基于词项聚类的文本语义标签抽取研究

李 雄 丁治明 苏 醒 郭黎敏
(北京工业大学信息学部 北京 100124)

摘 要 本研究主要解决在大量文本数据中抽取关键语义信息的问题。文本是自然语言的信息载体,在分析和处理文本信息时,由于目标与方式不同,对文本信息的特征表达方式也各不相同。已有的语义抽取方法往往是针对单篇文本的,忽略了不同文本间的语义联系。为此,文中提出了基于词项聚类的文本语义标签提取方法。该方法以语义抽取为目标,以 Hinton 的分布式表示假说为文本信息的表达方式,并以最大化语义标签与原文本数据间的语义相似度为目标,使用聚类算法对语义标签进行聚类。实验表明,所提方法由于是基于全体词汇表对语义信息分布进行聚类计算的,因此在语义丰富度和表达能力上相比很多现有方法具有更好的表现。

关键词 语义抽取,分布式表示假说,聚类,相似度
中图法分类号 TP391 **文献标识码** A

Word Clustering Based Text Semantic Tagging Extraction Method

LI Xiong DING Zhi-ming SU Xing GUO Li-min
(Department of Information, Beijing University of Technology, Beijing 100124, China)

Abstract This research mainly solves the problem of extracting key semantic information from a large number of text data. Text is the information carrier of the natural language. When the text information is analyzed and processed, the characteristics of text messages are different, due to different goals and methods. In the past, the semantic tagging extraction method is usually focused on the single text, but the semantic relationships between different texts are ignored. To this end, this paper proposed a text semantic tagging extraction method based on word clustering. The proposed method is based on semantic tagging extraction processing target, which employs a distributed Hinton representation hypothesis to express text information, and uses word clustering algorithm to maximize the semantic tagging and semantic similarity between the original text data. Experiments show that since the method involves all vocabularies in the cluster computing, the semantic richness and power of information expression of the proposed method outperform many existing methods.

Keywords Semantic extraction, Distributed representation hypothesis, Clustering, Similarity

1 引言

随着互联网的快速发展以及大数据时代的到来,对各类由自然语言组成的文本的信息采集和存储的需求越来越强烈。因此,对大量蕴含巨大潜在价值的文本数据进行语义分析和计算显得尤为重要。语义标签作为文本数据的核心语义体现以及主题信息的浓缩,不仅有助于人们快速了解一篇文本的主要内容,也有助于人们在检索大量文本数据时快速定位所需信息。另外,语义标签也可作为文本分类、情感分析等多种自然语言处理任务的输入特征,以提高任务处理的效率。因此,语义标签的提取不仅是自然语言处理领域一个非常重要的问题,也是该领域的基础核心问题。语义标签提取技术在信息检索[1]、话题跟踪[2]、自动摘要[3]、文本聚类[4]等领域都有着广泛的应用。

目前,在对文本进行语义标签自动抽取的任务中,有两个主要的难点。1)文本信息的可计算表示形式。由于在文本中信息主要由自然语言形式表示,每个词项是信息的表示单元,但是词项这种表达形式却无法直接代入数学模型中进行语义计算,因此寻找对应文本的一种可计算表示形式是语义标签抽取任务中非常重要和基本的问题。2)在大量无标注环境下进行语义计算。在互联网包含的大量文本数据中,带有语义标签标注信息的文本只占极少一部分,而手工标注文本的语义标签的方式非常耗费人力成本,因此在语义标签提取任务中,更多是在非监督环境下进行文本的语义计算,这给语义计算的可行性和精确性都提出了相当大的挑战。

国内外已经在语义标签提取问题上进行了大量的研究[5-9]。

程伟鹤等[10]基于统计的思想,将文本中具有高度共现关系的高频度字串作为文本的语义标签,从而将原问题转化为最长公共子串匹配问题。在计算过程中,该方法使用字串(词项)频率作为词项的语义表示,最后依靠权重排序的方式提取出文本语义标签。王立霞等[11]则使用图结构表示文本词项

本文受国家自然科学基金(91546111)资助。

李 雄(1992—),男,硕士生,主要研究方向为自然语言处理、数据挖掘;丁治明(1966—),男,博士,教授,主要研究方向为感知大数据、数据挖掘,E-mail:zmding@bjut.edu.cn(通信作者)。

间的语义关系,并定义了词项居间密度作为词项的语义表示,同时提出了基于图迭代计算的居间密度计算方法,再通过居间密度值的排序提取出文本的语义标签。李鹏等[12]使用带tag标注的网页数据,将相似的文本信息建立在同一个语义网络中,以图节点权重表示词项的语义,并使用PageRank算法对图节点的权重进行计算,最后依据节点权重提取语义标签。基于统计思想的算法普遍存在的问题是,词项的特征信息没有被充分表示,损失了较多的语义信息,从而限制了算法效率;而基于图迭代思想的算法普遍存在的问题是,通常会依赖更多外部信息,这限制了这类算法的适用范围。

