Journal of Network Intelligence
Taiwan Ubiquitous Information
 ISSN 2414-8105
Volume **9**, Number **3**, August **2024**

Knowledge Graph-based Algorithm for Text Data Mining

Yu-Feng Zhao*

School of Electronics and IoT Engineering
Chongqing Industry Polytechnic College, Chongqing 401120, P. R. China
zhaoyf@cqipc.edu.cn

Jie He

Lee Kong Chian Faculty of Engineering and Science
Tunku Abdul Rahman University, Kuala Lumpur 50774, Malaysia
hejie@cqipc.edu.cn

*Corresponding author: Yu-Feng Zhao
Received August 23, 2023, revised December 20, 2023, accepted March 3, 2024.

Abstract. *Affinity Propagation (AP) clustering algorithm is a message-passing based clustering algorithm, which can adaptively determine the clustering structure of data samples by means of nearest-neighbour propagation and competitive learning.AP clustering algorithms are mainly applied in unsupervised clustering tasks, such as community discovery, text data mining and other fields. Although the implementation of AP clustering algorithm is relatively simple and there is no need to specify the number of clusters, it is ineffective for high-dimensional sparse data and is prone to local optimality. Therefore, an AP clustering algorithm based on Knowledge Graph is proposed for text data mining. Knowledge Graph is a kind of structured knowledge base for knowledge representation and computation, which can extract natural language text and structured data sources to build a greatly complex network. Firstly, Knowledge Graph is used for sample preprocessing, and the text to be clustered is analysed by Knowledge Graph ternary analysis, and a collection of samples corresponding to concepts, entities and relationships is generated. Then, the "semantic network" in the knowledge graph is constructed by improving the similarity measure between classes. Finally, by optimising the a priori constraints on the number of clusters and the distance calculation method between word vectors, the influence of isolated points on the clustering results is reduced. Comparison experiments show that compared with other existing clustering algorithms, the improved AP clustering algorithm is able to produce higher quality clustering results with less fluctuation.*
Keywords: data mining; text clustering; Affinity Propagation; knowledge graph;

1. **Introduction.** The difficulty of text data mining has been increasing due to the influence of many factors such as the number of texts, text structure and multilingual conformity symbols, while text clustering, as one of the core methods of text mining, has obvious advantages in the quality improvement of text mining [1, 2], especially contributing to the analysis of non-labelled data mining.

Through clustering, the relationship between massive heterogeneous multi-dimensional data can be effectively mined, discrete non-labelled data classification can be completed, data can be effectively categorized and organized, data availability can be improved, and technical support can be provided for the analysis of data models in various industries.

The feature extraction and disambiguation of text in the process of multi-class and multi-language fusion composite text clustering has a greater impact on the accuracy of text clustering [3, 4], so the feature preprocessing of the clustered samples is particularly important.

Cluster analysis is one of the most important techniques in the field of text data mining, which largely solves the problems caused by information explosion and clutter by organising and managing large amounts of text [5, 6]. Text clustering algorithms are unsupervised machine learning algorithms. The algorithms do not need to pre-train the data set, nor do they need to divide the source data documents into different labelled categories, which makes them more flexible and intelligent in data processing, and is of great help in improving the performance of information retrieval and search engines. By discovering the potential information of unstructured or semi-structured text collection data [7, 8], this information can help to organise and search a huge number of document collections, and better discover the intrinsic category characteristics in the collection. Not only that, text clustering can effectively reduce the waste and impact of manpower due to human factors during document processing, and reduce the possibility of errors due to human judgement [9, 10]. At the same time, it can also reduce the workload when performing category processing. The current clustering performance improvement research, on the one hand, from the perspective of platform deployment, through the super computing cloud platform and parallel technology to improve the efficiency of large-scale clustering, on the other hand, is to seek a stronger applicability of algorithms to improve the accuracy and stability of clustering, this paper's research focus is mainly on the latter.

Affinity Propagation (AP) algorithm is a parameter-free clustering algorithm based on the message passing mechanism [11], which iteratively updates the suitability of each sample to become a clustering centre by passing the two messages of responsibility and availability between the samples, automatically decides the number of clusters and finds the optimal clustering centre during the iterative process, and thus performs clustering on all the samples. The AP algorithm can be applied to image segmentation to perform semantic segmentation by clustering pixel information. Compared with K-means algorithm, AP algorithm can determine the number of clusters automatically. AP algorithm can be used for text clustering in topic detection, sentiment analysis and other tasks to discover text topic information [12]. In conclusion, as a parameter-free clustering method, AP algorithm can be widely used in clustering tasks that need to explore the intrinsic structure of the data, especially suitable for complex scenarios where the number of clusters is uncertain. However, the efficiency of the algorithm and the stability of the results still need to be improved.

