个人资料

皮果提

+ 加关注 发私信

访问： 1224390次
积分： 19134
等级： BLOG 7
排名： 第454名

原创： 104篇
转载： 13篇
译文： 2篇
评论： 750条

文章分类

数据挖掘 (35)
深度学习 (20)
语言模型 (10)
文本挖掘 (1)
强化学习 (1)
数学天地 (18)
编程知识 (5)
隐马模型 (8)
杂七杂八 (14)
机器学习 (37)
并行计算 (3)

阅读排行

word2vec 中的数学原理详 (93579)
word2vec 中的数学原理详 (55150)
受限玻尔兹曼机（RBM） (42965)
word2vec 中的数学原理详 (39988)
word2vec 中的数学原理详 (39890)
word2vec 中的数学原理详 (39408)
牛顿法与拟牛顿法学习笔 (33615)
牛顿法与拟牛顿法学习笔 (32578)
牛顿法与拟牛顿法学习笔 (32361)
受限玻尔兹曼机（RBM） (31081)

评论排行

word2vec 中的数学原理详 (153)
word2vec 中的数学原理详 (61)
Community Detection 算 (60)

原 word2vec 中的数学原理详解（二）预备知识

标签： word2vec CBOW Skip-gram Hierarchical Softmax Negative Sampling

2014-07-19 22:46 39412人阅读 评论(11) 收藏 举报

分类： 语言模型（9）

word2vec 是 Google 于 2013 年开源推出的一个用于获取 word vector 的工具包，它简单、高效，因此引起了很多人的关注。由于 word2vec 的作者 Tomas Mikolov 在两篇相关的论文 [3,4] 中并没有谈及太多算法细节，因而在一定程度上增加了这个工具包的神秘感。一些按捺不住的人于是选择了通过解剖源代码的方式来一窥究竟，出于好奇，我也成为了他们中的一员。读完代码后，觉得收获颇多，整理成文，给有需要的朋友参考。

相关链接

（一）目录和前言
（二）预备知识
（三）背景知识
（四）基于 Hierarchical Softmax 的模型
（五）基于 Negative Sampling 的模型
（六）若干源码细节

受限玻尔兹曼机（RBM） (48)
word2vec 中的数学原理i (43)
发表在 Science 上的一种 (32)
受限玻尔兹曼机（RBM） (26)
利用 word2vec 训练的字 (22)
受限玻尔兹曼机（RBM） (21)
word2vec 中的数学原理i (20)

最新评论

word2vec 中的数学原理详解（匹
jacksonjack001: @celia01:这个问题貌似楼下有人解释，说跟bp类似！不能求平均！

word2vec 中的数学原理详解（匹
jacksonjack001: @neopenx:没错吧，我看的gensim的源码，sg算法于cbow的主要区别就是在每个当前词处理...

word2vec 中的数学原理详解（匹
jacksonjack001: @m0_37369113:应该没有问题吧，在更新v_{w}的时候已经加上了吧。另外我看过gensim...

受限玻尔兹曼机（RBM）学习笔
DouMiaoO_Oo: 想要请教一下大家，MCMC方法是说在概率分布函数P(X)很复杂的情况下，我们不好直接从分布函数中采样...

word2vec 中的数学原理详解（三
PJ-Javis: 这的确是个坑，我也掉进去了

A Painless Q-learning Tutorial (一
星辰旋风: 赞

word2vec 中的数学原理详解（一
phybrain: 求pdf 楼主 692114871@qq.com

word2vec 中的数学原理详解（六
lreaderl: 楼主，我感觉你的亚采样的公式写的不是很对呀

受限玻尔兹曼机（RBM）学习笔
zuzhangxian7307: @qq_36010258:你好 这边我感觉其实应该是对应状态出现的频数。

受限玻尔兹曼机（RBM）学习笔
zuzhangxian7307: 楼主你好 ，看了楼主的文章有豁然开朗的感觉，另外希望楼主能发一份pdf学习，谢谢!157719785...

