中文分词，即 Chinese Word Segmentation(CWS).有两种理解方式:
1.将一个汉字序列进行切分,得到一个个单独的词。
2.对连续的字进行合理的合并,得到一个个单独的词。

中文分词的几大困难:
1.新词发现
2.歧义
3.可靠的语料少

中文分词方法的划分:
1.基于规则的中文分词方法
2.基于统计的中文分词方法
3.基于语义的中文分词方法
4.基于理解的中文分词方法

实际上也可理解为三种:
1.基于字典的中文分词方法
2.基于统计的中文分词方法
3.基于深度学习的中文分词方法

实际上好的切词方法往往将这三者结合起来的。

参考:1,2

基于规则的中文分词方法

这种方法也叫机械分词方法、基于字典的分词方法，按一定的策略将待分析的汉字串与词典中的词条进行匹配，若字典中找到某个字符串,则匹配成功。该策略的三要素为:分词词典、文本扫描顺序和匹配规则。
文本扫描顺序有正向扫描、逆(反)向扫描和双向扫描。
匹配原则主要有最大匹配、最小匹配、逐词匹配和最佳匹配。

- 正向最大匹配法（FMM）。基本思想是：假设自动分词词典中的最长词条所含汉字的个数为 i，则取被处理材料当前字符串序列中的前i个字符作为匹配字段，查找分词词典，若词典中有这样一个i字词，则匹配成功，匹配字段作为一个词被切分出来；若词典中找不到这样的一个i字词，则匹配失败，匹配字段去掉最后一个汉字，剩下的字符作为新的匹配字段，再进行匹配，如此进行下去，直到匹配成功为止。统计结果表明，该方法的错误率为 1/169.

- 逆向最大匹配法（RMM）。该方法的分词过程与 MM 法相同，不同的是从句子（或文章）末尾开始处理，每次匹配不成功时去掉的是前面的一个汉字。统计结果表明，该方法的错误率为 1/245。注:有文献表明，中文更适合逆向最大匹配法。
  代码实现demo

- 逐词遍历法。把词典中的词按照由长到短递减的顺序逐字搜索整个待处理的材料，一直到把全部的词切分出来为止。不论分词词典多大，被处理的材料多么小，都得把这个分词词典匹配一遍。这种方法保证长的词可以分出来，处理速度较慢。

- 设立切分标志法。切分标志有自然和非自然之分。自然切分标志是指文章中出现的非文字符号，如标点符号等；非自然标志是利用词缀和不构成词的词（包 括单音词、复音节词以及象声词等）。设立切分标志法首先收集众多的切分标志，分词时先找出切分标志，把句子切分为一些较短的字段，再用MM、RMM 或其它的方法进行细加工。这种方法并非真正意义上的分词方法，只是自动分词的一种前处理方式而已，它要额外消耗时间扫描切分标志，增加存储空间存放那些非自然切分标志。
  注:自然切分标志法在医学文本领域还是挺有用的，可以根据句号、逗号来判断对于某个事物的描述是否结束。

- 最佳匹配法（OM）。此法分为正向的最佳匹配法和逆向的最佳匹配法，其出发点是：在词典中按词频的大小顺序排列词条，以求缩短对分词词典的检索时间，达到最佳效果，从而降低分词的时间复杂度，加快分词速度。实质上，这种方法也不是一种纯粹意义上的分词方法，它只是一种对分词词典的组织方式。OM法的分词词典每条词的前面必须有指明长度的数据项，所以其空间复杂度有所增加，对提高分词精度没有影响，分词处理的时间复杂度有所降低。

- N最短路径方法
  最短路径法 假设待切词的句子是：
  提高人名生活水平：
  这个字符串可以构成一个有向图：节点为句子首尾两端和字与字之间的间隔，上面一句话有8个字也就是有9个节点.在前面的节点可以指向后面的节点，两个节点之间的连线即可认为是边，至于边的权重有多种计算方式：
  若该边构成的词在字典里出现了，则该边的权重记为1，否则记为0.
  把-ln(词的频率)作为边权重，那么求最短路径，便转化为求最大概率P(w1,w2,…,wn).
  n-gram:把-P(wi|w_{i-1},w_{i-2},w_{i-(n-1)})作为边的权重。