为了改进现有方法在关键词提取任务中存在的问题,我们在总结了现有语义标签提取算法的基础上,针对语义标签提取任务中的两个主要难点,提出了新的语义标签提取算法。该算法在语义标签提取任务上具有以下优点:

1)文本的表示形式具有更加充足的信息容量。本算法使用词项的分布式词向量来对文本中的语义信息进行可计算的表示,这种表示形式采用低维向量作为词项的语义表示形式,相比目前算法中常用的统计表示形式和图表示形式,不仅具有更加丰富的语义信息,而且使得数学模型的计算更加方便。

2)在文本数据中做语义计算不依赖于其他的语义标注信息,模型更加简洁、易用。该算法在计算语义时使用了聚类算法,不依赖于其他任何标注数据,使得该算法的应用场景更加广泛。

本文第2节介绍了相关的语义标签提取算法并分析了它们的优缺点;第3节给出语义标签提取问题中模型概念的定义以及模型的优化目标;第4节详细阐述了本文的语义标签提取的算法模型;第5节给出了我们对语义标签提取算法模型的实验评估以及相应的结果分析;最后总结全文,并对未来工作进行展望。

2 相关工作

文本是人思想的高层次抽象表达,形式丰富、语义复杂。对计算机而言,纯粹的文本是不可计算的,因此将文本转化成计算机可处理的数学表示形式,是计算机处理文本的基础。根据文本的表示形式,目前常用的语义标签抽取技术主要可分为两类:基于统计表示形式的权重排序算法和基于图表示形式的节点加权排序算法。下面将从基于统计表示形式和基于图表示形式两个方面来介绍和分析现有的语义标签提取方法。

2.1 基于统计表示形式的权重排序算法

基于统计表示形式的权重排序算法的主要思想是:根据统计学,使用指定的词项统计指标来衡量词项对于文本的语义贡献程度,并由此衍生出一系列文本统计领域特有的语义计量指标,最后通过权重排序的方式对词项语义进行计算,筛选出能代表文本核心语义的词项。基于这种思想,国内外学者做了大量的研究工作[10,13-16]。

程伟等鹤[10]基于统计的思想,将文本中具有高度共现关系的高频度字串作为文本的语义标签,从而将原问题转化为最长公共子串匹配问题。在计算过程中,该方法使用字串(词项)频率作为词项的语义表示,最后依靠权重排序的方式提取出文本语义标签。该算法的优点是空间复杂度较低,这对在大量文本数据的环境下做语义计算非常有利;其不足之处在于,单以共现频率作为词项的语义表示,会损失较多的语义信息,致使语义计算,结果不够理想。

罗燕等[13]结合齐普夫定律和词频统计方法,将TF-IDF作为词项的语义表示,并通过齐普夫定律计算出有效的文本长度阈值,对较短文本进行过滤,只对长文本进行语义计算,以提高效率,最后通过权重排序的方式提取出文本的语义标签。该算法的优点是文本语义表示较充分,计算精度较高;其缺点在于提取短文本语义标签时的效率有所不足,其应用范围较狭窄。

李晓超等[14]利用齐普夫定律推导出了适合中文文本的同频词数的数学表达式,能更准确地表示出不同长度的文本中各频次的同频词数;并依据同频词统计规律,对参与TF-IDF计算的词汇进行过滤,在不丢失语义标签信息的情况下,提高了TF-IDF算法的计算效率。该方法的优点在于通过预先过滤减轻了计算负担,提高了计算效率;不足之处在于,在计算高频次同频词时计算结果不够准确,缺乏稳定性。

2.2 基于图表示形式的节点加权排序算法

基于图表示形式的节点加权排序算法的主要思想是:将文本中的词项当作图的节点,将词项之间的共现关系和序列关系当作节点之间的连接边,从而以图架构作为文本的语义表示,同时为每个节点赋以权重作为词项的语义表示,通过不同的图迭代算法计算词项的语义权重,再由权重排序算法筛选文本的语义标签。基于这种思想,国内外学者做了大量研究工作[11-12,17-19]。

王立霞等[11]使用图结构表示文本词项间的语义关系,并定义了词项居间密度作为词项的语义表示,同时提出了基于图迭代计算的居间密度计算方法。该算法通过建立表示文本语义结构特征的网络,迭代计算出合适的词项居间密度值,再通过居间密度值的排序筛选出文本的语义标签。该方法的优点在于计算的时间复杂度较低,而且迭代控制比较灵活,易于根据语料环境的变化做出相应的修改;不足之处在于,图结构只能表示单篇文本的语义信息,缺乏从全体语料上计算词项语义的过程,因此在词项的语义表示上会有一定的信息损失。

