# 文本话题识别算法研究与实现 [>>](marginnote3app://note/A53BB89D-6DF1-4F4E-B3CA-C340456EA4E5)

## 1 绪论 [>>](marginnote3app://note/9839A4DE-0BD9-4B39-9592-EEB8DFB3F276)

### 基本思想 [>>](marginnote3app://note/C903791A-493F-45D5-AF81-5E9C3E679405) 本算法的主要脉络是通过利用有监督学习对文本进行筛查后，使用 LDA（Latent Dirichlet Allocation）文档主题生成模型对已经筛查后的具有新闻价值的文本进行主题聚类， 根据 LDA 模型得到的主题-文档相关性关系， 选择合适的无监督学习算法对文档进行聚类， 确定其最终的主题和划分， 最终实现从海量的新闻语料中提取出有限的主题并对主题进行描述这一研究目标。

### 文本聚类三种方法 [>>](marginnote3app://note/5F421FB4-9912-4A44-B3DA-7AA7D0FF05EC)

#### 基于向量空间模型（Vector Space Model，VSM） [>>](marginnote3app://note/FC2AB5FB-B820-47E6-B6A4-BB8BD004DA9B) 依靠文本中词频的相关统计数据来实现文本聚类方法的方法， 这种传统的方法的代表就是基于向量空间模型（Vector Space Model，VSM）的文本聚类方法。

#### 引入外部语义知识 [>>](marginnote3app://note/5D22B94E-98FE-416C-A1D3-01DD7A3196FF) 引入外部语义知识。例如可以利用 Wikipedia 进行概念知识库的构造，将数据集的特征词空间映射到概率知识空间上， 结合词相似度、 概率知识相似度及类别相似度来加强文本相似度的计算[2]；或者利用 Wikipedia 检索文本集，提取 Wikipedia 中自有的类别信息、概念信息等来辅助加强计算文本相似度[3]。

#### 主题模型 [>>](marginnote3app://note/0BC268FA-1F6E-47DB-A6B5-584C39EEB8AF) 挖掘文档内部的语义知识。 而语义挖掘的利器正是主题模型[4]。 当前在这个领域主流的主题模型可以分别有三类， 即： LSA （Latent Semantic Analysis， 隐形语义分析）[5]、 PLSI （ probabilistic latent semantic analysis））[6]和 LDA（Latent Dirichlet Allocation，隐形狄利克雷分布）[7]。

##### LDA 主题模型的优点 [>>](marginnote3app://note/0D3B1345-F9C1-4494-9121-6A8D65FA0117) 对比 LSA 和 PLSI 这两种主题模型，LDA 主题模型的优点在于它是一个完全概率生成模型， 模型的概率算法理论非常成熟， 因而相对适用于大规模的文本数据集。

### 实验内容 [>>](marginnote3app://note/9B8F45EA-A541-453A-8049-1D4337C60450) 本文的实验文本数据来源于项目合作公司对于微信公众号的部分爬取数据。 对于这样一类包含了大量无用信息，内聚性、关联性均较弱的文本数据，最终的研究目标是实现从原始数据垃圾清洗到最终主题聚类的一套完整的算法， 其主要包含有以下几个部分：

#### 文本预处理算法 [>>](marginnote3app://note/369C9A17-4420-45CE-9231-7025B8468485) 1、文本预处理算法， 包括对实验文本进行中文分词、 停用词去除， 将连续、繁杂的原始文本信息转化为特征词集合的文档形式，以方便后续算的处理

#### 文档垃圾信息过滤 [>>](marginnote3app://note/394453D4-23A9-4E06-801E-40484D4F4D35) 2、文档垃圾信息过滤，通过对部分文档进行人工标定获取训练集，在其上使用文本分类算法训练模型对全部原始数据进行分类以过滤掉原始文本中含有的大量无用、无价值的文档；

#### 主题聚类数量确定方法 [>>](marginnote3app://note/189D5A15-ACBB-4AF6-8CF5-A9218DCB56F0) 3、主题聚类数量确定方法，主要通过根据实验数据得到的聚类有效性函数与最终主题聚类结果人工评定指标之间的经验关系， 结合当前可能的特别要求来确定最合适的聚类数量 T；