Therefore, the research objective of this work is to effectively extract, integrate and evaluate the text features to be clustered with the help of knowledge graph technology to obtain more accurate and effective text features, and then realise text clustering through AP clustering algorithm.

1.1. **Related Work.** The Affinity Propagation (AP) algorithm, as a parameter-free clustering method, has some advantages in general, but there are some disadvantages or improvements that can be made as follows.

As a parameter-free clustering method, the nearest neighbor propagation algorithm has the advantages of automatically determining the number of clusters and searching for cluster centres, and it has been applied to a certain extent in the fields of image segmentation and data mining, etc. However, the disadvantages of poor algorithmic stability and high time and space complexity have limited its further development, and the current

research focuses on enhancing the robustness of the algorithm, improving the propagation mechanism, and speeding up the operation in order to improve the performance and scalability of the algorithm and make it a highly efficient and reliable clustering tool in practical applications. Wang et al. [13] proposed an adaptive similarity propagation mechanism to improve the robustness of the AP algorithm by dynamically adjusting the propagation matrix, and proved that the clustering accuracy of the improved algorithm is higher. Geng et al. [14] designed a similarity measure for non-spherical clusters, so that the AP algorithm can better handle the clustering of non-spherical shapes. Subedi et al. [15] designed an integrated clustering framework based on AP that incorporates multiple AP models, which can improve the stability of clustering and enhance the robustness to noise. However, the above studies have some following problems: (1) The choice of benchmark algorithms and indicators for comparison is relatively limited, and the persuasive power of evaluating the improvement effect is insufficient. (2) The improvement means are more limited, mostly in terms of similarity measure and accelerated calculation, without considering the integration of different algorithms.

Knowledge graph [16, 17] expresses the associated knowledge between concepts and events through a network of relationally connected entities, which is an important resource for realising knowledge management and intelligent analysis, as well as a knowledge base for semantic websearch and intelligent systems. Knowledge graph organises and expresses knowledge graphically, and its main components include entities, attributes and relationships. Entities are connected to each other through different relationships, constituting a greatly complex network [18, 19]. The use of graph database and association rules to organise knowledge has the advantages of direct navigation of relationships and rapid access to knowledge.

1.2. **Motivation and contribution.** Through the above analyses, it can be seen that the existing AP clustering algorithm still has room for improvement in terms of poor stability and high time and space complexity. The combination of knowledge graph and AP algorithm can promote each other and enhance the clustering effect, knowledge discovery and interpretation ability, which has a good application prospect.

Therefore, in order to solve the situation that AP clustering algorithm is prone to bias when dealing with high-dimensional sparsity data, this work introduces the knowledge graph into the AP clustering algorithm to optimise the method of calculating the distance between text objects. The improved algorithm is compared and analysed with the original algorithm in terms of clustering results using the evaluation function. The main innovations and contributions of this work include: (1) A text similarity metric based on knowledge graph is proposed. With the help of the graph structure characteristics of the knowledge graph, the similarity size between texts is calculated. The semantic similarity between its entity nodes is calculated through the path relationship between nodes in the hierarchical structure. In the constructed undirected bipartite graph based on semantic similarity, the value of conceptual distance is obtained by finding its maximum matching path.

(2) Propose to combine knowledge graph with AP clustering algorithm for text data mining. The knowledge of knowledge graph is fully utilised to achieve relationship-based sample representation, clustering interpretation, and iterative knowledge discovery. The semantic content is taken as a consideration for distance calculation, and the similarity distance between two entities is used to represent the semantic closeness between entities by using the structure function of the knowledge graph diagram.

2. Introduction to knowledge Graph.

2.1. **Technical structure.** The technical structure of knowledge graph is mainly through the classification of knowledge collection, through the extraction of knowledge units, and then through the integration and evaluation to obtain the knowledge graph, its main structure is shown in Figure 1.