李鹏等[12]使用带tag标注的网页数据,根据网页的tag标注建立用户集合与标注集合,然后根据集合上的相似性对网页文本进行初次划分,将相似的文本信息建立在同一个语义网络中,并使用PageRank算法对图节点的权重进行计算,最后依据节点权重提取语义标签。该算法的优点在于,将语义相似的文本纳入同一个网络中计算,一定程度上加强了文本的语义表示能力;其不足之处在于需要依赖外部的tag标注信息,从而限制了该算法的使用场景。

夏天[17]基于TextRank的算法思想,构建了候选语义标签图,引入覆盖影响力、位置影响力和频度影响力来计算词语之间的影响力概率转移矩阵,通过迭代法实现候选关键词分值的计算,并挑选前 n 个作为语义标签抽取结果。他们通过实验论证了该算法在语义标签提取任务中相比传统TextRank算法和LDA算法的优势。该算法的优点在于计算结果具有很好的精度;其不足在于,选取每个影响力指标对应的参数值时比较困难,不具备较好的易用性。

3 问题描述和数学模型

本节首先将对应语义标签提取问题进行数学描述,然后提出该问题最终所需的优化目标。

3.1 问题描述

在语义标签提取任务中,有一份包含 n 篇文本的语料

$Corpus=\{T_1,T_2,\cdots,T_n\}$。在提取过程中,经过分词后每篇文本具有 $m_i(1\leqslant i\leqslant n)$个词项,因此有文本表示为 $T_i=\{w_1,w_2,\cdots,w_j,\cdots,w_{m_i}\}$,其中 T_i 表示语料中的第 i 篇文本,w_j 表示该文本中的第 j 个词项。最终语义标签提取的目标是针对每篇文本,提取其对应的 k 个语义标签的集合 $Keys_i=\{key_1,key_2,\cdots,key_k\}$,其中 $Keys_i$ 表示第 i 篇文本的语义标签的集合,且 $k\ll m_i$。

由于文本中每个词项都是以自然语言的形式存在,无法纳入数学模型中直接进行计算,因此在语义标签提取过程中需要对文本数据进行数学表示,再将数学表示输入模型中进行计算,计算后的结果也还需要经过一次转化后才能成为人类可直接辨识的语义标签。基于此,我们给出语义标签提取任务的定义。

定义 1　词项自然形式与数学形式间的关系可用映射函数表示:

$$f(w_i,Dict)=rep_i \tag{1}$$

其中,w_i 表示语料中的任意一个词项,$Dict$ 表示全体语料中所有出现词的字典,rep_i 表示词项 w_i 在模型中的数学表示形式。函数 $f(*)$则表示词项自然形式 w_i 与数学形式 rep_i 之间的映射关系,$f(*)$是单射函数。

定义 1 的推广:由定义 1 可知,一篇文本的映射表示关系有:

$$f(T_i,Dict)=f(\{rep_1,\cdots,rep_{m_i}\}) \tag{2}$$

其中,T_i 表示语料中的任意一篇文本,$\{rep_1,\cdots,rep_{m_i}\}$则表示该篇文本在模型中的数学表示。

定义 2　语料中涉及的所有词项的词典可表示为一个二元组集合:

$$Dict=\{(w_1,rep_1),\cdots,(w_r,rep_r)\} \tag{3}$$

其中,$Dict$ 为语料中涉及的所有词项的词典,以二元组的集合形式表示。该结构主要用于计算词项的自然形式与数学形式之间的映射。

定义 3　语义标签的自然形式与数学形式之间的关系可用映射函数表示:

$$f(Keys_i,Dict)=\{rep_1,\cdots,rep_k\} \tag{4}$$

$$f^{-1}(Reps_i,Dict)=\{key_1,\cdots,key_k\} \tag{5}$$

其中,$Keys_i$ 表示一篇文本的语义标签集合,$Reps_i$ 为该语义标签集合对应的数学表示。式(4)可由定义 1 的推广得到。函数 $f^{-1}(*)$为函数 $f(*)$的反函数,表示由词项的数学形式计算其自然形式。

3.2　解决方案和优化目标

根据 3.1 节中的定义,在语义标签提取中,其任务的主要目标是在文本 T_i 中借助语料集合 $Corpus$ 的语言信息,抽取出文本的语义标签集合 $Keys_i$,使得集合 $Keys_i$ 可以代表文本 T_i 的核心语义。因此,在语义分布上,$Keys_i$ 集合必须与文本 T_i 集合具有最大的语义相似性。以数学表示衡量两者的相似性,由式(2)可知,文本 T_i 的数学表示为 $Reps_i=\{rep_1,\cdots,rep_{m_i}\}$;由式(5)可知,语义标签集合的数据表示 $keys_i=\{key_1,\cdots,key_k\}$。因此,语义标签提取任务的优化目标即为:

$$\arg\max_{Reps_i}(similarity(Reps_i,Keys_i)) \tag{6}$$

$$Reps_i=\{rep_1,\cdots,rep_m\} \tag{7}$$

$$Keys_i=\{key_1,\cdots,key_k\} \tag{8}$$

其中,$similarity(*,*)$表示输入的两个参数之间的相似性,其值越大,相似度越高。

4　模型实现

根据优化目标与模型定义,语义标签提取算法的模型框架如图 1 所示。该模型框架将语义标签提取划分为两个步骤:1)将原始语料文本的自然语言形式映射为可计算的数学表示形式;2)根据文本的数学表示计算其语义中心的分布,得到文本的语义标签集合的数学表示形式,再根据自然语言形式与数学表示形式之间的映射函数,将语义标签集合的数学形式转化成由词项组成的语义标签集合,该集合即为算法的最终结果。

