Research Article

Approach for Text Classification Based on the Similarity Measurement between Normal Cloud Models

Jin Dai and Xin Liu

College of Software Engineering, Chongqing University of Posts and Telecommunications, Chongqing 400065, China

Correspondence should be addressed to Jin Dai; daijin@cqupt.edu.cn

Received 16 October 2013; Accepted 8 January 2014; Published 23 February 2014

Academic Editors: R. Valencia-García and Y.-B. Yuan

Copyright © 2014 J. Dai and X. Liu. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

The similarity between objects is the core research area of data mining. In order to reduce the interference of the uncertainty of nature language, a similarity measurement between normal cloud models is adopted to text classification research. On this basis, a novel text classifier based on cloud concept jumping up (CCJU-TC) is proposed. It can efficiently accomplish conversion between qualitative concept and quantitative data. Through the conversion from text set to text information table based on VSM model, the text qualitative concept, which is extraction from the same category, is jumping up as a whole category concept. According to the cloud similarity between the test text and each category concept, the test text is assigned to the most similar category. By the comparison among different text classifiers in different feature selection set, it fully proves that not only does CCJU-TC have a strong ability to adapt to the different text features, but also the classification performance is also better than the traditional classifiers.

1. Introduction

Text classification is key technology in text mining, which denotes the task of assigning raw text documents to one or more predefined categories. This is a direct concept from machine learning, which implies the declaration of a set of labeled categories as a way to represent the documents and a statistical classifier trained with a labeled training set. Classification is the process in which objects are initially recognized, differentiated, and understood and implies that objects are grouped into categories, usually for some specific purposes. Classification is fundamental in prediction, inference, and decision making. However, there are a variety of ways to approach classification task [1].

In recent years, with the continued development and innovation of information technology and natural language processing (NLP) fields, it has laid a solid theory and practice basis for text classification. An increasing number of supervised classification approaches have been developed for various types of classification tasks, such as decision trees (DT) [2, 3], neural networks (NN) [4, 5], naive Bayes (NB) [6–8], support vector machines (SVM) [9–12], and k nearest neighbor (kNN) [13–15]. These classifiers have their own characteristics, but in the perspective of comprehensive performance, SVM, kNN, and NB methods are more excellent [16].

SVM can handle high-dimensional sparse text and is not sensitive to the characteristics of relevance. But its classification speed is too slow and lacks the method for multi-class classification. In addition, the choice of kernel function and parameters still depended on experience [9]. The kNN method has the advantage of simple and stable performance and especially has good anti-interference to the noise data. But, it requires a large number of samples to get threshold. At the same time, how to choose k value has a direct impact on the classification performance [14]. NB is simple, efficient, and stable. It requires few parameters and is not sensitive to the missing data. However, NB assumes that the attributes are mutually independent, but in practice the assumption is often not established. Especially, if the amount of attributes is larger or the correlation is great between attributes, its efficiency is low [17].

Cloud model [18–20] is an uncertainty conversion model between the quality concepts and its quantity value expressed by the natural language value. The character of cloud can be expressed by expected value (Ex), entropy (En), and hyper entropy (He) [18]. Ex is the center value of concept in the theory field, which represents the value of the qualitative
The Scientific World Journal

2 TheScientificWorldJournal

0.75
0.5
0.25
0 1 01 52 02 55
0.75
0.5
0
0.25
y1
y2
S2S1
(a)

0 1 0 1 5 2 0 2 5 5
0.75
0.5
0
0.25
y1
y2
S1
S3
S2
(b) Ex1 > Ex2

0 1 0 1 5 2 0 2 5 5
0.75
0.5
0
0.25
y1
y2
S1
S3
S2
(c) Ex1 ≤ Ex2

Figure 1: Similar situation between normal clouds.

