A Novel Method for Reducing Overhead of Training Sentiment Analysis Network

Yuxia LEI*, Linkun ZHANG, Zhengyan WANG, Zhiqiang KONG, Hanru MA, Xiuhuan LIU
School of Computer Science, Qufu Normal University, Rizhao 276826, China
yx_lei@126.com

Abstract: Sentiment analysis based on statistics has rapidly developed in deep-learning. Bilateral attention neural network (BANN), especially Bidirectional Encoder Representations from Transformers (BERT), has reached high accuracy. However, with the increase of network depth and large-scale corpus, the computational overhead of BANN increases geometrically. How to reduce training corpus scale has correspondingly become an important research focus. This paper proposes a reduced corpus scale method called Concept-BERT, which consists of the following steps: firstly, using Formal Concept Analysis (FCA), Concept-BERT mines the association rules among corpus and reduces corpus attributes, and hence reducing corpus scale; secondly, reduced-corpus is inputed to BERT and the result is obtained; finally, the attention of Concept-BERT is analyzed. Concept-BERT is experimented for sentiment analysis on CoLA, SST-2, Dianping and Blogsenti, and its accuracy reaches 81.1, 92.9, 77.9 and 86.7 respectively. Our experimental results show that the proposed method has the same accuracy as BERT, using low-scale corpus and low overhead, and low-scale corpus doesn't affect model attention.

Keywords: sentiment analysis, bilateral attention, Formal Concept Analysis, corpus scale

1. Introduction

Sentiment analysis plays an important role in Natural Language Processing (NLP), which helps us to mine the emotional tendency in sentiment. With the advent of data time, sentiment analysis has become more extensive applications such as movie evaluation [1], opinion mining [2], behavior prediction [3] and social networks [4]. Generally, there are two kinds of sentiment analysis methods: one is based on rules, and another is based on statistics. In the past, the former was the main method of sentiment analysis, which was a combination of linguistics, semantics and pragmatics. Nowadays, with the improvement of computing and data scale, statistics-based method gradually replaces rule-based ones. There are many sentiment analysis models with excellent effects such as Neural Network Language (NNL) [5], Support Vector Machines (SVM) [6], Word2vec [7], Embedding from Language Models (ELMo) [8], Generative Pre-Training (GPT) [9] and BERT [10].

The understanding capability of these above models is closely related to the scale of corpus used in the training. In general, the larger training corpus is, the higher the accuracy of the models is. However, the current problem is that the corpus is unstructured. There are many interfering and invalid corpus in training corpus. The training requires a lot of computing overhead, which cost is high. Unstructured corpus causes a great waste of computing. Therefore, it is necessary to find a suitable method to deal with the unstructured corpus. Concept lattice, also called FCA, is a natural data processing tool, which was proposed by Wile in 1982 [11]. Concept lattice constructs the concept hierarchy according to binary relationship between attributes and objects. In fact, it embodies the relationship among object, attribute, concept extent and concept intent. It contains comprehensive information, which used in many situations, such as library detection [12], online...
community [13], recommendation system [14].

Concept lattice can be used in the preprocessing of sentiment analysis corpus. Analyzing the internal relevance, dependence and corpus structure is more important than focusing on corpus itself. Using concept lattice association rules to reduce corpus scale and constructing fuzzy concept lattice based on ontology can reflect the constraints of corpus relevance. Attribute dependence on fuzzy transformation and information structure dependence can measure corpus dependence. Therefore, this paper analyzes corpus, combining concept lattice, to mine corpus features, and reduces corpus scale for sentiment analysis.

The contributions of this paper are listed as follows: 1. The grammar structure of training corpus is generated by analyzing itself, in order to avoid over-learning same structure corpus for models. We extract feature information of corpus to form feature attribute concept lattice, and use high-frequency feature attribute clustering to reduce concept lattice. In this way, the reduced-scale concept lattice structure and concept lattice reduction rules are generated; 2. We use the generated concept lattice reduction rules to reduce the scale of same structure corpus and generate corpus concept lattice; 3. According to concept lattice reduction method and bidirectional attention neural network, we propose Concept-BERT sentiment analysis model. Concept-BERT is scientifically tested on CoLA, SST-2, Dianping and Blogsenti to prove its valid.

The paper is organized as follows: the next section gives the basic notations of FCA and related works; Section 3 introduces the framework of Concept-BERT. Section 4 provides some experiments. Finally, section 5 concludes this paper.

2. Related Works

In this section, we will introduce some statistic-based methods and some notations of FCA.

In 2003, Yoshua Bengio proposed the Neural Network Language (NNL) [5], which is the first model that applies neural networks to NLP. NNL uses one-hot, and it predicts words by learning position information of one sentence. The word embedding proposed by him has made outstanding contributions to the development of word embedding neural network in the future. However, at that time, data amount and computing was poor. The model was not well developed and was replaced quickly by a method based on Support Vector Machines [6]. In the era of small-scale data, SVM only requires a small number of samples to achieve good results. However, its robustness for missing data is poor, and its capability of high-dimensional kernel functions explanation is not strong.