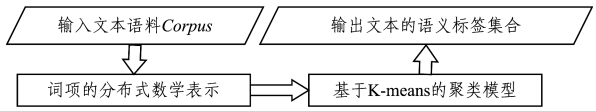

图 1　语义标签提取算法流程图

由图 1 的模型框架可知,在表示映射阶段,本算法基于词项的分布式假说,以低维向量作为词项的数学表示形式,并根据 n 元语言模型 CBOW 计算每个词项对应的向量表示;而在语义计算阶段,本算法基于 K-means 算法的思想计算文本的核心语义簇,由簇的簇心代表文本的语义标签的语义表示;最后在语义标签提取阶段,使用定义 3 中映射函数的反函数 $f^{-1}(*)$计算簇心的语义表示所对应的语义标签词项。

算法 1　语义标签提取算法

输入:语料 Corpus,参数 k
输出:全体语义标签集合 Res
1. for each T_i in Corpus:
2.　words←split_init(T_i)
3.　Dict.update(words)
4. Dict←CBOW(Corpus,Dict)
5. for each T_i in Corpus:
6.　$Reps_i$←f(T_i,Dict)
7.　Center←K-means($Reps_i$)
8.　$Keys_i$←f^{-1}(Center,Dict)
9.　Res.add($Keys_i$)
10. Return Res

对算法 1 解释如下:模型的输入为语料 $Corpus$ 和参数 k,其中语料 $Corpus$ 表示待计算的所有文本数据,参数 k 表示针对每篇文本需要抽取的语义标签个数,由用户手动指定;模型的输出为全体文本的语义标签集合 $Res=\{Keys_1,\cdots,Keys_n\}$,第 i 篇文本对应的语义标签为 $Keys_i$。为 $Corpus$ 构建字典 $Dict$,其中函数 $split_init(*)$表示对输入分词处理之后,为每个词项随机分配一个初始化向量,其返回值为一篇文本初始化后的二元组集合 $words=\{(w_1,rep_1),\cdots,(w_r,rep_r)\}$,$Dict$ 则表示全体文本初始化后的二元组集合。使用 CBOW 语言模型训练每个词项对应的向量,以此更新字典 $Dict$(第 1—4 行)。通过字典 $Dict$ 将文本 T_i 映射到向量空间中,得到其数学表示 $Reps_i$,因此 $Reps_i$ 为一组向量的集合(第 6 行)。在语义计算阶段,通过 K-means 算法计算出向量集合 $Reps_i$ 的簇心集合 $Center$,$Center$ 也是一组向量集合,但是其元素个数固定为 k 个(第 7 行)。在语义标签提取阶段,通过映射函数 $f^{-1}(*)$将语义标签集合的向量表示 $Center$ 转化为自然语言的词项表示 $Keys_i$,$Keys_i$ 是大小为 k 的词项的集合(第 8 行)。最后合并全部语料的计算结果,并将其返回(第 9—10 行)。

4.1 词项的分布式数学表示

在词项的数学表示形式上，本文依据分布式假说[20]的思想，采用长度为 50 维的向量表示每个词项的语义信息，其中单个词项所具有的多重抽象概念分别被表示为向量的不同维度值，而维度值则表示该词项在这种抽象概念下的语义强度。为了给每个词项计算出其对应的表示向量，本文使用 CBOW 语言模型为语料中出现的所有词项序列建模，训练语言模型的过程即是为每个词项寻找合适的向量表示的过程。

CBOW 语言模型属于 n 元语言模型的范畴，其核心思想是：根据文本中当前词项的上下文信息预测当前词项的概率分布，从而最大化预测结果为当前词的概率值，即可保证模型的可用性。将词项的向量表示作为模型的待定参数，在模型训练完成之后，每个词项的向量表示形式也完成计算。

CBOW 的模型结构如图 2 所示，主要分为输入层、映射层和输出层 3 层。图 2 为上下文宽度为 4 的 CBOW 模型：输入层输入的是词项 $W(t)$ 所处的上下文 $V(t-2)$，$V(t-1)$，$V(t+1)$，$V(t+2)$，其中 $V(t-2)$，$V(t-1)$，$V(t+1)$，$V(t+2)$ 分别为 $W(t)$ 所处的上下文词项 $W(t-2)$，$W(t-1)$，$W(t+1)$，$W(t+2)$ 对应的分布式表示向量；映射层用以合并输入层各项的值，计算各个维度上的和，生成映射层向量 $x_{w_i}=-V(t)+\sum_{i=t-2}^{t+2}V(i)$，且 $x_{w_i}\in R^d$。输出层根据映射层的值计算出输出 $W(t)$。另外，该模型在计算输出层时，使用了层次 softmax 结构进行优化，其主要思想是将原本扁平的全连接结构改写为一棵哈弗曼树，映射层与输出层只在树的非叶节点上连接，这样的优化减少了输出层的参数，加快了模型的计算速度。该算法的计算流程如算法 2 所示。