2. Similarity Measurement between Normal Cloud Models

In cloud model theory, the cloud vector $\vec{C} = \{Ex, En, He\}$ is used to describe the qualitative concept. When making comparison of the similarity between vectors, Euclidean Distance or Cosine Angle is widely used. However, the physical meaning and the importance of each dimensionality of the cloud vector are completely different. Therefore, a novel similarity measurement between cloud vectors is required. Taking into account the normal cloud model with most universal, we propose the similarity degree between normal clouds.

Normal random distribution is the core of this similarity degree, which is obtained by calculating the area of two intersecting normal cloud vectors. The intersecting stations of two normal clouds are shown in Figure 1 (the shadow area presents each other's similarity).

Given $\vec{C}_1 = \{Ex_1, En_1, He_1\}$, $\vec{C}_2 = \{Ex_2, En_2, He_2\}$, $y_1(x)$ and $y_2(x)$ are, respectively, the distribution function of $\vec{C}_1$ and $\vec{C}_2$, where $x_0$ is the intersection of the two curves $y_1$ and $y_2$. Then, the intersecting area of $\vec{C}_1$ and $\vec{C}_2$ is

$$S = \int_{-\infty}^{x_0} y_2(x) dx + \int_{x_0}^{\infty} y_1(x) dx$$

$$= \int_{-\infty}^{z_2} \phi(z) dz + \int_{z_1}^{\infty} \phi(z) dz,$$

where $z_1 = (x_0 - Ex_1)/En_1$, $z_2 = (x_0 - Ex_2)/En_2$, $\phi(z)$ is standard normal distribution. If $x_0$ is known, $z_1$ and $z_2$ are then obtained. Enquiring the table of standard normal distribution, the intersecting area $S$ can be calculated.

On the basis of the intersection between normal cloud vectors, $|z_1| = |z_2|$, then obtains that

$$x_0^{(1)} = \frac{Ex_2 En_1 - Ex_1 En_2}{En_1 - En_2},$$

$$x_0^{(2)} = \frac{Ex_1 En_2 + Ex_2 En_1}{En_1 + En_2}.$$  

In light of "3En" principle of normal distribution, 99.74% of values are in $[Ex - 3En, Ex + 3En]$. So, when calculating the similarity of two normal distributions, we only consider the
distribution of variables in the interval. It would be well as if
set Ex_1 \leq Ex_2; then, there are the three following situations
about the distribution of x^{(i)} and x^{(j)}:

1. If x^{(i)} \notin [Ex_3 - 3En_3, Ex_i + 3En_i], indicating that
   the values distribution of intersections can be neglected, so
   S = 0.

2. If there is a point (x^{(i)} or x^{(j)}) in [Ex_2 - 3En_2, Ex_1 +
   3En_1], as shown in Figure 1(a), then,

\[
S = S_1 + S_2 + S_3
= \int_{-\infty}^{z_1} \phi(x) \, dx + \left( \int_{-\infty}^{z_2} \phi(x) \, dx - \int_{-\infty}^{z_1} \phi(x) \, dx \right) + \int_{z_1}^{\infty} \phi(x) \, dx
\]

where x^{(i)} \leq x^{(j)} = (x^{(j)} - Ex_j)/En_j.

Considering the normalization of the similarity, S must be
normalized. The area is seen as the similarity of two normal
distributions after it is normalized.

**Definition 1** (similarity of normal cloud). Given normal cloud
\( \tilde{C}_i = \{Ex_i, En_i, He_i\} \), \( \tilde{C}_j = \{Ex_j, En_j, He_j\} \), their similarity
degree is

\[
\gamma(\tilde{C}_i, \tilde{C}_j) = \frac{2S}{\sqrt{2\pi} (En_i + En_j)},
\]

where S is the intersection area between \( \tilde{C}_i \) and \( \tilde{C}_j \).

3. Cloud Virtual Pan-Concept-Tree and Concept Jumping Up

3.1. Cloud Virtual Pan-Concept-Tree. The core idea of data
mining is to discover and obtain the potential knowledge.
In this process, the knowledge layers will be different with
different knowledge granularity. To deal with the uncertainty
of qualitative concepts, cloud model can be used to text
concept representation. Text concepts set C is composed of
a series of basic concepts, which can be represented as cloud
cloud vector \( C_i(Ex_i, En_i, He_i) \). On this basis, text virtual concept
tree can be structured layer by layer.