> N最短路径是在头到尾所有可能的路径中找出前N个最短路径。也就是N种分词结果作为粗分结果集。
> https://blog.csdn.net/thealgorithmart/article/details/6876871

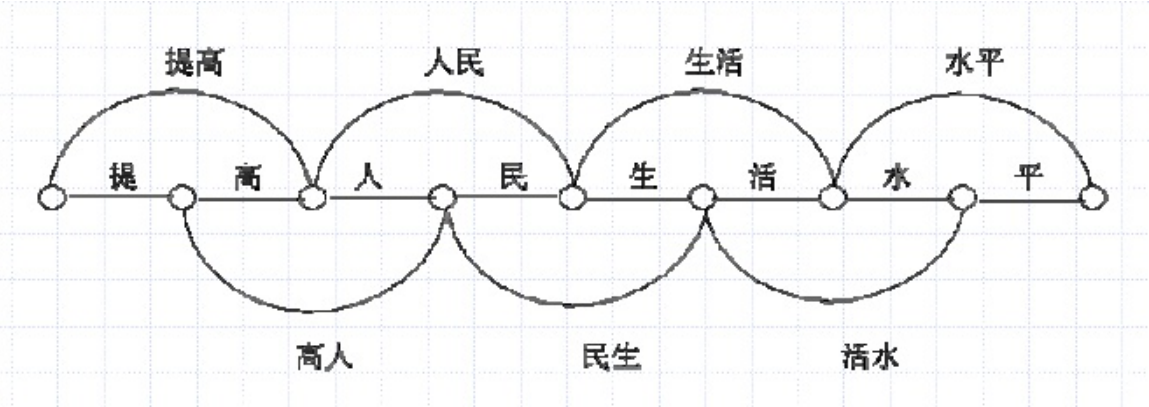
- 最少切分
  使每一句中切出的词数最小

> 考虑到中文分词存在切分歧义消除和未登录词识别两个主要问题。因此可将分词分成两个阶段：
> 1.用分词算法进行粗分
> 2.对粗分的结果进行歧义消除和未登录词识别取最好的粗分结果。

> 组合型歧义就是对于字串AB，可以切分为AB，又可以切分为A/B；交集型歧义就是

> ABC，可以切分为AB/C，又可以切分为A/BC。这其中交集型歧义有占了绝大多数，据统计达94%，因此处理好交集型歧义在汉语分词中有着非常重要的地位。
> https://blog.csdn.net/zoohua/article/details/4567802

消除歧义
因为同一个句子，在机械分词中经常会出现多种分词的组合，因此需要进行歧义消除，来得到最优的分词结果。相关算法有MMSEG机械分词算法等。

> 歧义消解还可以转换为对于上述在词图上寻找统计意义上的最佳路径。常用一元、二元模型进行。
>
> 
>

> 基于一元模型进行评价：
> 统计词表中每个词的词频，并将其转化为路径代价C=-log(f/N)切分路径的代价为路径上所有词的代价之和寻求代价最小的路径。上述例子就是根据词典中<提高><高人><人民><民生><生活><活水><水平><平>这几个词的词频f，认为词频越高的路径代价越小，找出最短的路径。
> 基于二元模型进行评价：
> 相对于一元模型，二元模型还需要一个词转移统计词典，例如记录了<提高>衔接<人民>的次数，词转移统计词典实质上是一个稀疏矩阵。基于二元模型进行评价需要在一元模型的基础上增加转移路径代价。词典中转移次数多的衔接认为该衔接转移路径代价小。计算方法可以用Viterbi算法。

参考3

基于规则的中文分词方法优点是简单，易于实现。但缺点有很多：匹配速度较慢(当然肯定比深度学习分词快)；存在交集型和组合型歧义切分问题；词本身没有一个标准的定义，没有统一标准的词集；不同词典产生的歧义也不同；缺乏学习未登陆词(即新词)的能力。