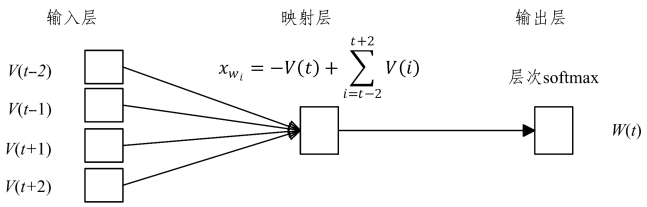

图 2 CBOW 的模型结构图

算法 2 词项的分布式表示计算

输入：语料 Corpus，向量维度 d

输出：词项的数学表示的映射字典 Dict

1. words_table←scan(Corpus)
2. (Dict,W)←init_dict(words_table,n)
3. for each T_i in Corpus:
4. for each tag in range(2,len(T_i)−2):
5. Window=[w_{tag-2},w_{tag-1},w_{tag+1},w_{tag+2}]
6. V(t−2),V(t−1),V(t+1),V(t+2)=f(Window,Dict)
7. $x_{w_i}=-V(t)+\sum_{i=t-2}^{t+2}V(i)$
8. $act_{w(t)}=g(x_{w(t)}\cdot W)$
9. w(t)=get_label(Window)
10. $\arg\min_{Dict}^{W}(loss(w(t),act_{w(t)}))$
11. return Dict

对算法 2 解释如下：模型输入 *Corpus* 和 d，其中 *Corpus* 表示待计算的所有文本数据，参数 d 表示词项向量的长度；输出为词项的数学表示的映射字典 *Dict*，其中存储了所有词项的自然表述到向量表示的映射关系，通常以矩阵形式表示。扫描初始化语料中出现的所有词项初始化词表 *words_table*，以此初始化每个词项的向量表示并存入 *Dict*，W 为语言模型中映射层到输出层间的连接权重，以矩阵形式表示（第 1−2 行）。在语言模型的前向计算阶段，*tag* 表示当前预测词所在的位置；*Window* 表示上下文宽度为 4 的窗口，它是一个包含 4 个连续词项的列表；$f(Window,Dict)$ 计算当前窗口的上下文环境，其中 $V(*)$ 为长度为 d 的向量，表示一个词项；$x_{w(t)}$ 为映射层的表示，由于只是求和映射，因此它是一个长度为 d 的向量；$act_{w(t)}$ 表示输出层接受的预测结果，由激活函数 $g(*)$ 计算得到，此处的激活函数为 sigmoid 函数（第 5−8 行）。在反向传播阶段，使用梯度下降算法调整 *Dict* 与 W 的值，使得输出层预测结果 $act_{w(t)}$ 与词项的实际结果 $w(t)$ 之间的损失最小化（第 9−10 行）。最后将映射字典 *Dict* 作为算法结果返回（第 11 行）。

4.2 基于 K-means 的聚类模型

通过对词项的分布式表示，文本中的每个词项都被映射为 d 维空间中的向量，因此文本的语义提取问题转化为：在当前向量集合中找出语义信息最强的有限个向量。这个问题可以使用聚类算法解决。聚类算法将数据集中的样本划分为若干个不相交的子集，每个子集被称作一个“簇”，通过这样的划分，每个“簇”都能对应到一些潜在的概念组合上，这些潜在的概念即是文本所期望提取的语义信息。

根据词项分布式特征表示方法，可得到每篇文本在 d 维空间中的向量集合表示，将此向量集合当作输入数据，可使用 K-means 算法对其聚类出文本中潜在信息分布的语义中心，最后由语义中心的向量表示反映射回词项的自然表示，从而得到文本的语义标签。但是，传统 K-means 算法以欧氏距离作为样本间相似度的度量，显然这在度量向量间的相似度时无效。考虑到训练后词向量在语义上拥有很好的向量平移性，例如“男人”−“女人”=“国王”−“王后”，因此使用向量间的余弦距离作为两个词项在语义上的相似性度量是更好的选择。改动 K-means 算法之后，算法以最大化样本与样本所属簇心的余弦距离为优化目标，采用贪心策略不断迭代其语义中心的位置，直到收敛。