The virtual concept tree, which is based on cloud model,
is uncertainty. On the same layer, the distinction between
the various concepts is flexible. A certain degree of overlap
is allowed. In other words, the attributes with the same
value may belong to different concepts and different attribute
does not have a contribution to concepts. In the construction
process, the layer of concept extraction is uncertain. It can
be extracted from the bottom layer or the upper layers.

In the data mining process, with the increasing of the
granularity and the abstraction degree to concept, there is
no physical structure of pan-concept-tree. In the same time,
4. Text Classification Approach Based on Cloud Concept Jumping Up

In order to apply cloud model theory to text mining, we need to implement the corresponding preoperation to text, which involves the construction of text information table and conversion.

4.1. Text Information Table and Conversion

**Definition 2.** An information table [5] is defined as \( S = \langle U, R, V, f \rangle \), where \( U \) is a nonempty finite set of objects, and \( R \) is a nonempty finite set of attributes, \( R = C \cup D \), where \( C \) is the set of condition attributes and \( D \) is the set of decision attributes, \( D \neq \phi \). \( V = \cup V_{p} \), \( p \in R \), and \( V_{p} \) are the domain of the attribute \( p \cdot f : U \times R \rightarrow V \) is a total function such that \( f(x_i, p) \in V_p \) for every \( p \in R, x_i \in U \).

Using cloud model to describe the uncertainty between texts, it must be met that the values of different attribute in the text belong to the same domain. That is, the values of different attribute should have the same physical meaning.

**Algorithm 4.** The conversion algorithm for text information table.

**Input.** text information table \( S \)

**Output.** conversion text information table \( S_c \)

**Step 1.** Set \( S_c = S \)

**Step 2.** For each \( D_k \) in \( D \), \( Do//D \) is categories set of \( S_c \)

\begin{align*}
En &= En_1 + En_2; \\
He &= \frac{He_1En_1 + He_2En_2}{En_1 + En_2}.
\end{align*}

\( (6) \)

**Step 3.** Return \( S_c \).
After the conversion by Algorithm 4, the attributes of the text information table are changed into the same physical space. The novel table shows the fluctuation degree of the values in the different category and describes their statistical distribution.

4.2. Cloud Concept Jumping Up Classifier. On the basis of text information table conversion and similarity calculation, the text classifier based on cloud concept jumping up (CCJU-TC) is proposed. The classifier works as follows (Figure 4).

Whole CCJU-TC algorithm is divided into text preprocessing, text information table conversion, category concept extraction, text cloud model conceptual similarity, and several other components.

**Algorithm 5. CCJU-TC (The text classifier based on cloud concept jumping up)**

**Input.** training text set $T$, test text $d$ (unknown category)

**Output.** the category of $d$

1. Text (include $T$ and $d$) segment and remove stopped items;
2. Compute the weight of items by TF-IDF [9] formula;
3. Text features (items) selection;
4. Construct information table $S = ⟨ T, C \cup D, V, f ⟩$, where $C$ is text feature set and $D$ is category set;
5. Invoke Algorithm 4 to convert $S$ to $S_c$;
6. Loop each category $D_j$ in $D$, calculate its concept cloud vector. Finally obtain categories concept set $C_{concept}$
   
   (a) For $j = 1$ to $|D_j| - 1$
   
   (b) $C_{T_j} = C_{T_j} \text{MERGE } C_{T_j_{\text{previous}}}$ //obtain category concept by concept merging, where $T_j$, $T_j_{\text{previous}}$ are texts with the same category $D_j$, $C_{T_j}$ and $C_{T_j_{\text{previous}}}$ are their corresponding cloud models.
   