基于统计的中文分词方法

该方法目前是主流的中文分词方法。该方法假设:字与字相邻(或n-gram相邻)的概率(由频率近似)与成词的可信度是正相关的。一种方法是可以对训练文本中相邻出现的各个字的组合的频度进行统计，计算它们之间的互现信息。互现信息体现了汉字之间结合关系的紧密程度。当紧密程度高

于某一个阈值时，便可以认为此字组可能构成了一个词。该方法又称为无字典分词。

该方法所应用的主要的统计模型有：N元文法模型（N-gram）、隐马尔可夫模型（Hiden Markov Model，HMM）、最大熵模型（ME）、条件随机场模型（Conditional Random Fields，CRF）等。

HMM、CRF把分词视为序列标注任务。
CRF的效果要好于HMM。
在实际应用中此类分词算法一般是将其与基于词典的分词方法结合起来，既发挥匹配分词切分速度快、效率高的特点，又利用了无词典分词结合上下文识别生词、自动消除歧义的优点。

例如，比较主流的中文切词框架jieba,采用的是HMM与词典结合的方法。
HMM分词C++实现

基于语义的中文分词方法

语义分词法引入了语义分析，对自然语言自身的语言信息进行更多的处理，如扩充转移网络法、知识分词语义分析法、邻接约束法、综合匹配法、后缀分词法、特征词库法、矩阵约束法、语法分析法等。

- 扩充转移网络法。该方法以有限状态机概念为基础。有限状态机只能识别正则语言，对有限状态机作的第一次扩充使其具有递归能力，形成递归转移网络 （RTN）。在RTN 中，弧线上的标志不仅可以是终极符（语言中的单词）或非终极符（词类），还可以调用另外的子网络名字分非终极符（如字或字串的成词条件）。这样，计算机在 运行某个子网络时，就可以调用另外的子网络，还可以递归调用。词法扩充转移网络的使用， 使分词处理和语言理解的句法处理阶段交互成为可能，并且有效地解决了汉语分词的歧义。
- 矩阵约束法。其基本思想是：先建立一个语法约束矩阵和一个语义约束矩阵， 其中元素分别表明具有某词性的词和具有另一词性的词相邻是否符合语法规则， 属于某语义类的词和属于另一词义类的词相邻是否符合逻辑，机器在切分时以之约束分词结果。

基于理解的中文分词方法

基于理解的分词方法是通过让计算机模拟人对句子的理解，达到识别词的效果。其基本思想就是在分词的同时进行句法、语义分析，利用句法信息和语义信息来处理歧义现象。它通常包括三个部分：分词子系统、句法语义子系统、总控部分。在总控部分的协调下，分词子系统可以获得有关词、句子等的句法和语义信息来对分词歧义进行判断，即它模拟了人对句子的理解过程。这种分词方法需要使用大量的语言知识和信息。目前基于理解的分词方法主要有专家系统分词法和神经网络分词法等。

- 专家系统分词法。从专家系统角度把分词的知识（包括常识性分词知识与消除歧义切分的启发性知识即歧义切分规则）从实现分词过程的推理机中独立出来，使知识库的维护与推理机的实现互不干扰，从而使知识库易于维护和管理。它还具有发现交集歧义字段和多义组合歧

义字段的能力和一定的自学习功能。

- 神经网络分词法。例如基于词感知机的分词方法、基于深度学习的端到端的分词方法
  BiLSTM-HMM
  介绍
  代码
  CNN-HMM-硬解码
  介绍
  代码
  BiLSTM-CNN-CRF
  介绍
  代码
  LSTM-CNNs-CRF Ma X, Hovy E. End-to-end sequence labeling via bi-directional lstm-cnns-crf[J]. arXiv preprint arXiv:1603.01354, 2016.
  BiLSTM-CRF Lample G, Ballesteros M, Subramanian S, et al. Neural Architectures for Named Entity Recognition[J]. 2016:260-270.