关闭

§2 预备知识

本节介绍 word2vec 中将用到的一些重要知识点，包括 sigmoid 函数、Beyes 公式和 Huffman 编码等.

§2.1 sigmoid 函数

sigmoid 函数是神经网络中常用的激活函数之一，其定义为

$$\sigma(x) = \frac{1}{1+e^{-x}},$$

该函数的定义域为 $(-\infty, +\infty)$，值域为 $(0,1)$. 图 1 给出了 sigmoid 函数的图像.

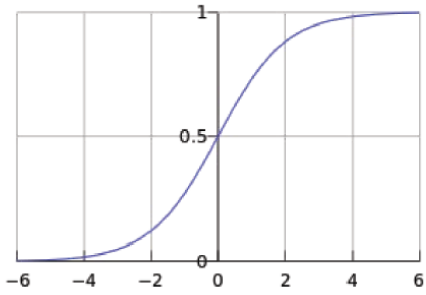

图 1 sigmoid 函数的图像

sigmoid 函数的**导函数**具有以下形式

$$\sigma'(x) = \sigma(x)[1-\sigma(x)],$$

由此易得，函数 $\log\sigma(x)$ 和 $\log(1-\sigma(x))$ 的**导函数**分别为

$$[\log\sigma(x)]' = 1-\sigma(x), \quad [\log(1-\sigma(x))]' = -\sigma(x), \tag{2.1}$$

公式 (2.1) 在后面的推导中将用到.

§2.2 逻辑回归

生活中经常会碰到**二分类问题**，例如，某封电子邮件是否为垃圾邮件，某个客户是否为潜在客户，某次在线交易是否存在欺诈行为，等等. 设 $\{(\mathbf{x}_i, y_i)\}_{i=1}^m$ 为一个二分类问题的样本数据，其中 $\mathbf{x}_i \in \mathbb{R}^n$，$y_i \in \{0,1\}$，当 $y_i = 1$ 时称相应的样本为**正例**，当 $y_i = 0$ 时称相应的样本为**负例**.

利用 sigmoid 函数，对于任意样本 $\mathbf{x} = (x_1, x_2, \cdots, x_n)^\top$，可将二分类问题的 hypothesis 函数写成

$$h_\theta(\mathbf{x}) = \sigma(\theta_0 + \theta_1 x_1 + \theta_2 x_2 + \cdots + \theta_n x_n),$$

其中 $\theta = (\theta_0, \theta_1, \cdots, \theta_n)^\top$ 为待定参数. 为了符号上简化起见，引入 $x_0 = 1$ 将 $\mathbf{x}$ 扩展为 $(x_0, x_1, x_2, \cdots, x_n)^\top$，且在不引起混淆的情况下仍将其记为 $\mathbf{x}$. 于是，h_θ 可简写为

$$h_\theta(\mathbf{x}) = \sigma(\theta^\top \mathbf{x}) = \frac{1}{1+e^{-\theta^\top \mathbf{x}}}.$$

取阀值 $T = 0.5$，则二分类的判别公式为

$$y(\mathbf{x}) = \begin{cases} 1, & h_\theta(\mathbf{x}) \geq 0.5; \\ 0, & h_\theta(\mathbf{x}) < 0.5. \end{cases}$$

那参数 θ 如何求呢？通常的做法是，先确定一个形如下式的**整体损失函数**

$$J(\theta) = \frac{1}{m}\sum_{i=1}^{m} cost(\mathbf{x}_i, y_i),$$

$i=1$

然后对其进行优化, 从而得到最优的参数 θ^*.