   (c) Next
   
   (d) $C_{concept} = C_{concept} \cup C_{T_j}$;

7. Compute $\max(\mu(C_d, C_{concept}))$, where $C_d$ is cloud vector of $d$, $\mu(C_d, C_{concept})$ is the cloud similarity between $C_d$ and any category concept in $C_{concept}$.

The category with most great similarity is the category of text $d$.

Step 8. Return.

5. Experiments and Evaluations

5.1. Experiment on Different Datasets. In order to evaluate the performance of CCJU-TC facility, we have acquired four datasets of varying characteristics. Each dataset has its own unique characteristics in terms of the degree of similarity between categories and the dimensionality of categories, as shown in Table 1.

The Featured Articles dataset was designed and organized by our research group by extracting different types of articles from Wikipedia website. A total of 1159 articles were acquired from twenty-three randomly selected categories. 10 documents from each category have been randomly selected to build the training set. The remaining documents were utilized for testing purposes. In other word, the training set consists of 230 documents, while the testing set consists of a total of 929 documents.

The Vehicles dataset was built by extracting vehicle related articles from Wikipedia website. This dataset was acquired by extracting articles from four subcategories in the category of “Vehicles.” All the four categories are easily differentiated and each category has its own unique keywords. This dataset consists of 640 documents. Each category consists of 160 documents where 50 documents were used to build the training set and the remaining 110 documents were utilized for testing purposes. In other words, the training set consists of a total of 200 documents, while the testing set contains a total of 440 documents.

A dataset containing articles about mathematical topics has been acquired from arxiv.org. This dataset consists of eight categories regarding mathematical topics. 40 documents for each category have been collected, and the entire dataset consists of a total of 320 documents. 10 documents from each category were extracted randomly to build the training set, while the remaining 30 documents from each category were used for testing purposes.

20-Newsgroups dataset is one of the standard benchmark datasets used by many text classification research groups to evaluate the performance of their presented classification approaches. 20-Newsgroups dataset is a collection of 20,000 Usenet articles from twenty different newsgroups with 1000 articles per newsgroup. 20-Newsgroups collection has become a widely used dataset for experiments in text applications of machine learning techniques, such as text classification and text clustering. 20-Newsgroups dataset used in our experiments was acquired from the CMU Text Learning Group’s website. In our experiments using this dataset, every category was divided into two subsets. 300 documents from each category were divided for training, while the remaining 700 documents were used for testing purposes.

<table>
<thead>
<tr>
<th>Datasets</th>
<th>Degree of similarity between categories</th>
<th>Number of articles</th>
<th>Dimensionality of categories</th>
</tr>
</thead>
<tbody>
<tr>
<td>Featured Articles (Wikipedia)</td>
<td>Normal</td>
<td>1159</td>
<td>23</td>
</tr>
<tr>
<td>Vehicles (Wikipedia)</td>
<td>Low</td>
<td>640</td>
<td>4</td>
</tr>
<tr>
<td>Mathematics (Arxiv.org)</td>
<td>High</td>
<td>320</td>
<td>8</td>
</tr>
<tr>
<td>20-Newsgroups (CMU Text Learning Group)</td>
<td>Normal</td>
<td>20,000</td>
<td>20</td>
</tr>
</tbody>
</table>
In other words, the training set consists of 6000 documents and the remaining 14,000 documents are used for testing purpose.

Many research works in text document classification apply preprocesses such as stop word elimination, word stemming, and feature selection to the datasets used in their experiments and evaluations in order to obtain better experimental results.

As our research goal in this paper is to evaluate the performance of CCJU-TC facility without sacrificing the simplicity and low cost classification algorithms, we did not perform any preprocess such as stop word elimination, word stemming to the datasets that we used in our experiments. In order to reduce the interference of different feature set, IG, CHI [21, 22], and FAS [23, 24] feature selection methods are adopted.

In all experiments, we employ SVM (Torch), kNN \((k = 30)\), and NB classifiers as a comparison. In the way to construct the training set and test set, we apply the 3-fold cross validation, which randomly divides the text sets into three parts. Two parts are used as training sets; the other left is test set. Finally, the average of three classification results is the experimental result, which is evaluated by precision rate \((p)\), recall rate \((r)\), and \(F_1(2 \times p \times r/(p + r))\). The details of experimental result are in Tables 2, 3, and 4.