#### 各文档间相似度计算 [>>](marginnote3app://note/F3229B53-8EB3-4EF0-B399-DE682594D812) 4、各文档间相似度计算，主要使用 LDA 模型算法实现，同时参考 VSM 模型下各文档相似度，结合得到最终的文档相似度；

#### 算法实现对文档进行主题聚类 [>>](marginnote3app://note/1D865078-FFF6-425D-8AD3-39E031AF70E5) 5、根据相似度，选择实现适当的算法实现对文档进行主题聚类，得到最终主题的描述（关键词描述）和各文档的主题划分结果。

## 2 相关理论与关键技术概述 [>>](marginnote3app://note/04DCFEF5-D798-41A3-B82C-95602192108A)

### 文本预处理 [>>](marginnote3app://note/F199559E-E782-403D-9130-974541CD6AC1)

#### 中文分词 [>>](marginnote3app://note/FE0D6114-DE87-4723-B4A1-BFBD27D5330D)

#### 停用词去除 [>>](marginnote3app://note/2C1DB9DF-E74D-47BF-A40C-2D003BAF4F27)

##### 停用词筛选方法 [>>](marginnote3app://note/ACDD1159-A5A7-44E0-9A0F-C86F78269A40) 常用的停用词筛选方法有词频法、文档频数法、熵值法、联合熵值法等。此外，也有学者提出了能够对文本进行筛选，自动提取文本自身的停用词的方法[10]。

#### 同义词去除 [>>](marginnote3app://note/FDB6AA04-6BFB-4909-9810-07FB7CFCA729)

##### 利用最新的 word2vec 的技术，可以叫方便地根据前后语境信息对当前特征词进行语义上的分类， 其精度对于在多义词的多个语义选项进行选择也绰绰有余。 [>>](marginnote3app://note/F3EA70F6-C683-410C-A506-3A17BFF6C4BA)

##### 进行同义词归并，除了需要准备完备的同义词表之外，还需要利用如 Wikipedia 这样的大型语料库训练 word2vec 模型用于对多义同义词归并进行确定。 [>>](marginnote3app://note/FD01635F-E184-4687-AE44-A043504ACD22)

### 文本建模以及相似度度量 [>>](marginnote3app://note/6019960C-19A1-4C14-A1EF-6A908A21A6A5)

#### 这就需要建立文本表示模型。通常使用的文本表示模型一般可以分为三种类型：向量空间模型（VSM，Vector Space Model）、概率模型、 布尔逻辑模型。 [>>](marginnote3app://note/8DD231BB-E4A1-4C41-B9E4-DB47837D7DEC)

#### VSM 向量空间模型的核心思 [>>](marginnote3app://note/1E59D7F1-CAC5-4191-997C-296F10A6AF89) VSM 向量空间模型的核心思想在于将每个文档都映射到向量空间中的一个高维向量中，根据向量间的余弦定理计算文档之间的相似度。

##### 每个文档映射而成的向量使用文档中出现的特征词以及词的权重进行表示。 [>>](marginnote3app://note/1685013D-8E29-4215-A691-FFB8D797D23F)

##### 特征项加权策略TF-IDF （词频-逆向文件频率） [>>](marginnote3app://note/FF09886E-8BE5-499C-82BF-7CD8ADA98DC4) 特征项加权策略是 TF-IDF （Term Frequency-Invert Document Frequency，词频-逆向文件频率）技术。TF 指的是该特征词在该文档中出现的次数，IDF 则指的是该特征词在整个文档集合中出现的次数。

###### TF-IDF 方法的一个基本假设 [>>](marginnote3app://note/90AC05F3-AF99-4936-AF8A-65F343DD5DA6) TF-IDF 方法的一个基本假设是， 若在文档中某个特征词出现的次数越多， 相对应的该特征词词在其他文档中出现的次数越深， 则该词在文档中的相关性权值就越高； 若在文档中某个特征词出现的次数较少， 该词在其他文档中出现的次数越多， 则该词在文档中的相关性权值就越低。