实际应用中, 单个样本的损失函数 $cost(\mathbf{x}_i, y_i)$ 常取为**对数似然函数**

$$cost(\mathbf{x}_i, y_i) = \begin{cases} -\log\left(h_\theta(\mathbf{x}_i)\right), & y_i = 1; \\ -\log\left(1 - h_\theta(\mathbf{x}_i)\right), & y_i = 0. \end{cases}$$

注意, 上式是一个分段函数, 也可将其写成如下的整体表达式

$$cost(\mathbf{x}_i, y_i) = -y_i \cdot \log\left(h_\theta(\mathbf{x}_i)\right) - (1 - y_i) \cdot \log\left(1 - h_\theta(\mathbf{x}_i)\right).$$

§2.3 Bayes 公式

贝叶斯公式是英国数学家**贝叶斯** (Thomas Bayes) 提出来的, 用来描述两个条件概率之间的关系. 若记 $P(A)$, $P(B)$ 分别表示事件 A 和事件 B 发生的概率, $P(A|B)$ 表示事件 B 发生的情况下事件 A 发生的概率, $P(A, B)$ 表示事件 A, B 同时发生的概率, 则有

$$P(A|B) = \frac{P(A, B)}{P(B)}, \quad P(B|A) = \frac{P(A, B)}{P(A)},$$

利用上式, 进一步可得

$$P(A|B) = P(A)\frac{P(B|A)}{P(B)},$$

这就是 **Bayes 公式**.

§2.4 Huffman 编码

本节简单介绍 Huffman 编码 (具体内容主要来自百度百科的词条, [10]), 为此, 首先介绍 Huffman 树的定义及其构造算法.

§2.4.1 Huffman 树

在计算机科学中, **树**是一种重要的非线性数据结构, 它是数据元素 (在树中称为**结点**) 按分支关系组织起来的结构. 若干棵互不相交的树所构成的集合称为**森林**. 下面给出几个与树相关的常用概念.

- **路径**和**路径长度**

  在一棵树中, 从一个结点往下可以达到的孩子或孙子结点之间的通路, 称为路径. 通路中分支的数目称为路径长度. 若规定根结点的层号为 1, 则从根结点到第 L 层结点的路径长度为 $L-1$.

- **结点的权**和**带权路径长度**

  若为树中结点赋予一个具有某种含义的 (非负) 数值, 则这个数值称为该结点的权. 结点的带权路径长度是指, 从根结点到该结点之间的路径长度与该结点的权的乘积.

- **树的带权路径长度**

  树的带权路径长度规定为所有叶子结点的带权路径长度之和.

二叉树是每个结点最多有两个子树的有序树. 两个子树通常被称为“**左子树**”和“**右子树**”, 定义中的“有序”是指两个子树有左右之分, 顺序不能颠倒.

给定 n 个权值作为 n 个叶子结点, 构造一棵二叉树, 若它的带权路径长度达到最小, 则称这样的二叉树为**最优二叉树**, 也称为 **Huffman 树**.

§2.4.2 Huffman 树的构造

给定 n 个权值 $\{w_1, w_2, \cdots, w_n\}$ 作为二叉树的 n 个叶子结点, 可通过以下算法来构造一颗 Huffman 树.

算法 2.1 (*Huffman 树构造算法*)

(1) 将 $\{w_1, w_2, \cdots, w_n\}$ 看成是有 n 棵树的森林 (每棵树仅有一个结点).

(2) 在森林中选出两个根结点的权值最小的树合并, 作为一棵新树的左、右子树, 且新树的根结点权值为其左、右子树根结点权值之和.

(3) 从森林中删除选取的两棵树, 并将新树加入森林.

(4) 重复 (2)、(3) 步, 直到森林中只剩一棵树为止, 该树即为所求的 *Huffman* 树.

接下来, 给出算法 2.1 的一个具体实例.

例 2.1 假设 *2014* 年世界杯期间, 从新浪微博中抓取了若干条与足球相关的微博, 经统计, “我”、“喜欢”、“观看”、“巴西”、“足球”、“世界杯”这六个词出现的次数分别为 *15, 8, 6, 5, 3, 1*. 请以这 *6* 个词为叶子结点, 以相应词频当权值, 构造一棵 *Huffman* 树.

