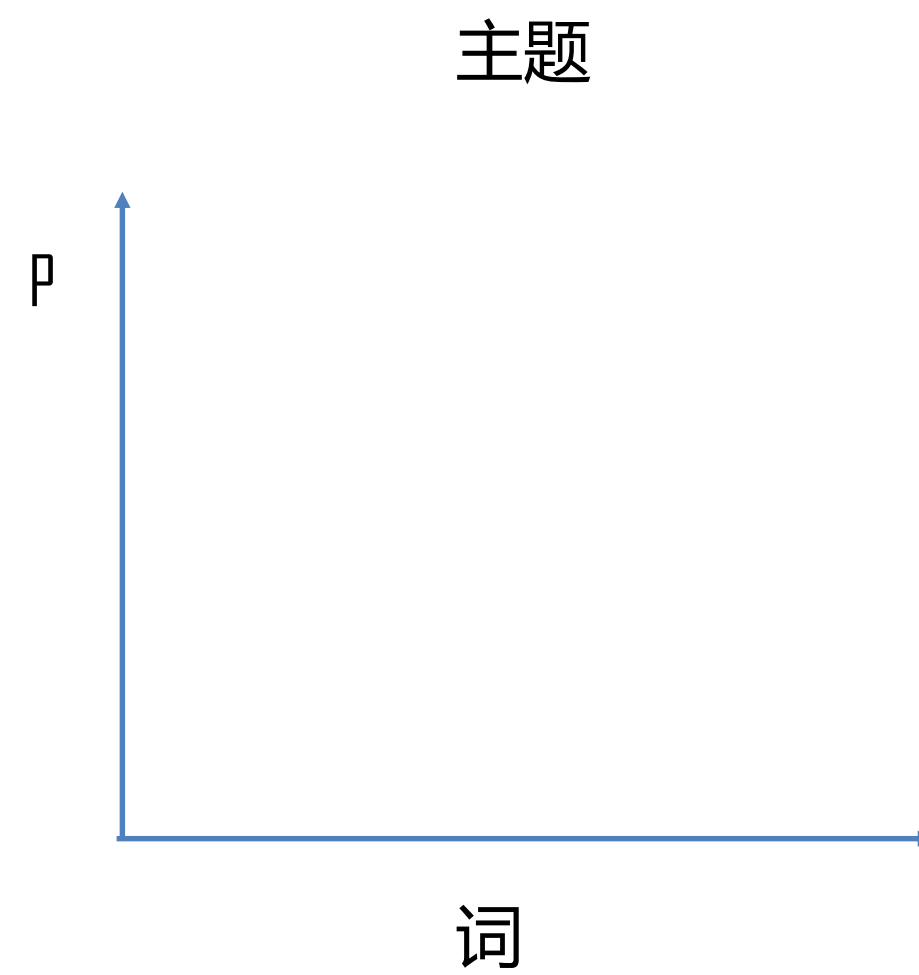
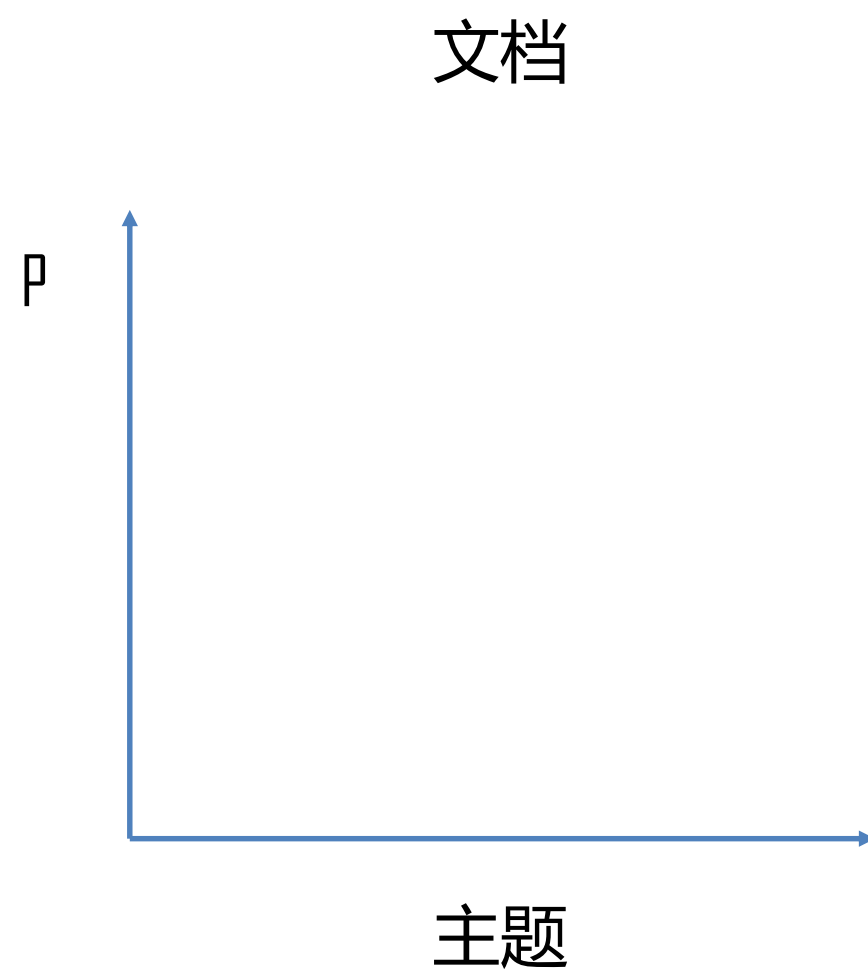
我们用LDA找寻的就是之前例子中总监大人统计出来的经验：