In order to compare the performance of the classifier more intuitive, Figure 5 is created based on the above tables.

5.2. Summary of Experiment. In the above classification experiment, CCJU-TC has the better performance, followed by kNN and SVM, NB classification performance is the worst (Figure 5). Meanwhile, the different feature selection method for text categorization performance impact is also different. The performance of the text classifier, which selected features by FAS, is higher and stable (Figure 5(c)). CHI feature selection method for text classification more significant impact, mainly due to over-reliance on low-frequency features, especially in the 20-Newsgroups dataset (Figure 5(b)).

Through the comparison test among multiple text classifiers with a variety of feature selection methods on the different datasets, CCJU-TC has shown an excellent text classification capability. It not only has good ability to adapt

### Table 2: Classification performance of different classifier with IG.

<table>
<thead>
<tr>
<th>Datasets</th>
<th>SVM (p) (%)</th>
<th>SVM (r) (%)</th>
<th>SVM (F_1)</th>
<th>kNN (p) (%)</th>
<th>kNN (r) (%)</th>
<th>kNN (F_1)</th>
<th>NB (p) (%)</th>
<th>NB (r) (%)</th>
<th>NB (F_1)</th>
<th>CCJU-TC (p) (%)</th>
<th>CCJU-TC (r) (%)</th>
<th>CCJU-TC (F_1)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Featured Articles</td>
<td>72</td>
<td>75</td>
<td>73.47</td>
<td>75</td>
<td>77</td>
<td>75.99</td>
<td>70</td>
<td>71</td>
<td>70.50</td>
<td>89</td>
<td>92</td>
<td>90.48</td>
</tr>
<tr>
<td>Vehicles</td>
<td>78</td>
<td>81</td>
<td>79.47</td>
<td>81</td>
<td>82</td>
<td>81.50</td>
<td>72</td>
<td>73</td>
<td>72.50</td>
<td>86</td>
<td>89</td>
<td>87.47</td>
</tr>
<tr>
<td>Mathematics</td>
<td>80</td>
<td>81</td>
<td>80.50</td>
<td>78</td>
<td>80</td>
<td>78.99</td>
<td>73</td>
<td>75</td>
<td>73.99</td>
<td>87</td>
<td>88</td>
<td>87.50</td>
</tr>
<tr>
<td>20-Newsgroups</td>
<td>75</td>
<td>76</td>
<td>75.50</td>
<td>77</td>
<td>78</td>
<td>77.50</td>
<td>70</td>
<td>73</td>
<td>71.47</td>
<td>81</td>
<td>83</td>
<td>81.99</td>
</tr>
<tr>
<td>Average</td>
<td>76.25</td>
<td>78.25</td>
<td>77.23</td>
<td>77.75</td>
<td>79.25</td>
<td>78.49</td>
<td>71.25</td>
<td>73</td>
<td>72.11</td>
<td>85.75</td>
<td>88</td>
<td>86.86</td>
</tr>
</tbody>
</table>