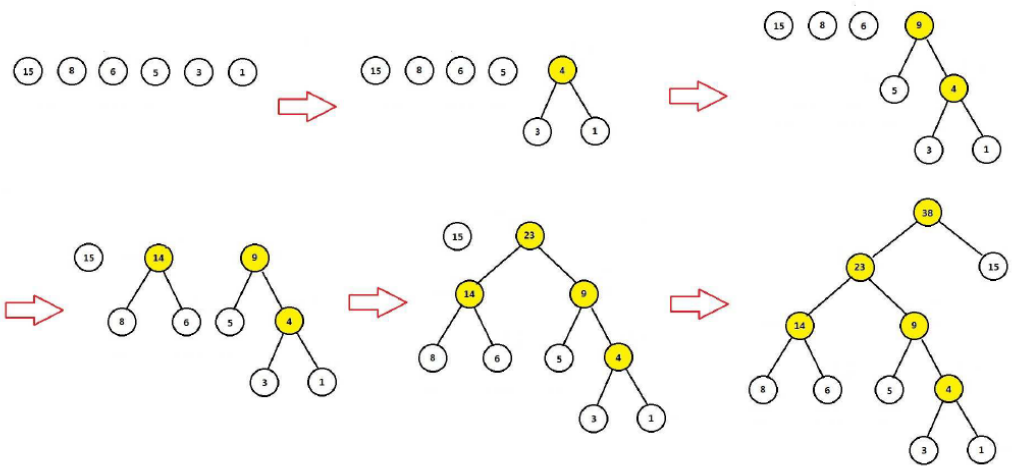

图 2 Huffman 树的构造过程

利用算法 2.1, 易知其构造过程如图 2 所示. 图中第六步给出了最终的 *Huffman* 树, 由图可见**词频越大的词离根结点越近**.

构造过程中, 通过合并新增的结点被标记为黄色. 由于每两个结点都要进行一次合并, 因此, 若叶子结点的个数为 n, 则构造的 *Huffman* 树中新增结点的个数为 $n-1$. 本例中 $n=6$, 因此新增结点的个数为 5.

注意, 前面有提到, 二叉树的两个子树是分左右的, 对于某个非叶子结点来说, 就是其两个孩子结点是分左右的, 在本例中, 统一**将词频大的结点作为左孩子结点, 词频小的作为右孩子结点**. 当然, 这只是一个约定, 你要将词频大的结点作为右孩子结点也没有问题.

§2.4.3 Huffman 编码

在数据通信中, 需要将传送的文字转换成二进制的字符串, 用 0, 1 码的不同排列来表示字符. 例如, 需传送的报文为 "AFTER DATA EAR ARE ART AREA", 这里用到的字符集为 "A, E, R, T, F, D", 各字母出现的次数为 8, 4, 5, 3, 1, 1. 现要求为这些字母设计编码.

要区别 6 个字母, 最简单的二进制编码方式是**等长编码**, 固定采用 3 位二进制 ($2^3=8>6$), 可分别用 000、001、010、011、100、101 对 "A, E, R, T, F, D" 进行编码发送, 当对方接收报文时再按照三位一分进行译码.

显然编码的长度取决报文中不同字符的个数. 若报文中可能出现 26 个不同字符, 则固定编码长度为 5 ($2^5=32>26$). 然而, 传送报文时总是希望总长度尽可能短. 在实际应用中, 各个字符的**出现频度**或**使用次数**是不相同的, 如 A、B、C 的使用频率远远高于 X、Y、Z, 自然会想到设计编码时, 让使用频率高的用短码, 使用频率低的用长码, 以优化整个报文编码.