In 2013, the Google team led by Tomas Mikolov proposed Word2vec [7], including two important models: Continuous Bag-Of-Words Model (CBOW) and Continuous Skip-gram Model (Skip-gram) [15]. Word2vec uses context to reduce the dimensionality of word embedding, and has also achieves good results in implicit semantics [16]. In 2018, Matthew E. Peter proposed Embedding from Language Model (ELMo) [8]. ELMo uses dynamic embedding, generating different word embeddings according to different contexts, to solve word ambiguity [17]. In the same year, OpenAI proposed Generative Pre-training (GPT) [9]. GPT is the first model to replace LSTM with Transformer for sentiment analysis [18]. Transformer is an encoder with strong feature extraction capabilities, which is stack of Encoder and Decoder. Few months later, Google team proposed the Bidirectional Encoder Representation from Transformers (BERT) [10]. It is a general model for NLP. It adds bidirectional Transformer on the basis of GPT, which improves model's capability to predict specific locations. Compared with existing model, BERT has achieved good results on most tasks.
In the background of deep-learning and big data, deep networks and large-scale corpus have become the development direction of sentiment analysis. Many studies have shown that a deeper network and a large-scale corpus tend to improve the performance of model. However, its computational cost is high, and many corpora is unstructured. It causes model learning many invalid and interference corpus during training. Therefore, corpus preprocessing becomes an essential process of sentiment analysis. How to effectively reduce corpus scale and discover hidden associations of corpus has gradually become a new research focus.

In 2016, Tao Chen and Ruifeng Xu et al. proposed a method of regional sentiment analysis [19]. This method divides corpus into multiple regions, and uses a Convolutional Neural Networks to assign regional weights. It does sentiment analysis on the important corpus. In 2017, Bryan McCann, James Bradbury et al. verified the importance of sentiment granularity for sentiment analysis [20]. They proposed that the smaller emotion granularity, the more necessary it is to filter important emotions in context. In 2020, Qianming Xue, Wei Zhang et al. proposed a domain-recognizer, which uses two feature extractors to detect features of invariant sentiment [21]. Tomoki Ito, Kota Tsubouchi et al. proposed an interpretable neural network lexical initialization learning method, judging primitive-level word emotions out of context first and judging global emotions [22]. Su-Jin Shin, Kyungwoo Song et al. proposed hierarchical clustering to reduce corpus scale [23].

Concept lattice has excellent development, and its reduction methods are also very mature, such as K-means clustering reduction [24], generalized spectral clustering reduction [25], granular computing [26], fuzzy attribute parameterization [27], context vector reduction [28]. These methods are very effective for attribute reduction of concept lattice. In addition, concept scaling can effectively filter the formal concepts [29], and BiC2PAM algorithm can improve concept lattice generation effect after adding context knowledge constraints [30].

Concept lattice takes the concept as the basic analysis element and obtains the background from formal context. It sets partial order relationship to get a complete concept lattice.

**Definition 1 (Formal Background):** Object $G$, attribute $A$ and binary relation $I$ between object and attribute constitutes a new formal background $K$. In general, it is defined as triplet $K = (G, A, I)$. In $I$, for any object $g$ in $G$ and attribute $a$ in $A$, if object $g$ has an attribute $a$, it is recorded as $(g, a) \in K$ or $gla$ [31].

**Definition 2 (Formal Concept):** If two-tuple $(X, M)$ satisfies $X = M$ and $M = X$, then $(X, M)$ is called a formal concept. $X$ is called the extent of concept $(X, M)$. $M$ is called the intent of concept $(X, M)$ [32].

\[
X' = \{a | a \in A, \forall x \in X, xla\} \subseteq G \tag{1}
\]

\[
M' = \{x | x \in G, \forall a \in M, xla\} \subseteq A \tag{2}
\]

**Definition 3 (Partial Order):** In concept $(X_1, M_1)$ and $(X_2, M_2)$, if $X_1 \subseteq X_2$ or $M_2 \subseteq M_1$, $(X_1, M_1)$ is called sub-concept of $(X_2, M_2)$, and $(X_2, M_2)$ is called super-concept of $(X_1, M_1)$. It is denoted as $C_1 \leq C_2$. The association between concept is called partial order, and it is expressed as follows [33]:

\[
(X_1, M_1) \leq (X_2, M_2) \Leftrightarrow A_1 \subseteq A_2 \tag{3}
\]

\[
(X_1, M_1) \leq (X_2, M_2) \Leftrightarrow B_1 \subseteq B_2 \tag{4}
\]

**Definition 4 (Attribute Reduce):** For sub-attribute $A_i \subseteq A$ and $A = A_i \cup \{a\}$, their output
equivalent categories are $G / A_1 = \{x_1, x_2, \ldots, x_n\}$ and $G / A_2 = \{y_1, y_2, \ldots, y_m\}$. If $\forall x_i \in U / A_1$, $y_j \in U / A_2$ and $x_i \subseteq y_j$, attribute $a$ is called $A_1$ unnecessary element in $A_2$. It is also called $A_1$ is a reduction of $A_2$ [34].

3. Method and Model

In this section, we firstly explain the basic idea and framework of Concept-BERT, and then propose the method for training corpus reduction.