### Table 3: Classification performance of different classifier with CHI.

<table>
<thead>
<tr>
<th>Datasets</th>
<th>SVM (p) (%)</th>
<th>SVM (r) (%)</th>
<th>SVM (F_1)</th>
<th>kNN (p) (%)</th>
<th>kNN (r) (%)</th>
<th>kNN (F_1)</th>
<th>NB (p) (%)</th>
<th>NB (r) (%)</th>
<th>NB (F_1)</th>
<th>CCJU-TC (p) (%)</th>
<th>CCJU-TC (r) (%)</th>
<th>CCJU-TC (F_1)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Featured Articles</td>
<td>70</td>
<td>71</td>
<td>70.50</td>
<td>71</td>
<td>72</td>
<td>71.50</td>
<td>68</td>
<td>68</td>
<td>68.00</td>
<td>83</td>
<td>85</td>
<td>83.99</td>
</tr>
<tr>
<td>Vehicles</td>
<td>72</td>
<td>74</td>
<td>72.99</td>
<td>75</td>
<td>77</td>
<td>75.99</td>
<td>70</td>
<td>71</td>
<td>70.50</td>
<td>80</td>
<td>82</td>
<td>80.99</td>
</tr>
<tr>
<td>Mathematics</td>
<td>74</td>
<td>74</td>
<td>74.00</td>
<td>73</td>
<td>73</td>
<td>73.00</td>
<td>66</td>
<td>69</td>
<td>67.47</td>
<td>81</td>
<td>82</td>
<td>81.50</td>
</tr>
<tr>
<td>20-Newsgroups</td>
<td>69</td>
<td>70</td>
<td>69.50</td>
<td>70</td>
<td>72</td>
<td>70.99</td>
<td>62</td>
<td>65</td>
<td>63.46</td>
<td>75</td>
<td>78</td>
<td>76.47</td>
</tr>
<tr>
<td>Average</td>
<td>71.25</td>
<td>72.25</td>
<td>71.75</td>
<td>72.25</td>
<td>73.5</td>
<td>72.87</td>
<td>66.5</td>
<td>68.25</td>
<td>67.36</td>
<td>79.75</td>
<td>81.75</td>
<td>80.74</td>
</tr>
</tbody>
</table>
### Table 4: Classification performance of different classifiers with FAS.

<table>
<thead>
<tr>
<th>Datasets</th>
<th>SVM</th>
<th>kNN</th>
<th>NB</th>
<th>CCJU-TC</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>p (%)</td>
<td>r (%)</td>
<td>F₁ (%)</td>
<td>p (%)</td>
</tr>
<tr>
<td>Featured Articles</td>
<td>78</td>
<td>79</td>
<td>78.50</td>
<td>80</td>
</tr>
<tr>
<td>Vehicles</td>
<td>80</td>
<td>82</td>
<td>80.99</td>
<td>83</td>
</tr>
<tr>
<td>Mathematics</td>
<td>82</td>
<td>84</td>
<td>82.99</td>
<td>80</td>
</tr>
<tr>
<td>20-Newsgroups</td>
<td>79</td>
<td>79</td>
<td>79.00</td>
<td>79</td>
</tr>
<tr>
<td><strong>Average</strong></td>
<td>79.75</td>
<td>81</td>
<td>80.37</td>
<td>80.50</td>
</tr>
</tbody>
</table>

**Figure 5:** Classification results comparison.
different feature set, but also has better classification ability. It full proves that CCJU-TC is an efficient classifier.

6. Conclusion

In the present research to text mining (TM), traditional data mining methods still dominated. However, with further research, it faces more severe challenges. These difficulties, such as the huge dimensions and sparsity of text object, the high complexity of algorithm, and the requirement of prior knowledge, have seriously hampered the development of TM.

Through in-depth analysis, these problems in TM process are due to the uncertainty of natural language. The uncertainty of natural language (especially text) comes from the uncertainty of the human thinking in essence. Although it strengthens understanding of spatial and cognitive for people, brings a series of problems to TM. Therefore, from the point of reducing the complexity of natural language, if we can carry out the advanced innovation, which is based on making full use of these existing technologies, and find out a novel uncertainty artificial intelligence approach for TM, it will greatly facilitate the rapid development of TM.

CCJU-TC classifier is a novel attempt to apply uncertain knowledge acquisition tool (cloud model) to text classification in TM. The experimental result shows that it is an excellent solution. Future, more data mining methods, based on cloud model, will be carried out.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

Acknowledgments

This work is supported by the National Natural Science Foundation of China (NSFC) under Grant no. 61309014; the Natural Science Foundation Project of CQ CSTC under Grant no. cstc2013jcyjA40009, no. cstc2013jcyjA40063; and the Natural Science Foundation Project of CQUST under Grant no. A2012-96.

References