算法 3 文本的语义聚类

输入：文本 T_i，字典 Dict，语义标签个数 k

输出：语义标签集合 $Keys_i$

1. $Reps_i$←f(T_i,Dict)
2. Center←init_center(k)
3. repeat:
4. for each rep_t in $Reps_i$:
5. for each $center_k$ in Center:
6. $similarity(rep_t,center_k)=\cos\langle rep_t,center_k\rangle=\frac{rep_t\cdot center_k}{\|rep_t\|\ \|center_k\|}$
7. $\gamma_t=\arg\max_{t\in\{1,2,\cdots,k\}}similarity(rep_t,center_k)$
8. $C_{\gamma_t}=C_{\gamma_t}\cup\{rep_t\}$
9. for each $center_k$ in Center:
10. $\mu_k=\frac{1}{|C_k|}\cdot\sum_{rep_t\in C_k}rep_t$
11. $center_k=\mu_k$
12. until convergent
13. $Keys_i$←f^{-1}(Center,Dict)
14. Return $Keys_i$

对算法 3 解释如下：算法输入 T_i，*Dict* 和 k，其中 T_i 表示一篇文本，*Dict* 是由算法 2 计算得到的词项的分布式表示映射字典，k 为需要提取出的语义标签个数；输出 $Keys_i$ 表示最

终提取的语义标签集合,由 k 个词项组成。首先进行参数的初始化,先初始化向量表示文本 T_i 中的每个词项,得到向量集合 $Reps_i$,对文本的语义中心初始化后得到簇心集合 $Center$(第 1－2 行)。然后计算簇心,先计算当前词项与每个中心的相似度,再依据相似度将当前词项划分到与其最相似的簇中心上,然后合并划分结果。接着重新计算簇中心向量。最后使用新的中心向量更新簇心集合 $Center$,进行多次迭代,当簇的划分不再变化时算法收敛(第 6－11 行)。其次,使用映射函数 $f^{-1}(*)$将语义标签集合的向量表示 $Center$ 转化为自然语言的词项表示 $Keys_i$,$Keys_i$ 为大小为 k 的词项的集合(第 13 行)。最后,将语义标签集合 $Keys_i$ 作为算法的计算结果返回(第 14 行)。

5　实验

本实验的目的是评估本算法在语义标签提取任务中的表现。实验数据采集自中国知网,共包含 2000 篇学术论文;从查全率、查准率、F1 值 3 个指标上评估算法的质量,并通过与传统 TF-IDF 算法和 TextRank 算法的对比来论证该算法的优点。

5.1　实验数据和结果评估指标

本实验的实验数据采集自中国知网,共包含 2000 篇学术论文,其中机器学习、大数据、量子计算和密码学领域的文章各 500 篇。表 1 列出实验数据的设置。

表 1　实验数据配置

(单位:篇)

数据总量	机器学习领域	大数据领域	量子计算领域	密码学领域
2000	500	500	500	500

我们使用查全率 R、查准率 P、平衡 F 分数 $F1$ 3 个指标来衡量算法的实验效果。3 种指标的数学定义如下:

$$R=\frac{S\cap E}{S} \tag{9}$$

$$P=\frac{S\cap E}{E} \tag{10}$$

$$F1=\frac{2*R*P}{R+P} \tag{11}$$

其中,S 表示作者标注的语义标签集;E 表示应用相关算法自动提取的语义标签集。查全率 R 表示所有正确的语义标签中被算法抽取到的百分比;查准率 P 表示由算法抽取出的语义标签中正确的百分比;而平衡 F 分数 $F1$ 则是在这两种指标之间的一种平衡度量,它被定义为查准率和查全率的调和平均数。

5.2　实验结果及分析

本次实验主要分析基于 K-means 的语义抽取算法在数据集上的性能表现,将其与基于 TF-IDF 的提取算法、基于 TextRank 的提取算法进行比较,使用查全率 R、查准率 P、平衡 F 分数 $F1$ 3 个指标来衡量算法的实验效果,分析本文所提出的模型在语义标签抽取任务中的优势。

图 3 从查准率 P 的角度,对比了本文算法、TF-IDF 算法、TextRank 算法在知网数据集上的实验性能。由图 3 易知,本文算法比 TF-IDF 算法在查准率上提升了 13.2%;比 TextRank 算法在查准率上提升了 10.1%。

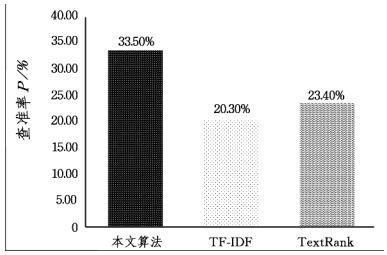

图 3　查准率的对比

图 4 从查全率 R 的角度对比了本文算法、TF-IDF 算法、TextRank 算法在知网数据集上的实验性能。由图 4 易知,本文算法比 TF-IDF 算法在查全率上提升了 11.1%;比 TextRank 算法在查全率上提升了 8.8%。