###### [>>](marginnote3app://note/CD56798B-EE01-45F6-B56D-EEF5F65351A5)

###### TF-IDF 方法的优点 [>>](marginnote3app://note/4F4464D0-3E11-424E-8C6F-2349593D9080) TF-IDF 方法的优点在于有效地抑制了文档中一些常见但无意义的高频特征词对文档语义信息的影响。

##### 余弦相似度 [>>](marginnote3app://note/3B4A41F6-0C82-435E-920D-B0D8905EF6AD) 将文档映射成为向量后，我们通常会使用余弦定理来计算文档之间的语义相似度

###### 除了可以使用余弦值表示相似度之外，文档间的相似度还可以使用明可夫斯基距离、KL 距离（Kullback-Leibler Divergence）等方法进行计算。 [>>](marginnote3app://note/B1D5CADA-7913-4597-8F3E-39A9AB505844)

### 文本分类算法 [>>](marginnote3app://note/93C4C6EC-CA7A-4935-A99B-CA46F512A5E7)

#### 文本分类算法按学习过程的特点， 可以分为有监督分类和半监督分类两类[12]。 [>>](marginnote3app://note/B7A39D29-A2C4-40BF-AAE9-1741FEFDFF95)

##### 有监督分类 [>>](marginnote3app://note/7DD4C5EF-4320-4A33-B096-C3D032577E3E) 通过某种学习算法寻找分类函数的过程就是一种有监督的学习过程。 当前大多数文本分类算法都是有监督的， 如朴素贝叶斯 （NB） 、 BP 神经网络、支持向量机（SVM）等。

##### 半监督分类 [>>](marginnote3app://note/9DE85ECB-D8C9-4F1F-84AA-1BB30E45BB29) 半监督分类，即采用半监督学习算法，利用少量的已标定类别的文档和大量的需要分类的文档进行训练和分类。 目前， 主要的半监督分类方法有协同训练类型、基于图的半监督分类[13]等。

#### 语料库 [>>](marginnote3app://note/EECF60C7-9414-4AC3-B07E-6A5B4B726862) 文本分类中一个重要的概念就是“语料库”，它又分为训练集和测试集。

#### 朴素贝叶斯（NB）算法 [>>](marginnote3app://note/6AD93335-CCB7-44B1-BD37-15984E84C278)

##### 贝叶斯算法关注的是文档属于某类别概率。文档属于某个类别的概率等于文档中每个词属于该类别的概率的综合表达式。 而每个词属于该类别的概率又在一定程度上可以用这个词在该类别训练文档中出现的次数 （词频信息） 来粗略估计，因而使得整个计算过程成为可行的。 使用朴素贝叶斯算法时， 在训练阶段的主要任务就是估计这些值。 [>>](marginnote3app://note/4CB80DA5-CC5F-42F9-9CB3-8D065E324DBD)

#### 支持向量机（SVM）算法 [>>](marginnote3app://note/94BB44B6-096F-4A4E-9C08-BEB8F2562923)

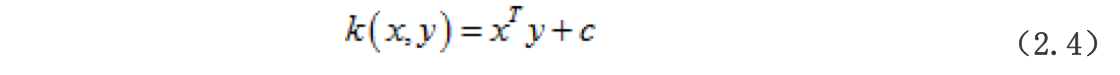
##### SVM 训练 [>>](marginnote3app://note/4630D7E2-6DDD-4EA9-82B4-13F9847EEC71) SVM 训练的本质是解决一个二次规划问题（Quadruple Programming， 指目标函数为二次函数， 约束条件为线性约束的最优化问题），得到的是全局最优解，这使它有着其他统计学习技术难以比拟的优越性。

##### 线性可以分类的二分类问题 [>>](marginnote3app://note/E9F06702-A392-4A85-9584-B1DD0CCE622F) 针对线性可以分类的二分类问题，SVM 通过寻找支持向量，减少运算复杂度，在两个类别的样本集之间寻找一个最优分界面, 将两类分开, 并使超平面离数据点的“间隔”最大。