3.1 Framework of Concept-BERT

Sentiment analysis can be understood as reflecting specific corpus to different sentiment categories. After giving corpus and tags, sentiment analysis is performed as the following operations [35]:

$$\phi : D \times C \rightarrow \{T, F\}$$

$$D = \{d_1, d_2, \ldots, d_n\} \quad C = \{c_1, c_2, \ldots, c_n\}$$

where, $T$ represents corpus belong to the sentiment. $D$ represents opposite situation. $D = \{d_1, d_2, \ldots, d_n\}$ represents sentiment analysis corpus, $C = \{c_1, c_2, \ldots, c_n\}$ represents sentiment category.

Concept-BERT is composed of two parts. One part uses high-frequency feature clustering method to reduce the feature attributes of corpus. Another part is sentiment analysis by input the reduced-corpus to BERT with one-hot encoding, as shown in Figure 1.

Figure 1. the Framework of Concept-BERT

3.2 Training Corpus Reduction

Sentiment features are important factors that affect the accuracy of sentiment analysis. The training method of BERT is randomly masking a word in a sentence with the [mask] tag. It predicts the probability of a word in a certain position. The grammar structure has a decisive influence on its effectiveness. For example, "This movie makes me happy" and "This movie makes me glad". These two sentences have same grammar structure and express positive sentiments. Their difference is sentiment word. One is “happy” and another is “glad”. In this case, according to Definition 4, these two words can be replaced by one of them, and another word is unnecessary. Therefore, these
two feature attributes can be reduced,

Unstructured corpus often has invalid and interference sentence, so the impact of these factor needs to be reduced. Considering the probability of negative factors, we use frequency-filtering to filter them. TF-IDF is a common method to compute the importance degree of words, which is composed of word frequency and reverse document frequency [36]. The calculation of TF-IDF is as follows:

\[
TF-IDF_{i,j} = tf_{i,j} \times idf_{i,j}
\]  

(6)

The calculation of TF is as follows:

\[
tf_{i,j} = \frac{n_{i,j}}{\sum_k n_{k,j}}
\]  

(7)

where, \( n_{i,j} \) is a word frequency in the corpus, and \( \sum_k n_{k,j} \) is the sum of all words frequency.

IDF is used to measure the universal importance of a word. The calculation of it is as follows:

\[
idf_{i} = \log \frac{|D|}{|\{j; t_i \in d_j\}|}
\]  

(8)

where, \( |D| \) represents the total of corpora, and \( |\{j; t_i \in d_j\}| \) represents the total of corpora containing \( t_i \).

In this way, we can not only delete the interference and invalid corpus, but also find the high-frequency features in corpus. For high-frequency features, we use K-means clustering to cluster them. Assuming that the cluster is divided into \( C_1, C_2, \ldots, C_k \), the goal is to minimize the square error \( E \):

\[
E = \sum_{i=1}^{k} \sum_{\omega \in C_i} \| \omega - \mu_i \|^2
\]  

(9)

where, \( \mu_i \) is the mean vector of \( C_i \):

\[
\mu_i = \frac{1}{|C_i|} \sum_{\omega \in C_i} \omega
\]  

(10)

We use association vector to calculate the semantic similarity and features cluster, the calculation of it is as follows:

\[
Sim(\omega_1, \omega_2) = \frac{\sum_{i=1}^{n} \delta(\omega_{1,i}, \omega_1)(\omega_{2,i}, \omega_2)}{\sqrt{\sum_{i=1}^{n} \omega_{1,i}^2 \sum_{i=1}^{n} \omega_{2,i}^2}}
\]  

(11)

where, \( \omega_{1,i} \) represents position \( i \) of one corpus, and \( \omega_{2,i} \) represents position \( i \) of another corpus.

Therefore, for the training corpus \( D \):

**Algorithm 1:**

**Input:** \( D \)

**Output:** \( Reduced - D \)
Step 1: Calculate \( tf_{i,j} \), \( idf_{i,j} \) and \( TF-IDF_{i,j} \)

Step 2: Calculate the semantic similarity of association vectors \( Sim(\omega_i, \omega_j) \)

Step 3: Run K-means clustering

Step 4: Reduce concept lattice by association rules and Definition 4

For example, the corpora “Nicholson’s understated performance is wonderful.” … “It’s both degrading and strangely liberating to see people working so hard at leading lives of sexy.” Firstly, Concept-BERT performs feature importance analysis, and screen key features to construct formal conceptual background, as shown in Table 1.

|   | A  | B  | C  | D  | E  | F  | G  | H  | I  |
|---|----|----|----|----|----|----|----|----|----|
| 1 | 1  | 0  | 1  | 0  | 0  | 0  | 1  | 0  | 0  |
| 2 | 0  | 1  | 0  | 0  | 1  | 0  | 0  | 0  | 1  |
| 3 | 0  | 0  | 1  | 1  | 0  | 0  | 0  | 0  | 0  |
| 4 | 1  | 0  | 1  | 0  | 0  | 0  | 0  | 1  | 0  |
| 5 | 0  | 1  | 0  | 0  | 1  | 0  | 0  | 0  | 1  |
| 6 | 1  | 0  | 0  | 1  | 0  | 0  | 0  | 0  | 0  |
| 7 | 0  | 0  | 0  | 0  | 0  | 0  | 1  | 0  | 1  |
| 8 | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 1  |
| 9 | 0  | 0  | 1  | 0  | 0  | 1  | 1  | 1  | 0  |
| 10| 0  | 0  | 0  | 1  | 0  | 0  | 0  | 0  | 0  |
| 11| 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 1  |
| 12| 1  | 0  | 0  | 0  | 0  | 0  | 1  | 0  | 1  |

Notes: A, B, C, D, E, F, G, H and I represents “funny”, “best”, “great”, “performance”, “fascinating”, “worst”, “just”, “too” and “hard”.