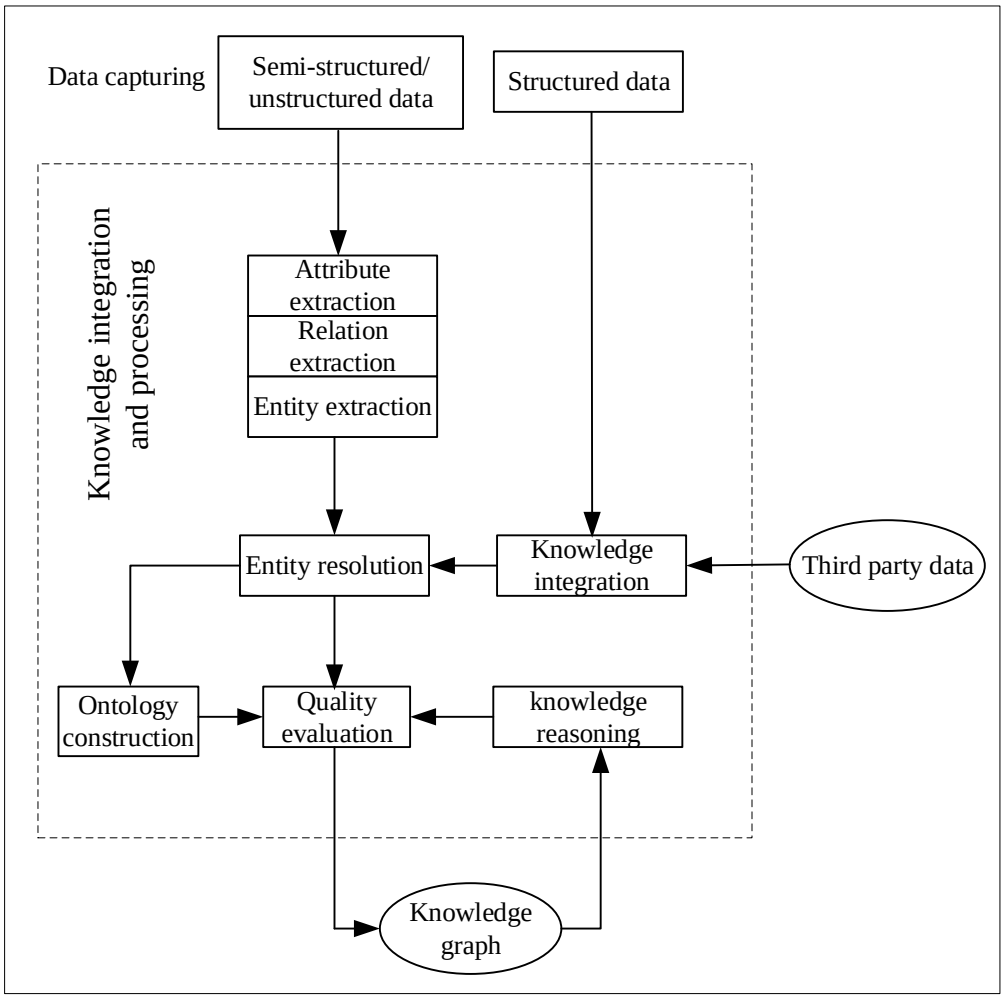


Figure 1. Knowledge Graph technology architecture

Knowledge extraction and knowledge fusion are two key techniques for constructing knowledge graphs [20, 21]. Knowledge extraction is mainly used to extract entities and relationships between entities from unstructured data sources such as text and web pages through natural language processing techniques to construct the initial knowledge graph. Knowledge extraction from various sources often has duplication, conflict and incompleteness problems, which need to be dealt with by knowledge fusion techniques. Knowledge fusion integrates and unifies the knowledge from different sources through entity alignment, relationship alignment, etc., and eliminates conflicts by using consistent reasoning, rule-based reasoning, etc. to normalise the knowledge and enhance the completeness, consistency and correctness of the knowledge graph. In addition, knowledge extraction based on statistical learning and deep learning is complemented by knowledge fusion based on symbolic logic and ontology theory. Knowledge reasoning technology further derives new knowledge based on existing knowledge and rules to achieve dynamic growth of knowledge graph. In summary, knowledge extraction, knowledge fusion and knowledge reasoning together drive the construction and evolution of knowledge graph.

2.2. **Working Principle.** Knowledge Graph is usually based on knowledge triples as the basic unit of knowledge expression, a triple consists of two entities and a connecting relationship, which represents a fact or a relational attribute between two entities. A large-scale triad constitutes a complex conceptual network. Knowledge graph adopts Concept, Entity, Relation and Attribute [22, 23] to represent knowledge, and the constituent elements of knowledge graph are knowledge elements, as shown in Figure 2.