- 神经网络专家系统集成式分词法。该方法首先启动神经网络进行分词，当神经网络对新出现的词不能给出准确切分时，激活专家系统进行分析判断，依据知识库进行推理，得出初步分析，并启动学习机制对神经网络进行训练。该方法可以较充分发挥神经网络与专家系统二者优势，进一步提高分词效率。

参考4,5

分词框架

1.jieba
专用于中文分词的 Python 库
GitHub：https://github.com/fxsjy/jieba

jieba 支持繁体分词，支持自定义词典。

其使用的算法是基于统计的分词方法，主要有如下几种：

基于前缀词典实现高效的词图扫描，生成句子中汉字所有可能成词情况所构成的有向无环图 (DAG)
采用了动态规划查找最大概率路径, 找出基于词频的最大切分组合
对于未登录词，采用了基于汉字成词能力的 HMM 模型，使用了 Viterbi 算法。

jieba默认开启新词发现(HMM = True)，可以设置关闭。
HMM = False时，采用DAG + Uni-gram的语言模型 + 后向DP的方式进行。

词性标记集见:https://gist.github.com/luw2007/6016931

2.SnowNLP
SnowNLP: Simplified Chinese Text Processing，可以方便的处理中文文本内容，是受到了 TextBlob 的启发而写的，由于现在大部分的自然语言处理库基本都是针对英文的，于是写了一个方便处理中文的类库，并且和 TextBlob 不同的是，这里没有用 NLTK，所有的算法都是自己实现的，并且自带了一些训练好的字典。GitHub地址：https://github.com/isnowfy/snownlp。
采用的中文分词方法是:Character-Based Generative Model
另外 SnowNLP 还支持很多功能，例如词性标注（HMM）、情感分析、拼音转换（Trie树）、关键词和摘要生成（TextRank）。

3.THULAC
THULAC（THU Lexical Analyzer for Chinese）由清华大学自然语言处理与社会人文计算实验室研制推出的一套中文词法分析工具包，GitHub 链接：https://github.com/thunlp/THULAC-Python 具有中文分词和词性标注功能。THULAC具有如下几个特点：

- 语料多。利用集成的目前世界上规模最大的人工分词和词性标注中文语料库（约含5800万字）训练而成，模型标注能力强大。注:jieba在评测中准确度不如THULAC,jieba的作者认为是字典的问题（也就是语料不够多）。
- 准确率高。该工具包在标准数据集Chinese Treebank（CTB5）上分词的F1值可达97.3%，词性标注的F1值可达到92.9%，与该数据集上最好方法效果相当。
- 速度较快。同时进行分词和词性标注速度为300KB/s，每秒可处理约15万字。只进行分词速度可达到1.3MB/s。

4.NLPIR
NLPIR 分词系统，前身为2000年发布的 ICTCLAS 词法分析系统，GitHub链接。北京理工大学张华平博士研发的中文分词系统，经过十余年的不断完善，拥有丰富的功能和强大的性能。NLPIR 是一整套对原始文本集进行处理和加工的软件，提供了中间件处理效果的可视化展示，也可以作为小规模数据的处理加工工具。主要功能包括：中文分词，词性标注，命名实体识别，用户词典、新词发现与关键词提取等功能。另外对于分词功能，它有 Python 实现的版本，GitHub 链接：https://github.com/tsroten/pynlpir。

5.NLTK
NLTK，Natural Language Toolkit，是一个自然语言处理的包工具，NLP处理相关功能相当全面，GitHub 链接：https://github.com/nltk/nltk。

但是 NLTK 对于中文分词是不支持的。

6.LTP
语言技术平台（Language Technology Platform，LTP）是哈工大社会计算与信息检索研究中心历时十年开发的一整套中文语言处理系统。LTP制定了基于XML的语言处理结果表示，并在此基础上提供了一整套自底向上的丰富而且高效的中文语言处理模块（包括词法、句法、语义等6项中文处理核心技术），以及基于动态链接库（Dynamic Link Library, DLL）的应用程序接口、可

视化工具，并且能够以网络服务（Web Service）的形式进行使用。

LTP 有 Python 版本，GitHub地址。另外运行的时候需要下载模型，模型还比较大，下载地址：http://ltp.ai/download.html

7.Ansj
pass

8.Hanlp
这个框架相当全面。
Hanlp支持基于HMM模型的分词、支持索引分词、繁体分词、简单匹配分词（极速模式）、基于CRF模型的分词、N-最短路径分词等。实现了不少经典分词方法。