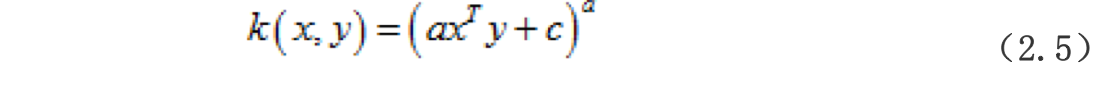
##### 线性不可分类的二分类问题 [>>](marginnote3app://note/A5D3661F-3241-4D73-A826-FA7A101B06E1) 针对线性不可分类的二分类问题， SVM 通过利用核函数 （Kernels） 将非线性可分的特征向量空间映射到线性可分的特征向量空间， 简化映射空间中的内积运算，避开了直接在高维空间中进行计算，通过将数据映射到高维空间，然后再利用线性可分的支持向量机进行分类， 以此来解决在原始空间中线性不可分的问题。

##### 常用到的核函数主要有 [>>](marginnote3app://note/8AB35946-4C81-4205-BD80-303A51F955D5)

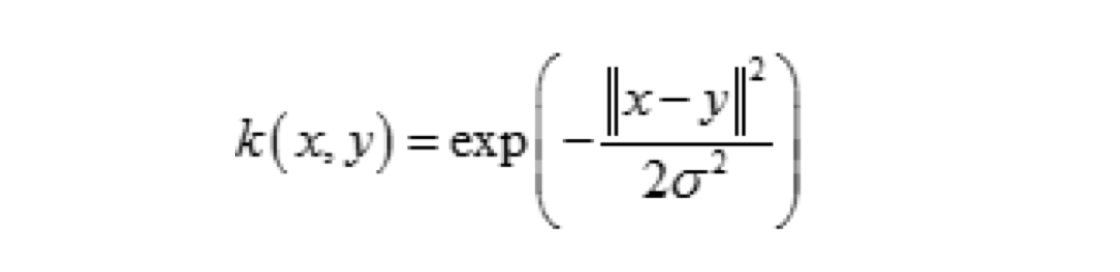
###### （1）线性核（Linear Kernel） [>>](marginnote3app://note/2A161B64-5E18-4464-886B-CF8B703905E2)

 [>>](marginnote3app://note/9F03C583-5712-4061-B714-80867348EFDA)

###### （2）多项式核（Polynomial Kernel） [>>](marginnote3app://note/576EC3D5-3C76-498D-BF61-F4A1F1484442)

 [>>](marginnote3app://note/0D5598E3-07FF-4336-BB7A-C2BD2859E937)

###### （3）径向基核函数 （Radial Basis Function） ， 也称为高斯核 （Gaussian Kernel） [>>](marginnote3app://note/4792CD0B-BA8B-443F-BD6F-5B69F5682E7E)

 [>>](marginnote3app://note/40456798-C0EE-4B6F-B0E6-622807E91FF9)

##### SVM 方法的缺点 [>>](marginnote3app://note/B059D989-E26A-420E-B607-3689ACBE5E12) SVM 方法也有其固有的缺点，首先是核函数的选择缺乏系统的指导方法，难以针对具体问题选择最佳的核函数和相应的参数；另一方面，SVM 方法的训练速度极大地受到训练集规模的影响，其计算开销比较大，。

##### SVM 分类器的优点 [>>](marginnote3app://note/64DFF00E-4C16-4649-B29C-48758BE04055) SVM 分类器的优点在于通用性较好， 且分类精度高、 分类速度与训练样本个数无关，在查准率和查全率两方面都优于 KNN 及朴素贝叶斯方法。

#### K 近邻（KNN）算法 [>>](marginnote3app://note/8FC18CC6-9C70-473D-AD1A-2B57D40D8700)

##### K 近邻（K Nearest Neighbor，KNN）算法 [>>](marginnote3app://note/635947FE-89C6-4C43-BC3D-13B62401452A)

K 近邻（K Nearest Neighbor，KNN）算法是一种基于向量空间的文本分类算法，因此，KNN 算法同样要找到这样一个合适的分类边界，只不过该算法是通过局部信息来确定其类别边界的。KNN 是一种基于类比的分类方法，通过计算  
各已知类别的文本与要确定类别的文本之间的距离， 哪篇文本与它距离最近， 就将其类别标号赋给它[18]。