一份简历的每个特征，都是因为本人有一定概率是好/坏程序员，并从好/坏这个分类中以一定概率选择某些特征而组成的。

一篇文章的每个词，都是以一定概率选择了某个主题，并从这个主题中以一定概率选择某个词语而组成的。

$$P(\text{单词} | \text{文档}) = P(\text{单词} | \text{主题}) * P(\text{主题} | \text{文档})$$

2. 标准版



2. 标准版

LDA生成过程：

对于语料库中的每篇文档，LDA定义了如下生成过程（generative process）：

- 1.对每一篇文档，从主题分布中抽取一个主题；
- 2.从上述被抽到的主题所对应的单词分布中抽取一个单词；
- 3.重复上述过程直至生成一个『符合这个主题的文档』。

2. 标准版

LDA生成过程（具体）：

稍微具体点儿讲：

（ w 代表单词； d 代表文档； t 代表主题；大写代表总集合，小写代表个体。）

D 中每个文档 d 看作一个单词序列 $\langle w_1, w_2, \dots, w_n \rangle$ ， w_i 表示第 i 个单词。

D 中涉及的所有不同单词组成一个词汇表大集合 V （vocabulary），LDA以文档集合 D 作为输入，希望训练出的两个结果向量（假设形成 k 个topic， V 中一共 m 个词）：

- 对每个 D 中的文档 d ，对应到不同Topic的概率 $\theta_d \langle \theta_{d1}, \dots, \theta_{dk} \rangle$ ，其中， θ_{di} 表示 d 对应 T 中第 i 个topic的概率。计算方法是直观的， $\theta_{di} = n_{di} / n$ ，其中 n_{di} 表示 d 中对应第 i 个topic的词数目， n 是 d 中所有词的总数。
- 对每个 T 中的 $\text{topic } t$ ，生成不同单词的概率 $\phi_t \langle \phi_{t1}, \dots, \phi_{tm} \rangle$ ，其中， ϕ_{ti} 表示 t 生成 V 中第 i 个单词的概率。计算方法同样很直观， $\phi_{ti} = N_{di} / N$ ，其中 N_{di} 表示对应到 $\text{topic } t$ 的 V 中第 i 个单词的数目， N 表示所有对应到 $\text{topic } t$ 的单词总数。

2. 标准版

LDA生成过程：

所以，LDA的核心公式如下：

$$P(w|d)=P(w|t)*P(t|d)$$

直观的看这个公式，就是以Topic作为中间层，可以通过当前的 θ_d 和 ϕ_t 给出了文档 d 中出现单词 w 的概率。其中 $p(t|d)$ 利用 θ_d 计算得到， $p(w|t)$ 利用 ϕ_t 计算得到。

实际上，利用当前的 θ_d 和 ϕ_t ，我们可以为一个文档中的一个单词计算它对应任意一个Topic时的 $p(w|d)$ ，然后根据这些结果来更新这个词应该对应的topic。然后，如果这个更新改变了这个单词所对应的Topic，就会反过来影响 θ_d 和 ϕ_t 。