As shown in Table 1, each column represents an attribute, and each row represents an object. When one object including the attribute, it is recorded as 1. On the contrary, it is recorded as 0. Then, it reduces the concept lattice by algorithm 1. The reduce rules, concept lattice and reduced concept lattice is shown in Figure 2.

Figure 2. Reduce Concept Lattice by Association Rules

As shown in Figure 2, reduce rules are “Fascinating hard=[100%] => best”,
“worst=[100%]=>$too” and “just=[100%]=>$great”. In order to let Concept-BERT learn more attributes from a small-scale corpus, reduction rule also calculates each weight of object-attribute as object weight besides association rules, and then smaller weight object is reduced. From Figure 2, after reducing by association rules, the weights of 12 objects are calculated respectively. Where, 5 of 12 objects need to be reduced due to smaller weights with only one attribute. There are 20 concepts, 9 attributes and 12 objects before reduction. There are 13 concepts, 6 attributes and 7 objects after reduced. Compared with the two cases, 7 concepts, 3 attributes and 5 objects are reduced by association rules. The upper-level objects are abstract, and it contains few attributes. The under-level objects are specific, and it contains more attributes. The object-attribute detail information before reduction is shown in Table 2, and the object-attribute detail information after reduction is shown in Table 3.

| Number  | Feature word | Number  | Feature word |
|---------|--------------|---------|--------------|
| Concept_1 | too          | Concept_10 | great, funny |
| Concept_2 | great        | Concept_11 | performance  |
| Concept_3 | funny        | Concept_12 | best, hard, fascinating |
| Concept_4 | hard         | Concept_13 | fascinating, performance |
| Concept_5 | fascinating  | Concept_14 | worst, too, great, just |
| Concept_6 | worst        | Concept_15 | worst, too, funny |
| Concept_7 | too, great   | Concept_16 | too, great, funny |
| Concept_8 | just         | Concept_17 | just, great, funny |
| Concept_9 | too, funny   | Concept_18 | performance, funny |

| Number  | Feature word | Number  | Feature word |
|---------|--------------|---------|--------------|
| Concept_1 | worst        | Concept_7 | worst, funny |
| Concept_2 | just         | Concept_8 | just, funny  |
| Concept_3 | funny        | Concept_9 | funny, performance |
| Concept_4 | performance  | Concept_10 | Performance, fascinating |
| Concept_5 | fascinating  | Concept_11 | fascinating, hard |

From Table 2 and Table 3, Concept_1 to Concept_18 represents each concept that generates a concept lattice. There may be multiple objects and attributes in each concept.

### 3.3 Bidirectional Encoder Representation from Transformers

Concept-BERT is improved based on BERT. The description of BERT is as follows [10]:

\[
P(\omega_t = "BERT" | \omega_1, \omega_2, \ldots, \omega_{t-1}; \theta) \tag{12}
\]

\[
L = \sum_{\omega \in C} \log P(\omega | context(\omega)) \tag{13}
\]

BERT is a stack of bidirectional Transformer. Its encoder is composed of Self-attention and Feed Forward. Its decoder is composed of Self-attention, Encoder-Decoder Attention and Feed Forward. Concept-BERT inherits from BERT, and its presentation layer, hidden layer, Self-attention heads and Feed Forward are 12, 768, 12 and 3072, respectively, as shown in Figure 3.
From Figure 3, in generally, the input of BERT is one-hot coded word embedding composed of token embedding, segment embedding and position embedding. It is transmitted among different layers through residuals. The calculation of feature matrix is as follows [18]:

\[
z = \text{Attention}(Q, K, V, R) = \text{Softmax} \left( \frac{QK^T}{\sqrt{d_z}} \right) V
\]

\[
Z = \text{MultiHead}(Q, K, V, R) = \text{Concat} (\text{head}_1, \text{head}_2, ..., \text{head}_h) W^o
\]

\[
\text{head}_i = \text{Attention}(QW^q_i, KW^K_i, VW^v_i)
\]

where, \( W^0 \in R^{\text{head}_h \times d_z} \), \( W^K \in R^{\text{head}_h \times d_z} \), \( W^v \in R^{\text{head}_h \times d_z} \) and \( W^o \in R^{d_z \times d_z} \).

### 3.4 Computational Overhead of Concept-BERT

Concept-BERT uses the forward-matching regular expression form, and combines with the following loss function:

\[
L = \sum_{i \in R} l(\{e_i, P(e_i | E, PV; \theta) | i \in S_i \})
\]

\[
L = \sum_{i \in R} \sum_{e_i} e_i \log P(e_i | E, PV; \theta)
\]

Using this method can reduce the computational overhead for deep networks. The forward-matching regular expression is as follows:

\[
\frac{\partial E_i}{\partial E_i} = \frac{\partial E_i}{\partial E_i} \times (1 + \sum_{k=1}^{i-1} \frac{\partial E_k}{\partial E_i})
\]

Under the optimization of forward-matching regular expression, the state of \( L+1 \) layer as follows: \( x_{i+1} = Y(y_0, y_1, ..., y_i) = \sum_{k=0}^{i} W^{i+1}_k LN(e_k) \). Research shows that the preprocessing of corpus greatly improves the performance of models.