###### 距离用的是余弦相似度 [>>](marginnote3app://note/EA554F17-9936-4E19-9974-43F135797C8C)

###### [>>](marginnote3app://note/FF4DB1F7-836A-473C-9CFE-A206CBCAF24A)

### 文本聚类算法 [>>](marginnote3app://note/AD7E8036-6137-47AF-B340-A66A98979A25)

#### 基于划分的聚类方法 [>>](marginnote3app://note/285C778C-B87B-417F-8D93-3B686E3BED16)

#### 基于层次的聚类方法 [>>](marginnote3app://note/9800043E-85E4-44C4-9D7A-08279C3BDE29)

#### 基于密度的聚类方法 [>>](marginnote3app://note/C0BD6B03-383E-4E36-B4A6-F83FD81A852A)

#### 基于网格的聚类方法 [>>](marginnote3app://note/281C49E4-F203-4724-9575-098B7E90D713)

#### 基于模型的聚类方法 [>>](marginnote3app://note/43D460A7-BEE5-48C5-8EF3-8E6C04B4D6F6)

### 主题模型 LDA 介绍 [>>](marginnote3app://note/381ACB6C-23B9-4FC0-A42C-A7BF4A2B9A6F)

#difficult #important

#### LDA 模型的核心思想是将每个文档都看做为一个主题向量， 所有的文档都共享潜在的主题集合，这些主题由一系列特征词所组成。LDA 模型是一个三层的贝叶斯网络结构，分为文档层，主题层，特征词层，每一层都有相应的随机变量或参数进行控制。 [>>](marginnote3app://note/A2DF6ADB-DBCE-4A0D-A82B-32D698799F36)

#### LDA 主题模型图 [>>](marginnote3app://note/97E9E52D-4E05-4254-9617-EB24E81E0210)

##### 所以， 对于一篇文档 d 中的每一个单词， LDA 模型首先根据先验知识α确定某篇文档的主题分布θ，然后从该文档所对应的多项分布（主题分布）θ中抽取一个主题 z，接着根据先验知识β确定当前主题的词语分布Φ，然后从主题 z 所对应的多项分布 （词分布） Φ中抽取一个单词 w。 然后将这个过程重复 N 次， 就产生了文档 d。 [>>](marginnote3app://note/ED23030B-59A2-4007-AC1A-4389080DB526)

##### 综上，M 篇文档会对应于 M 个独立的 Dirichlet-Multinomial 共轭结构，K 个topic 会对应于 K 个独立的 Dirichlet-Multinomial 共轭结构。 由此可以得到整个模型中所有变量的联合分布，相关公式如下所示： [>>](marginnote3app://note/E78329AB-E374-4D9A-AE38-33F8CD0CE487)

##### [>>](marginnote3app://note/461EC4A3-BE8B-4EA0-9254-E2E9A304C11D)

#### 狄利克雷分布 [>>](marginnote3app://note/4529CEB3-0BA5-4151-8F10-8CEE961ED32F)

##### 狄利克雷分布（Dirichlet Allocation）是多项分布的共轭先验概率分布[33]。 所谓多项分布， 是指在多次试验， 单次试验的随机变量为多个离散值的分布。 一个最简单的例子为多次投掷六面骰子的试验， N 次试验结果服从 K=6 的多项分布。 [>>](marginnote3app://note/94B99CB9-FAAB-4AC9-A390-06A45F5CF242)

#### 吉布斯采样 [>>](marginnote3app://note/9C18E784-A3B1-445B-9A94-083D53715AFF)

##### LDA 主题模型的参数通常采用吉布斯 （Gibbs） 采样方法进行估计， Gibbs 采样方法是 MCMC 的一种较为简单的实现方式。 MCMC （Markov Chain Monte Carlo，马尔科夫链蒙特卡洛方法） 是一种近似迭代样本值的方法， 该方法允许马尔科夫链在经过不断的迭代后收敛到目标的概率分布下，迭代过程遵守一定的规则。 [>>](marginnote3app://note/D0FDE704-D1E2-4BF2-8560-BB822AF75394)