为使**不等长编码**为**前缀编码** (即要求一个字符的编码不能是另一个字符编码的前缀), 可用字符集中的每个字符作为叶子结点生成一棵编码二叉树, 为了获得传送报文的最短长度, 可将每个字符的出现频率作为字符结点的权值赋予该结点上, 显然字使用频率越小权值越小, 权值越小叶子就越靠下, 于是频率小编码长, 频率高编码短, 这样就保证了此树的最小带权路径长度, 效果上就是传送报文的最短长度. 因此, 求传送报文的最短长度问题转化为求由字符集中的所有字符作为叶子结点, 由字符出现频率作为其权值所产生的 Huffman 树的问题. 利用 Huffman 树设计的二进制前缀编码, 称为 **Huffman 编码**, 它既能满足前缀编码的条件, 又能保证报文编码总长最短.

本文将介绍的 word2vec 工具中也将用到 Huffman 编码, 它把训练语料中的词当成叶子结点, 其在语料中出现的次数当作权值, 通过构造相应的 Huffman 树来对每一个词进行 Huffman 编码.

图 3 给出了例 2.1 中六个词的 Huffman 编码, 其中约定 (词频较大的) 左孩子结点编码为 1, (词频较小的) 右孩子编码为 0. 这样一来, “我”、“喜欢”、“观看”、“巴西”、“足球”、“世界杯” 这六个词的 Huffman 编码分别为 0, 111, 110, 101, 1001 和 1000.

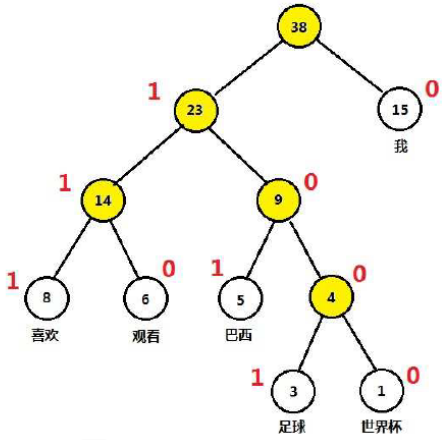

图 3 Huffman 编码示意图

注意, 到目前为止, 关于 Huffman 树和 Huffman 编码, 有两个约定: (1) 将权值大的结点作为左孩子结点, 权值小的作为右孩子结点; (2) 左孩子结点编码为 1, 右孩子结点编码为 0. 在 word2vec 源码中将权值较大的孩子结点编码为 1, 较小的孩子结点编码为 0. 为与上述约定统一起见, **下文中提到的 “左孩子结点” 都是指权值较大的孩子结点**.

参考文献

[1] David E Rumelhart, Geoffrey E Hintont, and Ronald J Williams. **Learning representations by backpropagating errors.** Nature, 323(6088):533-536, 1986.

[2] Yoshua Bengio, Rejean Ducharme, Pascal Vincent, and Christian Jauvin. **A neural probabilistic language model.** Journal of Machine Learning Research (JMLR), 3:1137-1155, 2003.

[3] Tomas Mikolov, Kai Chen, Greg Corrado, Jeffrey Dean. **Efficient Estimation of Word Representations in Vector Space.** arXiv:1301.3781, 2013.

[4] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg Corrado, Jeffrey Dean. **Distributed Representations of Words and Phrases and their Compositionality.** arXiv:1310.4546, 2013.

[5] Tomas Mikolov, Quoc V. Le, Ilya Sutskever. **Exploiting Similarities among Languages for Machine Translation.** arXiv:1309.4168v1, 2013.

[6] Quoc V. Le, Tomas Mikolov. **Distributed Representations of Sentences and Documents.** arXiv:1405.4053, 2014.

[7] Xiaoqing Zheng, Hanyang Chen, Tianyu Xu. **Deep Learning for Chinese Word Segmentation and POS tagging.** Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing, pages 647-657.

[8] Ronan Collobert, Jason Weston, Léon Bottou, Michael Karlen, Koray Kavukcuoglu and Pavel Kuksa. **Natural Language Processing (Almost) from Scratch.** Journal of Machine Learning Research (JMLR), 12:2493-2537, 2011.