### 4. Experiments and Result Analysis

We verify the effectiveness of Concept-BERT on CoLA, SST-2, Dianping and Blogsent by...
Tesla V100 [37]. Detail information on the sentiment analysis experiment settings and results are shown in the following section.

4.1 Dataset

In this paper, the corpora we used contains three labels: number, comment and sentiment. CoLA is a corpus for judging grammatical correctness. The purpose of adding CoLA is to test the scalability of Concept-BERT in other NLP tasks. SST-2 is a movie evaluation corpus. Dianping is a Chinese corpus that records customers' evaluations of products. Blogsenti is the comment of blog replies. Before experiment, we randomly sort the corpus to ensure its repeatability. Then, we use 60% of corpus as the training, 20% as the validation, and the remaining 20% as the test. We also divide reduced-corpus in the same way after using Concept-BERT. We keep same test to ensure the stability of the experiment in both cases. Finally, we added [CLS] and [SEP] at the beginning and ending of each sentence according to model requirements. Table 4 shows the detailed information of corpora.

| Dataset | Scale      | Train | Dev  | Test  |
|---------|------------|-------|------|-------|
| CoLA    | Normal     | 5472  | 1368 | 1711  |
|         | Concept Reduce | 2825  | 707  | 1711  |
| SST-2   | Normal     | 4428  | 1108 | 1384  |
|         | Concept Reduce | 2232  | 559  | 1384  |
| Dianping| Normal     | 7671  | 1918 | 2398  |
|         | Concept Reduce | 3202  | 801  | 2398  |
| Blogsenti| Normal   | 76792 | 19198| 23998|
|         | Concept Reduce | 41794 | 10449| 23998|

4.2 Construct Concept Lattice

We use Concept-BERT to extract the high-frequency features of CoLA, SST-2, Dianping and Blogsenti by Algorithm 1 and Definition 4. The four corpora are reduced under the semantic similarity and K-means clustering concept reduction rules. The reduced concept lattice is shown in Figure 4.
From Figure 4, node is concept, which contains multiple objects and attributes. The blue part represents concept has attributes, and the black part represents concept has objects. These objects and attributes have similar composition structures or association rules. The connection among concepts represents the partial ordering association on different concepts. The objects in upper-level concept include that in lower-level. The upper-level concepts are abstract and have less attributes, and the lower-level are specific and more attributes. The objects are comment, and the attributes are keywords and stars. It mines the derivation rules among keywords by the reduction rules of comment, star and keywords. Then, it reduces attributes for keywords that contains repetition, similarity, and deduction. This process reduces comments quantities, and it uses as few keywords as possible to contain as much sentiment information as possible, thereby reducing the scale of training corpus.

4.3 Parameter Setting

We use the following learning rate function during the training:

$$l = d^{-0.5} \text{mod} \cdot \min(step \_ num^{-0.5}, step \_ num \cdot warmup \_ steps^{-1.5})$$

We use randomly small-scale training corpus to set the learning rate function and find the best learning rate. The result is shown in Figure 5.
rate as $e^{-3}$ on CoLA, $e^{-4}$ on SST-2, $e^{-6}$ on Dianping and $e^{-5}$ on Blogsenti.

4.4 ACC and Epoch

After setting the learning rate of each corpus, we experiment 40 Epochs, and the experimental results are shown in Figure 6.

From Figure 6, ACC gradually rises and stabilizes with Epoch in all corpora. Loss has been decreasing. Val_loss decreases firstly, and then increases. When Loss and Val_loss decrease together, this indicates that the model is learning effectively. When Loss decreasing and Val_loss increasing, this indicates that the model works well in the training, but not works well in the validation and test. At this case, the model learns overfitted.

From Figure 6, when Concept-BERT reaching best effect, the ACC is similar as BERT best effect. Concept-BERT has reaches high accuracy using concept reduction corpus training.

4.5 Evaluation Metrics

In order to strictly analyze the performance of Concept-BERT, we adopted multiple measures, including precision (P), recall (R), F1-score (F1) and accuracy (ACC).

Accuracy, recall, and F1-score are a set of metrics widely used in model evaluation. TP indicates that the sample is positive, and the model predicts it as a positive sample. TN indicates that the sample is negative, and the model predicts it as a negative sample. FP indicates that the sample is negative, and the model predicts it as a positive sample. FN indicates that the sample is positive, and the model predicts it as a negative sample.