2. 标准版

LDA学习过程：

LDA算法开始时，先随机地给 θ_d 和 ϕ_t 赋值（对所有的 d 和 t ）。然后：

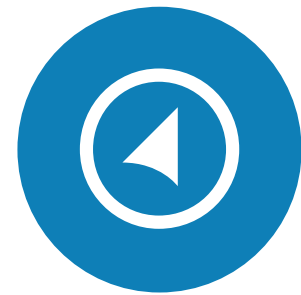
1. 针对一个特定的文档 d_s 中的第 i 单词 w_i ，如果令该单词对应的topic为 t_j ，可以把上述公式改写为：

$$P_j(w_i | d_s) = P(w_i | t_j) * P(t_j | d_s)$$

2. 现在我们可以枚举 T 中的topic，得到所有的 $p_j(w_i | d_s)$ 。然后可以根据这些概率值结果为 d_s 中的第 i 个单词 w_i 选择一个topic。最简单的想法是取令 $p_j(w_i | d_s)$ 最大的 t_j （注意，这个式子里只有 j 是变量）

3. 然后，如果 d_s 中的第 i 个单词 w_i 在这里选择了一个与原先不同的topic（也就是说，这个时候 i 在遍历 d_s 中所有的单词，而 t_j 理当不变），就会对 θ_d 和 ϕ_t 有影响了。它们的影响又会反过来影响对上面提到的 $p(w | d)$ 的计算。对 D 中所有的 d 中的所有 w 进行一次 $p(w | d)$ 的计算并重新选择topic看作一次迭代。这样进行 n 次循环迭代之后，就会收敛到LDA所需要的结果了。

要点总结



要点1

主题模型的直观理解



要点2

主题模型的算法理解



02

主题模型算法

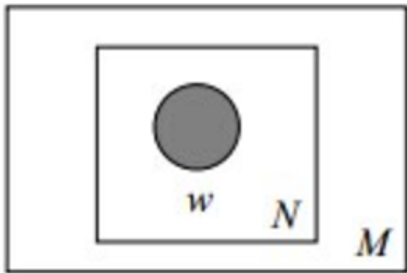
- 2.1 Unigram Model
- 2.2 Mixture of Unigram Model
- 2.3 pLSA
- 2.4 LDA

几个主题模型（循序渐进）：

Unigram model

对于文档 $\mathbf{w} = (w_1, w_2, \cdots, w_N)$ ，用 $p(w_n)$ 表示词 w_n 的先验概率，生成文档 \mathbf{w} 的概率为：

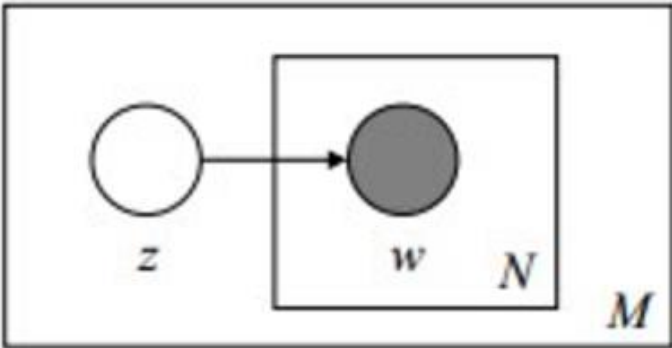
$$p(\mathbf{w}) = \prod_{n=1}^N p(w_n)$$