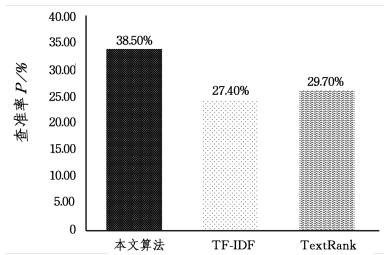

图 4　查全率的对比

图 5 从平衡 F 分数 $F1$ 的角度对比了本文算法、TF-IDF 算法、TextRank 算法在知网数据集上的实验性能。由图 5 易知,本文算法比 TF-IDF 算法在平衡 F 分数 $F1$ 上提升了 12.5%;比 TextRank 算法在平衡 F 分数 $F1$ 上提升了 9.6%。

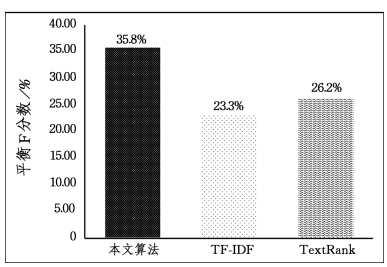

图 5　平衡 F 分数 $F1$ 的对比

基于以上结果可知,在语义标签提取任务中,本文算法明显优于 TF-IDF 算法和 TextRank 算法。其原因在于单词向量是一种基于全体文本信息的语义表示技术,这使得本文方法克服了 TextRank 算法与 TF-IDF 算法只在当前文本中做信息度量的“短视”问题,因而更加丰富的语义信息会被纳入本算法的计算过程。而本文算法在此基础上利用了单词向量良好的向量平移性,使用余弦距离作为语义相似度的度量指标,因此将词向量丰富的全文表达能力利用起来。这些特点都是 TF-IDF 算法与 TextRank 算法所不能兼顾到的,因此本文算法具有更好的实验效果。

结束语　本文将分布式假说纳入语义标签提取任务中,并结合 K-means 算法提出了一种效果较好的语义标签提取算法。而单词分布式表示是一种很有效的信息数字化技术,随着不同的词向量表示特性(比如向量平移性)的挖掘,使用不同的向量相似性度量方式可以使算法的效率得到进一步的提高。另外,在计算语义中心点时,可以考虑使用具有更强先验假设的聚类算法,比如高斯混合聚类等,这在算法的计算精度上会有较大的提升空间。

(下转第 438 页)

Means 聚类方法对数据进行聚类分析，将顾客细分为 5 种不同价值的群体。本文通过层次分析法将 R，F，A 的权重定义为 $w_R=0.072$，$w_F=0.279$，$w_A=0.649$，利用 $RFA=w_R\times R+w_F\times F+w_A\times A$，经过一系列 R 语言统计分析算法，得出具体的部分顾客价值分析结果，如表 5 所列。

表 5 RFA 模型的顾客价值分析结果

ID	R	F	A	W_RFA	LB	价值级别	N
1001	0.4523	0.4	0.4214	0.4392	1	1	755
1004	0.4660	0.4	0.3375	0.3642	4	2	111
1006	0.4037	0.2	0.2098	0.2211	3	3	802
1007	0.4080	0.2	0.1929	0.2031	2	4	480
1008	0.0331	0.2	0.1482	0.1544	5	5	453

从表 5 可以看出，价值最大的客户群体是第一类客户，包括 755 名顾客，占肯德基所有参与者的 29%。这类客户的订单频率高，单次订单消费金额大，可将其定义为肯德基商家的铂金顾客群，商家可以重点保持这类客户。

第二类最有价值的客户群体是第四类客户，包括 111 名客户，占肯德基所有参与者的 4%。这类客户的订单交易频繁，但平均单次订单金额不高，可将这类客户定义为肯德基商家的黄金客户群，将重点发展这类客户。

第三类客户群包括 802 名客户，占肯德基所有参与者的 31%。这类客户订单交易不太频繁，将其定义为肯德基商家的银质客户群，商家应重点培养这类客户，实施针对性策略，尽可能提升这类客户群的价值。

第四类客户群包括 480 名客户，占肯德基所有参与者的 18%。这类客户在 R 和 F 方面类似于第三类客户，区别在于这类顾客的单次消费金额较低，可将这类客户定义为肯德基商家的铜质客户群。

价值最低的客户群是第五类客户群，包括 453 名客户，占肯德基所有参与者的 17%。这类客户订单次数少，单次订单金额低，对商家价值低，可将其定义为肯德基商家的铁质客户群。

结束语 通过以上分析，综合 R，F，A 值对客户进行分类。以肯德基商家为例，从对商家的价值角度分析客户，能更全面地反映客户对商家的重要程度，可以辅助商家为不同价值的客户群体制定相对应的营销策略，例如商家可以对铂金价值的客户群体采取频繁下单打折、优惠券发放等行为，以提高客户满意度以及商家的业务盈利水平。

参 考 文 献

[1] 李雪苑. 浅谈第三方外卖平台的运营管理——以百度外卖为例[J]. 经贸实践，2016(8)：208.