The precision is the ratio of positive samples classified correctly to samples classified as positive.
The recall is the ratio of positive samples correctly classified to actual positive samples.

\[
P = \frac{TP}{TP + FP} \quad (21)
\]

F1-score is the weighted average of precision and recall.

\[
F_1 = 2 \cdot \frac{\text{precision} \cdot \text{recall}}{\text{precision} + \text{recall}} \quad (23)
\]

ACC is the ratio of correctly classified test samples to the total number of test samples. It is used to measure the capability of a model to correctly predict classification of new data.

\[
ACC = \frac{TP + TN}{TP + TN + FP + FN} \quad (24)
\]

### 4.6 Results

Table 5 records Concept-BERT results on each corpus.

| Dataset | ACC | F1  | Precision | Recall |
|---------|-----|-----|-----------|--------|
| CoLA    | 81.1| 88.0| 84.0      | 92.0   |
| SST-2   | 92.9| 94.0| 94.0      | 93.0   |
| Dianping| 77.9| 81.0| 78.0      | 83.0   |
| Blogsenti| 86.7| 87.0| 87.0      | 87.0   |

From Table 5, Concept-BERT achieves a higher score than the benchmark when using concept reduction in training, and the computational overhead is reduced. We compare the results with BERT, as shown in Table 6.

| Dataset | Model  | Train | ACC  | F1  | Precision | Recall | Time |
|---------|--------|-------|------|-----|-----------|--------|------|
| CoLA    | BERT   | 5472  | 79.2 | 84.8| 81.0      | 89.0   | 548  |
|         | Concept-BERT | 2825  | 81.1 | 88.0| 84.0      | 92.0   | 319  |
| SST-2   | BERT   | 4428  | 93.2 | 94.7| 93.9      | 95.6   | 569  |
|         | Concept-BERT | 2232  | 92.9 | 94.0| 94.0      | 93.0   | 261  |
| Dianping| BERT   | 7671  | 78.6 | 81.7| 80.3      | 83.1   | 1060 |
|         | Concept-BERT | 3202  | 77.9 | 81.0| 78.0      | 83.0   | 445  |
| Blogsenti| BERT | 76792 | 86.6 | 88.6| 88.2      | 89.0   | 17630 |
|         | Concept-BERT | 41794 | 86.7 | 87.0| 87.0      | 87.0   | 9559 |

From Table 6, it shows that Concept-BERT has achieved the same ACC as BERT. In terms of training corpus scale, Concept-BERT’s scale is about half of BERT. When we experimenting on Tesla V100, we are able to reduce about 46.6% time during training. The computing of Tesla V100 is 7.5 TFLOPS. Therefore, the computational overhead of Concept-BERT and BERT training on each corpus is shown in Table 7.

| Dataset | BERT | Concept-BERT |
|---------|------|--------------|
| CoLA    | 4110.0| 2392.5       |
| SST-2   | 4267.5| 1957.5       |
| Dianping| 7950.0| 3337.5       |
| Blogsenti| 132225.0| 71692.5     |

From Table 7, Concept-BERT reduces about 46.6% computational overhead than BERT.
4.7 Attention Analysis

Attention is the core of bidirectional attention model. Therefore, the attention of Concept-BERT is also an important basis for judging its effectiveness. In order to verify Concept-BERT attention, we use "I like this movie. I hate the ending of this movie" to analyze it after model training. The analysis results are shown in Figure 7.

Figure 7. Attention Analysis of Concept-BERT

Figure 7 (a) depicts most attention focusing on the next word. For [SEP], its attention is directed to [CLS], not the next word. Therefore, attention to the next word only fits inside of sentences. Figure 7 (b) describes most attention focusing on the previous word. For [SEP], its attention is directed to [SEP], not the previous word. Therefore, attention to the previous word only fits inside of sentences. Figure 7 (c) depicts most attention focusing on related word, including the etymology. This attention goes beyond single sentence. It is applied in two different sentences. Figure 7 (d) describes most attention focusing on the related word in other sentences. Figure 7 (e) depicts most attention focusing on the word which can predict the current word excepted etymology. A lot of attention has been turned to the [CLS] in this case. Figure 7 (f) describes most attention focusing on the [CLS] and [SEP]. Using reduced-corpora, Concept-BERT doesn't impact attention by the attention analysis.

5. Conclusion

This paper proposes a method called Concept-BERT for sentiment analysis using low-scale
training corpus. It analyzes the structure of large-scale corpus, and uses high-frequency clustering method to extract features on sentiment. Then, it uses association rules to reduce corpus scale for constructing the reduced concept lattice. The reduction rules are generated during constructing concept lattice. It reduces the impact of invalid and interference corpus by generated reduction rules, thereby reducing corpus scale. We use Tesla V100 with 7.5 TFLOPS computing to do experiment on CoLA, SST-2, Dianping and Blogsent. The experimental results show that Concept-BERT achieves the same score in low-scale training corpus as BERT in large-scale training corpus. Concept-BERT reduces computational overhead while ensuring the effect of sentiment analysis. Combining the four corpora, its computational overhead reduced about 46.6%.

Several problems remain unsolved. Some of the interesting problems are how to improve the attention and feature extraction capabilities of Concept-BERT to relevance of different word embeddings, and how to applicate Concept-BERT to spatio-temporal data and service recommendation [38-45]. In the future, we will focus on solving this problem.

Abbreviations
Bilateral Attention Neural Network (BANN), Bidirectional Encoder Representations from Transformers (BERT), Formal Concept Analysis (FCA), Natural Language Processing (NLP), Neural Network Language (NNL), Support Vector Machines (SVM), Language Models (ELMo), Generative Pre-Training (GPT), Continuous Bag-Of-Words Model (CBOW).

Authors’ contributions
YL and LZ designed the study. LZ and ZW performed the experiments and wrote the paper. ZK, HM and XL collected and reshuffled corpora. YL reviewed and edited the manuscript. All authors read and approved the final manuscript.