Mixture of unigrams model

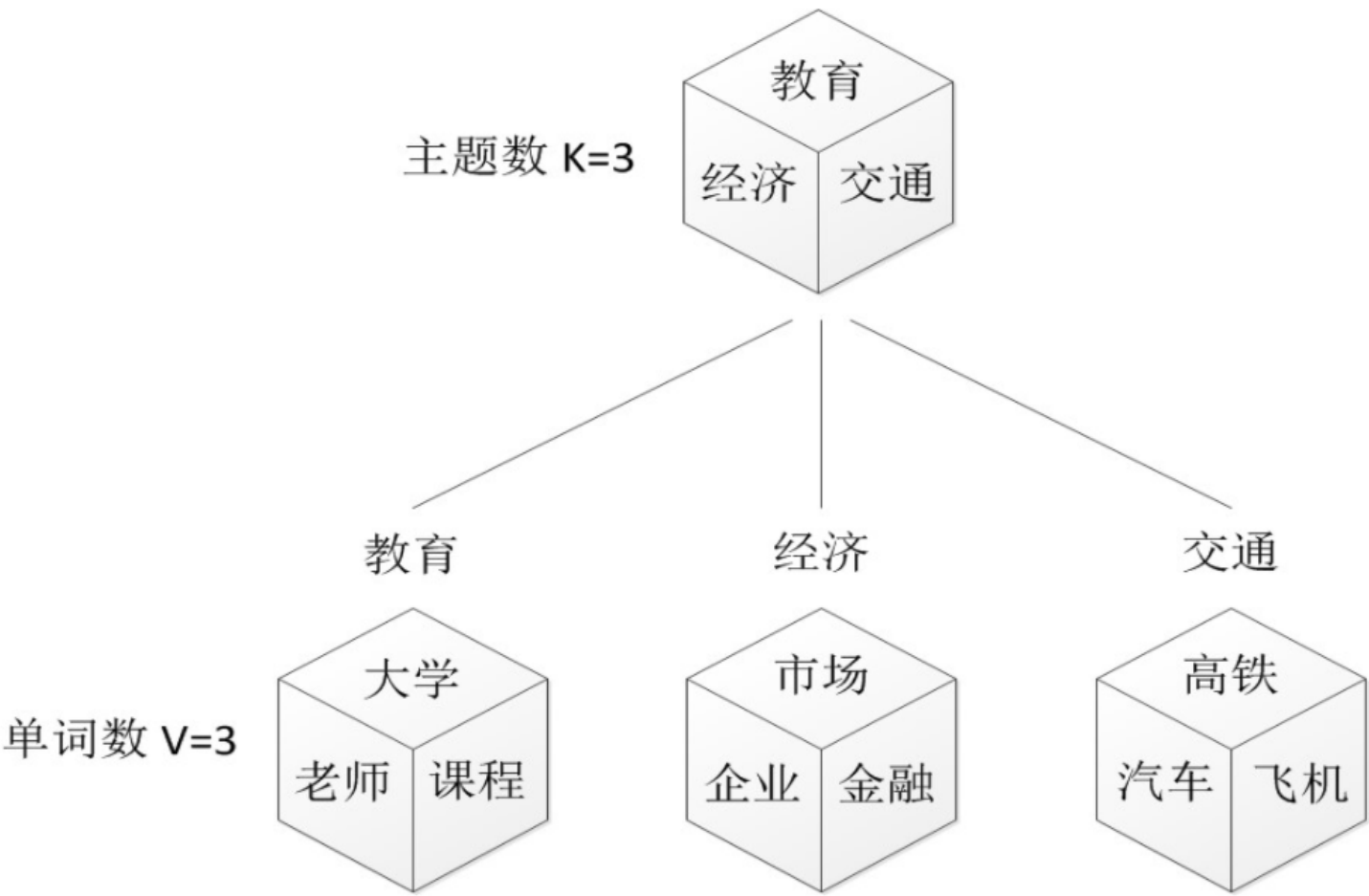
该模型的生成过程是：给某个文档先选择一个主题 z ，再根据该主题生成文档，该文档中的所有词都来自一个主题。假设主题有 $z_1, z_2, z_3, \dots, z_k$ ，生成文档的概率为：

$$p(\mathbf{w}) = p(z_1) \prod_{n=1}^N p(w_n|z_1) + \cdots + p(z_k) \prod_{n=1}^N p(w_n|z_k) = \sum_z p(z) \prod_{n=1}^N p(w_n|z)$$



PLSA模型

刚刚的mix unigram模型里面，一篇文章只给了一个主题。但是现实生活中，一篇文章可能有多个主题，只不过是『出现的几率』不一样。



我们定义：

- $P(d_i)$ 表示海量文档中某篇文档被选中的概率。
- $P(w_j|d_i)$ 表示词 w_j 在给定文档 d_i 中出现的概率。
 - 怎么计算得到呢？针对海量文档，对所有文档进行分词后，得到一个词汇列表，这样每篇文档就是一个词语的集合。对于每个词语，用它在文档中出现的次数除以文档中词语总的数目便是它在文档中出现的概率 $P(w_j|d_i)$ 。
- $P(z_k|d_i)$ 表示具体某个主题 z_k 在给定文档 d_i 下出现的概率。
- $P(w_j|z_k)$ 表示具体某个词 w_j 在给定主题 z_k 下出现的概率，与主题关系越密切的词，其条件概率 $P(w_j|z_k)$ 越大。

