Atitit snownlp nlp 常见功能

目录

[1.1. 主要功能： 1](#_Toc21026)

[1.2. 官网信息： 2](#_Toc8381)

*[1.3. # 自动摘要 vs 关键词提取](#_Toc9800)* [2](#_Toc9800)

[1.4. Tf idf算法 2](#_Toc28739)

[1.5. print('文本相似：') 3](#_Toc13934)

[1.6. 中文分词： 3](#_Toc7115)

[1.7. 词性标注： 4](#_Toc6024)

[1.8. 情感分析： 4](#_Toc26242)

[1.9. 转成拼音： 4](#_Toc8695)

[1.10. 繁体转简体： 4](#_Toc27037)

[1.11. 关于训练（分词，词性标注，情感分析）： 4](#_Toc2823)

## ****主要功能：****

* 中文分词（[Character-Based Generative Model](http://aclweb.org/anthology//Y/Y09/Y09-2047.pdf" \t "http://www.ppkanshu.com/index.php/post/_blank)）
* 词性标注（[TnT](http://aclweb.org/anthology//A/A00/A00-1031.pdf" \t "http://www.ppkanshu.com/index.php/post/_blank) 3-gram 隐马）
* 情感分析（现在训练数据主要是买卖东西时的评价，所以对其他的一些可能效果不是很好，待解决）
* 文本分类（Naive Bayes）
* 转换成拼音（Trie树实现的最大匹配）
* 繁体转简体（Trie树实现的最大匹配）
* 提取文本关键词（[TextRank](http://acl.ldc.upenn.edu/acl2004/emnlp/pdf/Mihalcea.pdf" \t "http://www.ppkanshu.com/index.php/post/_blank)算法）
* 提取文本摘要（[TextRank](http://acl.ldc.upenn.edu/acl2004/emnlp/pdf/Mihalcea.pdf" \t "http://www.ppkanshu.com/index.php/post/_blank)算法）
* tf，idf
* Tokenization（分割成句子）
* 文本相似（[BM25](http://en.wikipedia.org/wiki/Okapi_BM25" \t "http://www.ppkanshu.com/index.php/post/_blank)）
* 支持python3（感谢[erning](https://github.com/erning" \t "http://www.ppkanshu.com/index.php/post/_blank)）

## ****官网信息：****

## *# 自动摘要 vs 关键词提取*

text = **'自然语言处理是计算机科学领域与人工智能领域中的一个重要方向。它研究能实现人与计算机之间用自然语言进行有效通信的各种理论和方法。自然语言处理是一门融语言学、计算机科学、数学于一体的科学。因此，这一领域的研究将涉及自然语言，即人们日常使用的语言，所以它与语言学的研究有着密切的联系，但又有重要的区别。自然语言处理并不是一般地研究自然语言，而在于研制能有效地实现自然语言通信的计算机系统，特别是其中的软件系统。因而它是计算机科学的一部分。'**s = SnowNLP(text)  
print(s.keywords(10))  
print(s.summary(3))  
*## SnowNLP.summary*

## Tf idf算法

s = SnowNLP([[u'这篇', u'文章'],

[u'那篇', u'论文'],

[u'这个']])print(s.tf) #TF意思是词频(Term Frequency)print(s.idf) #IDF意思是逆文本频率指数(Inverse Document Frequency)

## print('文本相似：')

print(s.sim(['文章']))*# [0.3756070762985226, 0, 0]*

print(s.sim(['文章','真']))*# [0.7731414846187967, 0, 0*

1. 词频：
2. [{'这篇': 1, '文章': 1, '真': 1, '不错': 1}, {'那篇': 1, '论文': 1}, {'这个': 1}]
4. 逆向文件频率：
5. {'这篇': 0.5108256237659907, '文章': 0.5108256237659907, '真': 0.5108256237659907, '不错': 0.5108256237659907, '那篇': 0.5108256237659907, '论文': 0.5108256237659907, '这个': 0.5108256237659907}
7. 文本相似：
8. [0.38657074230939836, 0, 0]
9. ---------------------

## 中文分词：

['这个', '东西', '真心', '很', '赞']

## 词性标注：

<zip object at 0x12638b388>

## 情感分析：

0.9769551298267365

## 转成拼音：

['zhe', 'ge', 'dong', 'xi', 'zhen', 'xin', 'hen', 'zan']

## 繁体转简体：

「繁体字」「繁体中文」的叫法在台湾亦很常见。

提取文本关键词：

---------------------

## 关于训练（分词，词性标注，情感分析）：

from snownlp import seg

seg.train('data.txt')

seg.save('seg.marshal')

# from snownlp import tag

# tag.train('199801.txt')

# tag.save('tag.marshal')

# from snownlp import sentiment

# sentiment.train('neg.txt', 'pos.txt')

---------------------

(9+条消息)NLP snownlp 实际用例 - \_yuki\_ - CSDN博客.html