Authors’ information
Yuxia Lei obtained his Ph.D. degree in computer software and theory from the Institute of Computing Technology, CAS, in 2010. He is an associate professor. His current research interests include concept lattice, artificial intelligence, and machine learning. Linkun Zhang received the B.Sc.degree in computer science from Qufu Normal University, Rizhao, China, in 2018. Zhengyan Wang received the B.Sc.degree in computer science from Qufu Normal University, Rizhao, China, in 2018. Zhiqiang Kong received the B.E.degree from Shandong First Medical University, Taian, China, in 2019. Hanru Ma received the B.E.degree from Qilu University of Technology, Jinan, China, in 2019. Xiujuan Liu received the B.E.degree from Weifang Institute of Technology, Weifang, China, in 2019.

Funding
This work is partly supported by the Undergraduate Education Reform Project in Shandong Province (no. Z2018S022).

Availability of data and materials
The original corpora come from github and it can be found in https://github.com/SophonPlus/ChineseNlpCorpus/tree/master/datasets.

Competing interests
The authors declare that they have no competing interests.

References
[1] Palkar, R. K., Gala, K. D., Shah, M. M., & Shah, J. N. (2016). Comparative evaluation of
supervised learning algorithms for sentiment analysis of movie reviews. International Journal of Computer Applications, 142(1), 20-26.

[2] Pak, A., & Paroubek, P. (2010, May). Twitter as a corpus for sentiment analysis and opinion mining. In LREc (Vol. 10, No. 2010, pp. 1320-1326).

[3] Sisk, J. (2013). U.S. Patent Application No. 13/308,496.

[4] Deitrick, W., & Hu, W. (2013). Mutually enhancing community detection and sentiment analysis on twitter networks.

[5] Bengio, Y., Ducharme, R., Vincent, P., & Jauvin, C. (2003). A neural probabilistic language model. Journal of machine learning research, 3(Feb), 1137-1155.

[6] Chang, Y. W., Hsieh, C. J., Chang, K. W., Ringgaard, M., & Lin, C. J. (2010). Training and testing low-degree polynomial data mappings via linear SVM. Journal of Machine Learning Research, 11(4).

[7] Le, Q., & Mikolov, T. (2014, January). Distributed representations of sentences and documents. In International conference on machine learning (pp. 1188-1196).

[8] Peters, M. E., Neumann, M., Iyyer, M., Gardner, M., Clark, C., Lee, K., & Zettlemoyer, L. (2018). Deep contextualized word representations. arXiv preprint arXiv:1802.05365.

[9] Radford, A., Narasimhan, K., Salimans, T., & Sutskever, I. (2018). Improving language understanding by generative pre-training.

[10] Devlin, J., Chang, M. W., Lee, K., & Toutanova, K. (2018). Bert: Pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805.

[11] Wang, D., Cui, L., Wang, Y., Yuan, H., & Zhang, J. (2010, April). Association Rule Mining Based on Concept Lattice in Bioinformatics Research. In 2010 International Conference on Biomedical Engineering and Computer Science (pp. 1-4). IEEE.

[12] Liu, P., Wu, Q., Mu, X., Yu, K., & Guo, Y. (2015). Detecting the intellectual structure of library and information science based on formal concept analysis. Scientometrics, 104(3), 737-762.

[13] Hao, F., Min, G., Pei, Z., Park, D. S., & Yang, L. T. (2015). K-clique community detection in social networks based on formal concept analysis. IEEE Systems Journal, 11(1), 250-259.

[14] Zou, C., Zhang, D., Wan, J., Hassan, M. M., & Lloret, J. (2015). Using concept lattice for personalized recommendation system design. IEEE Systems Journal, 11(1), 305-314.

[15] Mikolov, T., Chen, K., Corrado, G., & Dean, J. (2013). Efficient estimation of word representations in vector space. arXiv preprint arXiv:1301.3781.

[16] Rong, X. (2014). word2vec parameter learning explained. arXiv preprint arXiv:1411.2738.

[17] Habibi, M., Weber, L., Neves, M., Wiegandt, D. L., & Leser, U. (2017). Deep learning with word embeddings improves biomedical named entity recognition. Bioinformatics, 33(14), i37-i48.

[18] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., ... & Polosukhin, I. (2017). Attention is all you need. In Advances in neural information processing systems (pp. 5998-6008).

[19] Chen, T., Xu, R., He, Y., & Wang, X. (2017). Improving sentiment analysis via sentence type classification using BiLSTM-CRF and CNN. Expert Systems with Applications, 72, 221-230.

[20] McCann, B., Bradbury, J., Xiong, C., & Socher, R. (2017). Learned in translation: Contextualized word vectors. In Advances in Neural Information Processing Systems (pp. 6294-6305).

[21] Xue, Q., Zhang, W., & Zha, H. (2020). Improving Domain-Adapted Sentiment Classification by Deep Adversarial Mutual Learning. In AAAI (pp. 9362-9369).
[22] Ito, T., Tsubouchi, K., Sakaji, H., Yamashita, T., & Izumi, K. (2020). Word-Level Contextual Sentiment Analysis with Interpretability. In AAAI (pp. 4231-4238).

[23] Shin, S. J., Song, K., & Moon, I. C. (2020). Hierarchically Clustered Representation Learning. In AAAI (pp. 5776-5783).