#### LDA 生成模型 [>>](marginnote3app://note/240EB972-72C4-455E-AECD-44141A34F079)

### 基于 LDA 模型的文本相似度度量 [>>](marginnote3app://note/4AFBB888-5393-4292-869C-915161044606)

#difficult #important

#### [>>](marginnote3app://note/BD6A49CD-3911-4729-99D6-B37D6A9045F9)

由于文本的主题分布是文本向量空间的简单映射，因此在文本-主题模型的映射情况下，计算两个文本在LDA模型下的相似度可以通过计算与之  
对应的主题概率分布来实现[35]。因为文档-主题的映射是文档在混合主题下的概率空间分布，因此可以使用 KL 距离的改进版本 JS 距离进行计算。JS 距离比欧氏距离等其他距离度量方式更适用于概率空间， 它从信息的相对熵角度度量文本之间的距离。

#### [>>](marginnote3app://note/8318A65A-E3FD-4D29-B5E8-D8D5FCD9C690)

#### [>>](marginnote3app://note/C84160B9-D74D-4417-BA3D-9759D5DF08F5)

### 主题聚类数量的选择 [>>](marginnote3app://note/D5F182D8-7C67-45C6-9AFE-4EB63133715E)

在 LDA 主题模型中，主题数 T 的选择需要人工指定，而 T 的取值会直接影响到模型拟合文档集的质量。  
#difficult #important

#### 在 LDA 主题模型中，调节主题数量主要可以参考两种方式，其一是贝叶斯统计标准方法[36]；其二是困惑度，也就是 perplexity。 [>>](marginnote3app://note/A297E7CD-0733-42C7-BEF8-9F9DAFE194C2)

#### 贝叶斯统计标准 [>>](marginnote3app://note/2BF11EBE-7B10-4528-A567-6A25F5C9C713)

#### 困惑度（perplexity） [>>](marginnote3app://note/FC4B02BB-98B5-46DE-B803-A348EC6A974D)

### 聚类效果的评价 [>>](marginnote3app://note/808D55BC-EE65-4E1E-A684-C664472A3C49)

#### 基于人工标注的指标 [>>](marginnote3app://note/4188529B-ABCB-4D1C-8392-196C786E0B96)

##### 目前被广泛使用的基于人工标注的方法是 F-measure 值评价方法[30]。该值包含了两个度量值：准确率（precision）和召回率（recall）。 [>>](marginnote3app://note/50E74F01-13FA-414D-B5B4-BDCCA0BDF8E2)

#### 基于目标函数的指标 [>>](marginnote3app://note/85042D96-FB25-4FC1-9EBD-216C2B08F53F)

##### 如文献[32]中评价聚类结果与使用聚类结果进行分类得到的分类结果的吻合度。 [>>](marginnote3app://note/9441D4E0-4EFC-46A0-9C55-317B0BFA7ED4)

##### 聚类有效性函数作为评价无监督学习算法结果的一种途径，若单纯使用， 则无法建立与最终时间聚类结果准确度等硬性指标的关系。 聚类有效性函数往往通过聚类结果内聚性， 类间距等指标进行评价， 更偏重于从聚类方法内部横向对聚类结果进行比较， 因而在接下来讨论的确定最佳聚类数量的问题上，我们将着重研究聚类有效性函数在其中的作用。 [>>](marginnote3app://note/63AE6010-6D8D-4D0F-8400-3AD7E48E87F9)

### 分类效果的评价 [>>](marginnote3app://note/ABE365DF-D2B2-4668-B3AE-337B05E54A5D)

#### 对于文档进行分类的效果的评价，其指标与聚类算法基于人工标注进行评价的指标是一致的，即准确率、召回率、F-measure、正确率等。 [>>](marginnote3app://note/D239FF2A-2A94-4B7D-B221-34AEDA5ED840)

## 3 算法的需求分析 [>>](marginnote3app://note/E3FBCB85-FF65-4FE8-8B1D-47BDA1DF1D6B)

### 需求分析 [>>](marginnote3app://note/338563BA-1111-475D-882B-C4A1B04B4F74)

#### 系统的使用场景 [>>](marginnote3app://note/E14C9FE3-08A7-4089-A8B4-A61158D2AA4B) 本系统的使用场景主要为： 用户将通过数据收集得到的文本集输入系统，设置系统参数，如最后聚类数目、聚类文本选取规则等等，获取中间文本模型输出，系统读取文本模型选取部分文档进行聚类，最后输出聚类结果，包括各个主题以及主题的描述词、 各文档划分到的主题等， 用户可能还需要根据最后生成的结果进行聚类效果评价。