[9] Michael U Gutmann and Aapo Hyvärinen. Noise-contrastive estimation of unnormalized statistical models, with applications to natural image statistics. The Journal ofMachine Learning Research, 13:307-361, 2012.

[10] 百度百科中的“哈夫曼树”词条.

[11] 吴军.《**数学之美**》. 人民邮电出版社, 2012.

[12] http://ml.nec-labs.com/senna/

[13] http://www.looooker.com/archives/5621

[14] licstar. **Deep Learning in NLP (一) 词向量和语言模型**.
http://licstar.net/archives/328

[15] **深度学习 word2vec 笔记之基础篇**.
http://blog.csdn.net/mytestmy/article/details/26961315

[16] 深度学习 word2vec 笔记之算法篇.
http://blog.csdn.net/mytestmy/article/details/26969149

[17] 邓澍军, 陆光明, 夏龙. Deep Learning 实战之 word2vec, 2014.

[18] 杨超. Word2Vec 的一些理解.
http://www.zhihu.com/question/21661274/answer/19331979

[19] 基于权值的微博用户采样算法研究.
http://blog.csdn.net/itplus/article/details/9079297

[20] 利用 word2vec 训练的字向量进行中文分词.
http://blog.csdn.net/itplus/article/details/17122431

[21] Yoav Goldberg, Omer Levy. word2vec Explained: Deriving Mikolov et al.'s Negative-Sampling Word-Embedding Method. arXiv: 1402.3722v1, 2014. (http://arxiv.org/pdf/1402.3722v1.pdf)

上一篇 word2vec 中的数学原理详解（一）目录和前言
下一篇 word2vec 中的数学原理详解（三）背景知识

相关文章推荐

- word2vec 中的数学原理详解（三）背景知识
- 【直播】70天软考冲刺计划--任铄
- word2vec原理解析
- 【直播】打通Linux脉络 进程、线程、调度--宋宝华
- word2vec 中的数学原理详解（二）预备知识
- 【直播】机器学习之凸优化--马博士
- word2vec中的数学原理详解
- 【套餐】MATLAB基础+MATLAB数据分析与统计-...
- word2vec数学原理
- 【课程】3小时掌握Docker最佳实战--徐西宁
- word2vec 中的数学原理详解（五）基于 Negative...
- 【课程】深度学习基础与TensorFlow实践--AI100
- Word2Vec数据集
- word2vec工具下载
- word2vec-master
- word2vec源文件

谢谢楼主分享，很通俗易懂，把复杂的东西讲的这么生动的不多哦。

7楼 helh522 2017-01-05 16:09发表

求pdf版本 楼主赞！thx~ 534599152@qq.com

6楼 ustcer_iim 2016-11-07 20:51发表

您好，您写的太好了，请问可以发一份pdf版的给我吗？邮箱gaojin35@mail.ustc.edu.cn 谢谢您！

5楼 xjl19880927 2015-12-04 14:21发表

博主，来份RNN原理详解

4楼 阿良田木 2015-11-19 22:48发表

求博主的pdf文档啊，chenwangliangguo@qq.com

3楼 CATEMALIN 2015-10-30 12:59发表

你好，感谢你全面的讲解 求一份pdf。 邮箱是 migowei0621@163.com 谢谢

2楼 cgogonlp 2015-06-15 11:43发表

你好，可以把这篇文章的PDF版本发给我吗？我的邮箱luo17@126.com，谢谢！

1楼 liangmin0020163com 2014-11-13 17:26发表

你好！感谢您由浅入深的讲解，和无私的分享。能把pdf版本共享我吗？我的邮箱 liangmin0020@163.com 谢谢！

您还没有登录,请[登录]或[注册]

* 以上用户言论只代表其个人观点，不代表CSDN网站的观点或立场

关闭