[2] SONG M，ZHAO X，HAIHONG E，et al. Statistics-based CRM approach via time series segmenting RFM on large scale data[C]// International Conference on Utility and Cloud Computing. IEEE，2017：282-291.

[3] 徐翔斌，王佳强，涂欢，等. 基于改进 RFM 模型的电子商务客户细分[J]. 计算机应用，2012，32(5)：1439-1442.

[4] 吴晓雪. 基于 RFM 改进模型的互联网金融平台用户细分研究[D]. 北京：北京交通大学，2016.

[5] 王召义，汪琪. 基于改进 RFM 模型的产品推荐算法[J]. 宿州学院学报，2016，31(11)：101-104.

[6] 何敏，张洪伟，张波. 模糊 ISODATA 及在 CRM 中的应用[J]. 计算机应用，2005，25(6)：1455-1457.

[7] HAN J W，KAMBER M. Data mining：Concepts and techniques[M]. 北京：机械工业出版社，2002.

[8] 耿俊成，袁少光，万迪明，等. 基于改进 RFM 模型的电力客户缴费渠道分析预测[J]. 电力信息与通信技术，2017，15(8)：55-59.

[9] 邓雪，李家铭，曾浩健，等. 层次分析法权重计算方法分析及其应用研究[J]. 数学的实践与认识，2012，42(7)：93-100.

[10] 刘朝华. 基于客户价值的客户分类模型研究[D]. 武汉：华中科技大学，2008.

[11] 张良均，云伟标，王路，等. R 语言数据分析与挖掘实战[M]. 北京：机械工业出版社，2015：46-47.

(上接第 421 页)

参 考 文 献

[1] 文继军，王珊. SEEKER：基于关键词的关系数据库信息检索[J]. 软件学报，2005，16(7)：1270-1281.

[2] 张阔，李涓子，吴刚，等. 基于关键词元的话题内事件检测[J]. 计算机研究与发展，2009，46(2)：245-252.

[3] 李峰，黄金柱，李舟军，等. 使用关键词扩展的新闻文本自动摘要方法[J]. 计算机科学与探索，2016，10(3)：373-380.

[4] 吴舜尧，邵峰晶，王金龙，等. 融合语义资源和关键词的文本聚类[J]. 计算机工程，2014，40(4)：223-227.

[5] VIDAL M，MENEZES G V，BERLT K，et al. Selecting Keywords to Represent Web Page Using Wikipedia Information[J]. WebMedia，2012，4(10)：15-18.

[6] TURNEY P D. Learning Algorithms for Keyphrase Extraction[J]. Information Retrieval，2000，2(4)：303-336.

[7] BELLAACHIA A. NE-Rank：A Novel Graph-based Keyphrase Extraction in Twitter[J]. Web Intelligence and Intelligent Agent Technology，2013，1(12)：372-379.

[8] 李然，张华平，赵燕平，等. 基于主题模型与信息熵的中文文档自动摘要技术研究[J]. 计算机科学，2014，41(S2)：298-300.

[9] 刘通. 基于复杂网络的文本关键词提取算法研究[J]. 计算机应用研究，2016，33(2)：365-369.

[10] 陈伟鹤，刘云. 基于词或词组长度和频数的短中文文本关键词提取算法[J]. 计算机科学，2016，43(12)：50-57.

[11] 王立霞，淮晓永. 基于语义的中文文本关键词提取算法[J]. 计算机工程，2012，38(1)：1-4.

[12] 李鹏，王斌，石志伟，等. Tag-TextRank：一种基于 Tag 的网页关键词抽取方法[J]. 计算机研究与发展，2012，49(11)：2344-2351.

[13] 罗燕，赵书良，李晓超，等. 基于词频统计的文本关键词提取方法[J]. 计算机应用，2016，36(3)：718-725.

[14] 李晓超，赵书良，罗燕，等. 中文文本同频词统计规律及在关键词提取中的应用[J]. 计算机应用研究，2016，33(4)：1007-1012.

[15] 潘虹，徐朝军. LCS 算法在术语抽取中的应用研究[J]. 情报学报，2010，29(5)：853-857.

[16] 车海燕，冯铁，张家晨，等. 面向中文自然语言文档的自动知识抽取方法[J]. 计算机研究与发展，2013，50(4)：834-842.

[17] 夏天. 词语位置加权 TextRank 的关键词抽取研究[J]. 现代图书情报技术，2013，29(9)：30-34.

[18] 方康，韩立新. 基于 HMM 的加权 TextRank 单文档的关键词抽取算法[J]. 信息技术，2015，4(4)：114-116.

[19] 顾益军. 融合 LDA 与 TextRank 的关键词抽取研究[J]. 现代图书情报技术，2014，30(7)：41-47.

[20] BENGIO Y，DUCHARME R，VINCENT P，et al. A Neural Probabilistic Language Model[J]. Journal of Machine Learning Research，2003，3(6)：1137-1155.