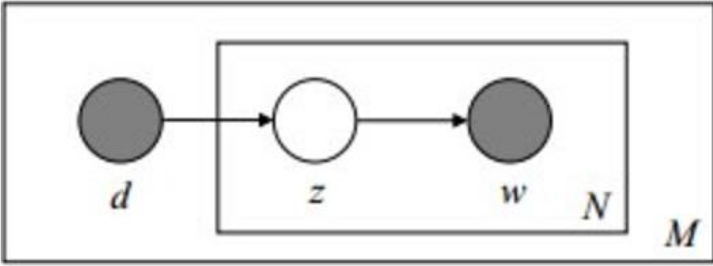
我们的文本生成模型就是：

（还记得什么是文本生成模型嘛？返回标准版part回顾一下）

1. 按照概率 $P(d_i)$ 选择一篇文档 d_i
2. 选定文档 d_i 后，从主题分布中按照概率 $P(z_k|d_i)$ 选择一个隐含的主题类别 z_k
3. 选定 z_k 后，从词分布中按照概率 $P(w_j|z_k)$ 选择一个词 w_j

我们通过观测，得到了『知道主题是什么，我就用什么单词』的文本生成模型，那么，根据贝叶斯定律，我们就可以反过来推出『看见用了什么单词，我就知道主题是什么』

从而可以根据大量已知的文档-词项信息 $P(w_j|d_i)$ ，训练出文档-主题 $P(z_k|d_i)$ 和主题-词项 $P(w_j|z_k)$ ，如下公式所示：



$$P(w_j|d_i) = \sum_{k=1}^K P(w_j|z_k)P(z_k|d_i)$$

故得到文档中每个词的生成概率为：

$$\begin{aligned} P(d_i, w_j) &= P(d_i)P(w_j|d_i) \\ &= P(d_i) \sum_{k=1}^K P(w_j|z_k)P(z_k|d_i) \end{aligned}$$

由于 $P(d_i)$ 可事先计算求出，而 $P(w_j|z_k)$ 和 $P(z_k|d_i)$ 未知，所以 $\theta = (P(w_j|z_k), P(z_k|d_i))$ 就是我们要估计的参数（值），通俗点说，就是要最大化这个 θ 。