[24] Kumar, C. A., & Srinivas, S. (2010). Concept lattice reduction using fuzzy K-means clustering. Expert systems with applications, 37(3), 2696-2704.

[25] Kwuida, L., Missaoui, R., Balamane, A., & Vaillancourt, J. (2014). Generalized pattern extraction from concept lattices. Annals of Mathematics and Artificial Intelligence, 72(1-2), 151-168.

[26] Kim, J., Kim, K., & Kim, J. H. (2016). Semantic signature: comparative interpretation of gene expression on a semantic space. Computational and Mathematical Methods in Medicine, 2016.

[27] Dias, S. M., & Vieira, N. J. (2015). Concept lattices reduction: Definition, analysis and classification. Expert Systems with Applications, 42(20), 7084-7097.

[28] Shao, M., Liu, M., & Guo, L. (2013). Vector-based attribute reduction method for formal contexts. Fundamenta Informaticae, 126(4), 397-414.

[29] Kaytoue, M., Kuznetsov, S. O., Napoli, A., & Duplessis, S. (2011). Mining gene expression data with pattern structures in formal concept analysis. Information Sciences, 181(10), 1989-2001.

[30] Henriques, R., & Madeira, S. C. (2015, September). Pattern-based biclustering with constraints for gene expression data analysis. In Portuguese Conference on Artificial Intelligence (pp. 326-339). Springer, Cham.

[31] Kovács, L. (2019). An algorithm using context reduction for efficient incremental generation of concept set. Fundamenta Informaticae, 165(1), 43-73.

[32] Sellami, M., Hacid, M. S., & Gammoudi, M. M. (2019). A FCA framework for inference control in data integration systems. Distributed and Parallel Databases, 37(4), 543-586.

[33] Min, W. K., & Kim, Y. K. (2019). Soft concept lattice for formal concept analysis based on soft sets: Theoretical foundations and Applications. Soft Computing, 23(19), 9657-9668.

[34] Tiwari, A. K., Shreevastava, S., Som, T., & Shukla, K. K. (2018). Tolerance-based intuitionistic fuzzy-rough set approach for attribute reduction. Expert Systems with Applications, 101, 205-212.

[35] Soh, C., Yu, S., Narayanan, A., Duraismy, S., & Chen, L. (2019). Employee profiling via aspect-based sentiment and network for insider threats detection. Expert Systems with Applications, 135, 351-361.

[36] Cekik, R., & Uysal, A. K. (2020). A novel filter feature selection method using rough set for short text data. Expert Systems with Applications, 160, 113691.

[37] Warstadt, A., Singh, A., & Bowman, S. R. (2019). Neural network acceptability judgments. Transactions of the Association for Computational Linguistics, 7, 625-641.

[38] Qi, L., Hu, C., Zhang, X., Khosravi, M. R., Sharma, S., Pang, S., & Wang, T. (2020). Privacy-aware Data Fusion and Prediction with Spatial-Temporal Context for Smart City Industrial Environment. IEEE Transactions on Industrial Informatics. DOI: 10.1109/TII.2020.3012157.

[39] Cai, Z., & Zheng, X. (2020). A private and efficient mechanism for data uploading in smart cyber-physical systems. IEEE Transactions on Network Science and Engineering, 7(2): 766-775.

[40] Sun, Z., Wang, Y., Cai, Z., Liu, T., Tong, X., Jiang, N. A Two-stage Privacy Protection Mechanism Based on Blockchain in Mobile Crowdsourcing. International Journal of Intelligent Systems, 2021. DOI: 10.1002/int.22371.

[41] Wang, L., Zhang, X., Wang, R., Yan, C., Kou, H., & Qi, L. (2020). Diversified service
recommendation with high accuracy and efficiency. Knowledge-Based Systems, 204, 106196. DOI: 10.1016/j.knosys.2020.106196.

[42] Cai, Z., He, Z., Guan, X., & Li, Y. (2018). Collective data-sanitization for preventing sensitive information inference attacks in social networks. IEEE Transactions on Dependable and Secure Computing, 15(4), 577-590.

[43] Liu, T., Wang, Y., Li, Y., Tong, X., Qi, L., & Jiang, N. (2020). Privacy Protection based on Stream Cipher for Spatio-temporal Data in IoT. IEEE Internet of Things Journal. 7(9): 7928-7940.

[44] Kou, H., Liu, H., Duan, Y., Gong, W., Xu, Y., Xu, X., & Qi, L. (2020). Building trust/distrust relationships on signed social service network through privacy-aware link prediction process. Applied Soft Computing, 100, 106942. DOI: 10.1016/j.asoc.2020.106942.

[45] Wang, Y., Gao, Y., Li, Y., & Tong, X. (2020). A worker-selection incentive mechanism for optimizing platform-centric mobile crowdsourcing systems. Computer Networks, 171, 107144.

**Figure Title and Legend**

| Figure 1. the Framework of Concept-BERT |
|----------------------------------------|
| Figure 2. Reduce Concept Lattice by Association Rules |
| Figure 3. the Framework of BERT |
| Figure 4. Reduce Corpora by Concept-BERT |
| Figure 5. Learning Rate of Each Corpora |
| Figure 6. ACC-Epoch of Corpora |
| Figure 7. Attention Analysis of Concept-BERT |