### 算法的功能性需求分析 [>>](marginnote3app://note/862EE36F-DE35-4FC1-AEB7-475A6C680289)

### 算法的非功能性需求 [>>](marginnote3app://note/9FC40EFF-9F4B-46A6-85C2-C46DEC58D437)

## 4 算法的设计与实现 [>>](marginnote3app://note/E9982D5A-41D3-4E57-ADD3-442A60C78AD8)

#code

### 算法模块设计图 [>>](marginnote3app://note/C92583FD-23FB-4C91-9EA9-36508BC927EF)

### 开发环境 [>>](marginnote3app://note/17BA4A11-0A18-40C6-9930-91222116C4E4)

#### 安装 linux-Ubuntu 操作系统， 使用 python 作为开发语言， 其中将应用到 python 中的 gensim、 sklearn、scipy 等开源的类库。 [>>](marginnote3app://note/4930D785-C60F-4A01-A337-28C20F216D60)

### 算法各模块间数据传输形式的设计 [>>](marginnote3app://note/B696A95F-1890-42C2-99B8-9AC68024BBAC)

### 算法各模块的设计与开发 [>>](marginnote3app://note/394849F2-40DD-42F8-A69A-B05561DFE60A)

#### 文本预处理模块 [>>](marginnote3app://note/82836087-9DD7-4B72-92A7-24198E7C9565)

##### （1）中文文本分词 [>>](marginnote3app://note/EA011067-96C2-4C35-9FB6-05F27B04289C)

###### 将文档按照词为单位进行分割，这里将使用 python 下的库包 jieba 分词进行分词。 [>>](marginnote3app://note/7EDEEC80-9E78-4A4E-9E13-E8460610FA0A)

##### （2）停用词去除 [>>](marginnote3app://note/774EDB5D-CF19-4DAE-A38C-272334453AB5)

###### 在分词后的文档集中将停用词表上的单词去除掉，我们在这里使用的是哈工大停用词表。 [>>](marginnote3app://note/08CA29AB-02B4-4FDF-9B5D-ABE166F76E05)

##### （3）关键词选取 [>>](marginnote3app://note/10625A29-068C-43D4-AF17-69144EE785FF)

###### 系统将使用 TF-IDF 算法计算每个文档单词的权重 （其中 TF-IDF 算法的原理上文已给出），根据单词的 TF-IDF 值选取最高的一部分的单词作为文档的关键词。 [>>](marginnote3app://note/9624A428-1686-4DB7-9202-CB720DC90C4E)

###### 综上所述， 我们在本文中选取每个文档中 TF-IDF 值最高的 50 个单词作为文档的关键词保留下来。如果文档不足 50 个单词，将选取文档的全部单词作为文档的关键词。 [>>](marginnote3app://note/E725E526-5812-443C-BCC2-25ABD3B3DEF5)

#### 垃圾信息过滤模块 [>>](marginnote3app://note/70071C08-9671-45A9-9AD0-3E59105FD77D)

##### 根据标定得到的文档集，将其划分为训练集和测试集，通过训练集对分类器进行训练后， 对测试集进行分类， 并根据分类结果与测试集的标定信息计算查全率、准确率、F-measure 等统计标准，根据这些标准作为改进分类器和参数的参照。 在选定合适的分类算法和参数后， 将整个标定的文档集作为训练集训练分类器后，对剩余所有未分类的文档进行分类。 [>>](marginnote3app://note/986825F7-993D-4B66-A3ED-C21BFA92EFD2)

##### 在本模块中， 我们选择使用 sklearn 包下的 SVM 支持向量机作为进行文档的分类，对于文档这种典型的高维数据，SVM 中最适合使用 RBF 径向基核函数。 [>>](marginnote3app://note/E2B8B70C-0D8B-492A-BD3B-06C349AA4DBD)