LDA模型

LDA就是在pLSA的基础上加层贝叶斯框架，即LDA就是pLSA的贝叶斯版本。

PLSA与LDA对比

PLSA

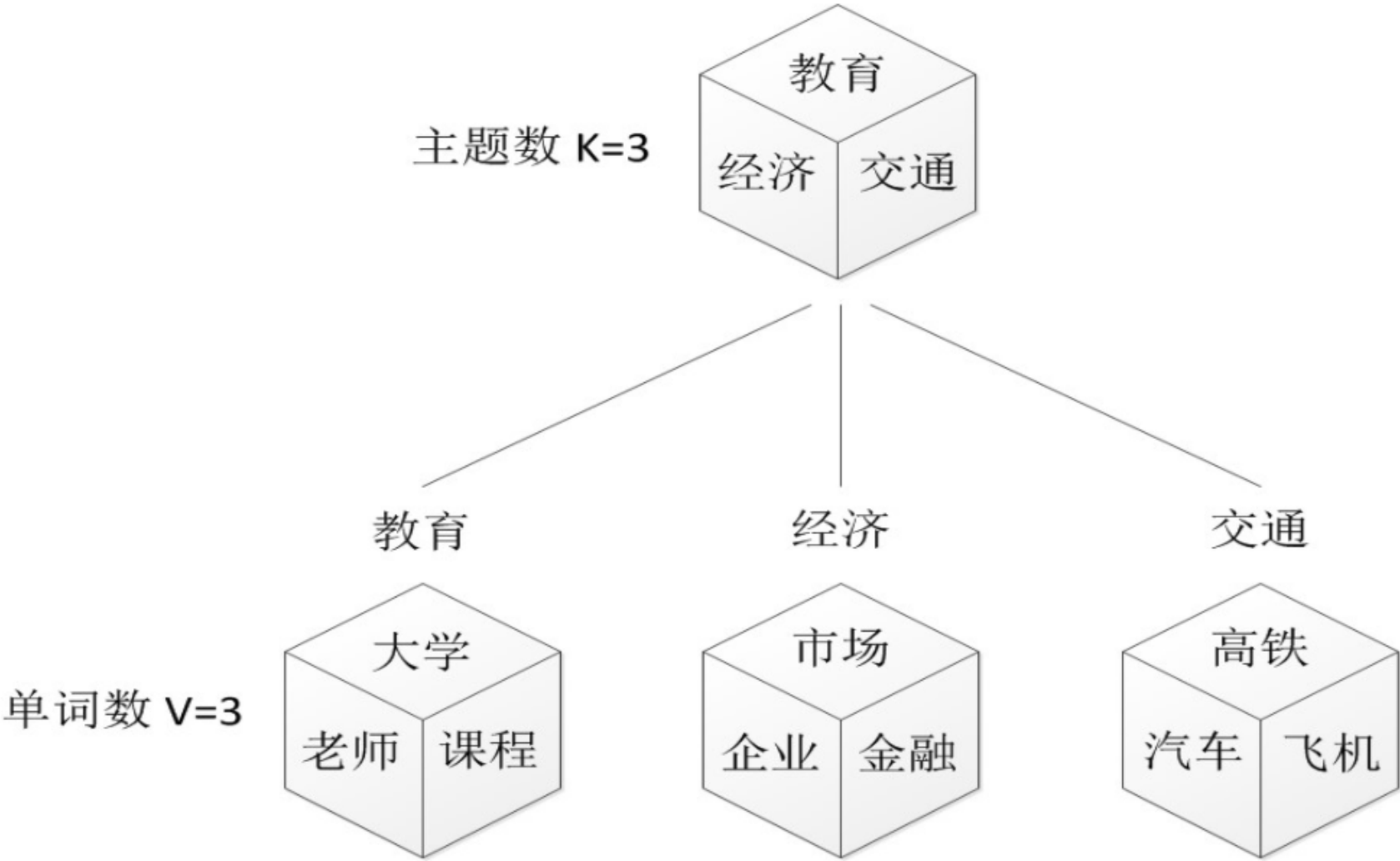
1. 按照概率 $P(d_i)$ 选择一篇文档 d_i
2. 选定文档 d_i 后，确定文章的主题分布
3. 从主题分布中按照概率 $P(z_k|d_i)$ 选择一个隐含的主题类别 z_k
4. 选定 z_k 后，确定主题下的词分布
5. 从词分布中按照概率 $P(w_j|z_k)$ 选择一个词 w_j ”

LDA

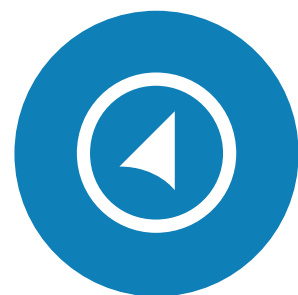
1. 按照先验概率 $P(d_i)$ 选择一篇文档 d_i
2. 从狄利克雷分布（即Dirichlet分布） α 中取样生成文档 d_i 的主题分布 θ_i ，换言之，主题分布 θ_i 由超参数为 α 的Dirichlet分布生成
3. 从主题的多项式分布 θ_i 中取样生成文档 d_i 第 j 个词的主题 $z_{i,j}$
4. 从狄利克雷分布（即Dirichlet分布） β 中取样生成主题 $z_{i,j}$ 对应的词语分布 $\phi_{z_{i,j}}$ ，换言之，词语分布 $\phi_{z_{i,j}}$ 由参数为 β 的Dirichlet分布生成
5. 从词语的多项式分布 $\phi_{z_{i,j}}$ 中采样最终生成词语 $w_{i,j}$ ”

PLSA与LDA对比

pLSA跟LDA的本质区别就在于它们去估计未知参数所采用的思想不同，前者用的是频率派思想，后者用的是贝叶斯派思想。



要点总结



要点1

理解几种主题模型的算法



要点2

pLSA与LDA的对比



02

实战案例

2.1

一眼看穿『希拉里邮件门』
详见随堂附加课件

THANK YOU !

Machine Learning Engineer
机器学习工程师微专业