9.ICTCLAS
出现较早，现在已经是商业系统了 (改名 NLPIR)，需要 License 才能运行。

> 从未登录词识别准确率上说，ICTCLAS 已经明显落后于基于 CRF 的分词系统了。尽管如此，它的优点仍然比较明显：很少出现“错得离谱”的切分结果，这在基于 CRF 模型的分词系统上不少见，尤其是迁移到其它领域时；模型和库不大，启动快；基于 C++ 实现，能够很快迁移到其它语言。
> 从分词稳定性上来说，ICTCLAS 值得信赖，从分词准确率、分词速度等方面来考量，有不少分词系统超过了它；NLPIR 的源代码已经不再开放，这让用户很纠结。

10.交大分词
不开源

11.Stanford 分词

> Stanford 分词系统的优点是准确率高，未登录词识别能力比较强；缺点非常明显，模型很大，约 300MB-400MB，启动非常慢，大概需要 10 秒 -20 秒。在所有分词系统中，没有比 Stanford 启动更慢的系统，分词速度也不快。代码优化的空间比较大。
> Stanford 系统支持自定义训练，只要用户提供训练数据，该系统可以训练新的模型参数。
> Stanford 分词系统只是验证作者论文的一种手段，为了非常微小的分词准确率提升，导致了模型参数膨胀。
> 在 Demo 环境下可以使用 Stanford 系统，在大规模数据环境下不适合使用该系统。

12.GPWS
不开源

13.IK
pass.维护人员少

14.百度

https://github.com/baidu/lac 百度中文词法分析（分词+词性+专名）系统
据说很强.

15.其他的一些商业系统

中文分词框架评测

> 在中文分词领域，比较权威且影响深远的评测有 SIGHAN - 2nd International Chinese Word Segmentation Bakeoff。它提供了2份简体中文和2份繁体中文的分词评测语料。

参考6,7

THULAC的评测

THULAC准确度高，jieba速度快。

工程实践

这个帖子的这一部分讲的挺好,如何工程解决?

> 能用规则解决的，就不要靠模型了
> 扩大训练语料
> 增加词表
> 最大匹配 + 大词表
> 实用的分词系统，都带有大量通用词表和领域词表。
> 从最新文献来看，利用神经网络来做分词，训练效率和运行效率都比较低，慢得无法忍受，不适合工程上部署，也不适合做 Demo。
> 根据个人经验，神经网络在 NLP 上的成功应用的领域往往是准确率不高或者运行效率很低的场合，例如问答系统、机器翻译、句法分析。在准确率比较高或者运行效率不错的场景下，利用深度学习会得不偿失。

公开的分词语料

微软亚洲研究院的语料 Bakeoff-2006 Bakeoff-2003

2011年人民日报的语料

Reference:
[1]https://zhuanlan.zhihu.com/p/29449441 (可以先看这篇有个大概的了解)
[2]https://zhuanlan.zhihu.com/p/35094414
[3]https://zhuanlan.zhihu.com/p/22047433
[4]

https://zhuanlan.zhihu.com/p/33261835
[5]https://zhuanlan.zhihu.com/p/35094414
[6]https://zhuanlan.zhihu.com/p/33261835
[7]
https://zhuanlan.zhihu.com/p/24119153
[8]martix67 漫话中文自动分词和语义识别（上）：中文分词算法
[9]中文分词系列1-基于AC自动机的快速分词
[10]中文分词系列2-基于切分的新词发现
[11]中文分词系列3-字标注法与HMM模型
[12]中文分词系列4-基于双向LSTM的seq2seq字标注
[13]中文分词系列5-基于语言模型的无监督分词
[14]中文分词系列6-基于全卷积网络的中文分词
[15]中文分词系列7-深度学习分词？只需一个词典
[16]中文分词系列-8更好的新词发现算法
[17]统计自然语言处理-第二版
[18]Speech and Language Processing-3rd

[19]深度学习中文分词调研

2001年写的,当时还没有深度学习分词
[20]汉语自动分词研究评述. 孙茂松

2007年写的。
[21]中文分词十年回顾,黄昌宁

[22]中文分词十年又回顾: 2007-2017,黄昌宁

相关词汇:
oov（out of vocabulary）