#### LDA 模型模块 [>>](marginnote3app://note/3FCE2CFB-A93A-4205-A927-4BCEA769DB08)

##### 本模块使用 python 下的库包 gensim 进行主题模型的搭建。在完成 LDA 主题模型的搭建后，将该模型以文件的形式保存下来并将文件名称返回。 [>>](marginnote3app://note/DB7CAF26-7771-4D6E-A298-3B26A99A0182)

##### 我们将对选定的有效性函数值，包括 LDA 模型的困惑度等目标函数与实际应用效果 （以人工判断和人工标注指标进行衡量） 之间的关系进行探讨，以作为应对不同实际需求是选择相应主题数量的判断依据。 [>>](marginnote3app://note/890A7074-4B94-4ED6-8365-3D6DCCFFC7C3)

#### VSM 模型模块 [>>](marginnote3app://note/0F0195F1-31B8-4C62-88ED-852767FA0F98)

##### 本模块将使用 VSM 模型对文档进行计算，使用 TF-IDF 方法对每个文档的单词进行加权，将文档转化为多维向量空间上的多维向量。本模块使用 python 下的库包 gensim 进行 VSM 模型的搭建。 [>>](marginnote3app://note/CD49EC29-2854-4429-9449-88A387750DBE)

#### 文本模型转换模块 [>>](marginnote3app://note/3C67BFD8-4E7C-4BCB-AAFC-C86C29DBA87A)

##### 考虑到文本集有可能非常巨大， 而且 LDA 模型和 VSM 模型使用同一个类库gensim， 因此将增加一个文本模型转换模块， 加入中间输出 tfidf 模型文档， LDA 模型文档， 在需要读取时再逐行读取模型文件， 最后输出的时候去除文档原型的关于两个模型的多维向量，存储文档的序号。 [>>](marginnote3app://note/9391B5B4-505A-418A-A347-E95210ACF08F)

#### 层次聚类模块 [>>](marginnote3app://note/0E40BE8C-8D67-407C-8679-AEDB30530742)

##### 根据两种向量分别计算相似度，使用加权值对两个相似度进行整合计算，其中 VSM 向量使用余弦相似度计算向量的相似度，LDA 向量使用 JS 距离的方法计算向量的相似度； [>>](marginnote3app://note/82F47CAB-8B24-4473-AB0D-F1570DBDC9B9)

##### [>>](marginnote3app://note/5C8B562B-1CE9-4085-A640-6042DE2C44F8)

##### 本模块使用凝聚法，即自底向上的策略，将每个文档作为一个类簇，然后不断合并相似度最小的类簇，最后直到剩下的类簇数量为一开始设定的类簇数量。 [>>](marginnote3app://note/3B5F1C16-055F-4511-86CA-E22ADCDC1874)

### 算法实现效果测试与改进 [>>](marginnote3app://note/C3E4C2F3-F832-4FF5-B1D0-4646C4B667B5)

### 实现过程中的一些问题 [>>](marginnote3app://note/06DFB204-E727-46BE-8658-9E08BB432441)

## 5 总结与展望 [>>](marginnote3app://note/84275AEE-69F1-43C7-89F9-143203079C52)

### 总结 [>>](marginnote3app://note/27D22B30-2200-4659-B3A2-FFD54078BEB0)

#### 文本聚类中文本相似度的计算方式将直接影响聚类的质量好坏。因此本文使用 LDA 主题模型挖掘文本自身蕴含的潜在主题信息， 通过对文本信息的深挖与降维， 提高文本相似度计算的深度，同时利用 LDA 主题模型获取文档-主题-描述词的三层概率分布，识别最后类簇的主题。 [>>](marginnote3app://note/0B3D49A2-0871-487A-82F1-C9914479EEA1)

#### 本文将 LDA 主题模型和 VSM 空间向量模型结合，利用 LDA 模型生成的文档-主题概率分布模型和 TF-IDF 特征词空间向量共同计算文档相似度并利用实验确定了使用与本文实验文本的最佳参数λ， 最后运用层次聚类算法对文档进行主题的计算。 [>>](marginnote3app://note/BB8A2324-C9C8-480C-A7A7-9D0C16C1A3D5)