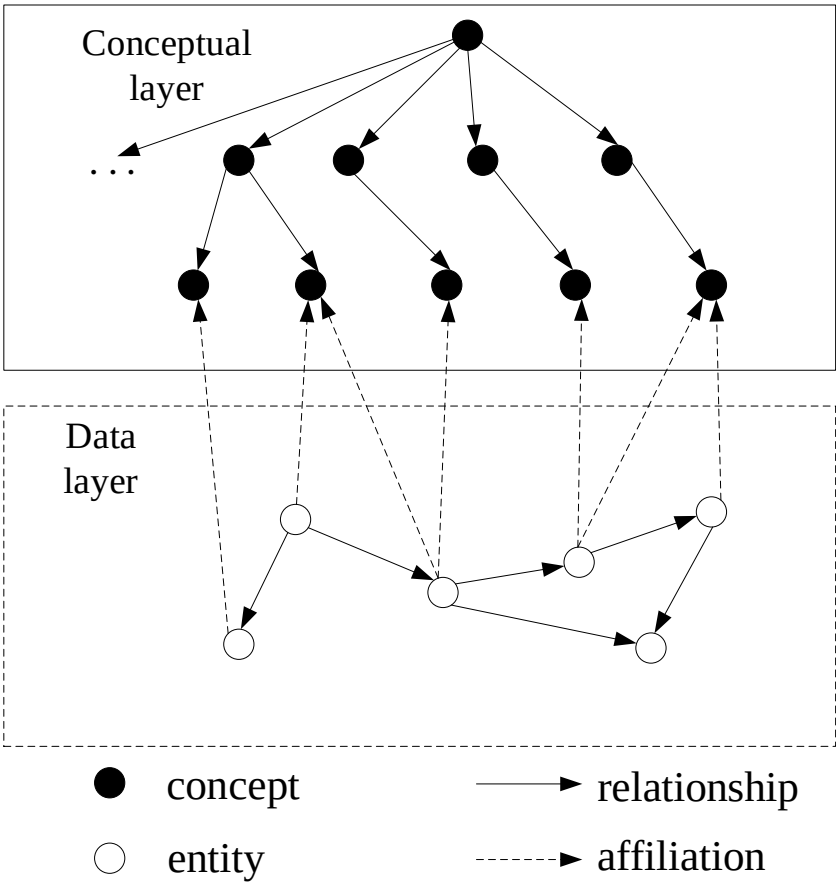


Figure 2. Structure of the knowledge graph

The set of knowledge elements within the knowledge domain d can be expressed as [24]: $KE_d = \{k_{e1}, k_{e2}, \ldots, k_{ei}, \ldots, k_{en}\}$, where the i-th element can be expressed as $k_{ei} = \{c_i, e_i, r_i, a_i\}$. c_i, e_i, r_i, and a_i denote conceptual knowledge, entity knowledge, relationship knowledge, and attribute knowledge, respectively.

The sets of concepts, entities, and relationships within the knowledge domain d are $C_d = \{c_1, c_2, \ldots, c_{nc}\}$, $E_d = \{e_1, e_2, \ldots, e_{ne}\}$, and $R_d = \{r_1, r_2, \ldots, r_{nr}\}$, where nc, ne, nr, and na represent the total number of concepts, entities, relationships, and attributes, respectively. The probability c_i can be expressed as $A_{c_i} = \{a_1, a_2, \ldots, a_{na}\}$ based on the attributes it contains.

The complex text is firstly classified into knowledge collections, followed by parsing of knowledge units, and finally extracting the knowledge elements and maps contained in the knowledge units, which are analyzed layer by layer to obtain the knowledge graph.

3. **Knowledge graph-based text similarity metrics.**

3.1. **Calculation of semantic similarity between entities.** The results of clustering algorithms depend on the calculation of distances or similarity metrics, so the choice of measures needs to be given more consideration. For text clustering, the distance between documents can be equated to the similarity between texts [25, 26].

Knowledge graphs transform specific things or abstract concepts into the form of directed graphs for easy observation and research. Logically, knowledge graphs can be divided into two layers: data layer and schema layer. The data stored in the data layer is composed of a certain number of facts, and the "knowledge" in the knowledge graph refers to these unit data volume of the "facts", you can choose such as the open source Neo4j. This paper uses the knowledge graph data with the help of Neo4j is used as a database to manage the network of relationships between entities.

The primary information contained in a text may be encapsulated in its words. In this paper, we consider that each word in a text can be regarded as an entity of the knowledge graph, and each entity in the text should be similar in the text set of the same category. The text dataset's entity storage structure is based on trees, and a knowledge graph structure is created when the entity tree structures of various categories are joined together. This structure may describe the relationship between entities more clearly. The similarity distance between two items often illustrates the semantic gap between them. In knowledge graph, different levels of tree structure will combine different categories of semantically related entities. In this paper, the path distance between entities in the tree structure is fully considered when performing the similarity calculation between entities in the knowledge graph. If two entities share a high degree of semantic similarity, their route distance will be proportionally low.

Suppose that entity $E = \{e_1, e_2, \ldots, e_n\}$ is a set of entities mapped to category $C = \{c_1, c_2, \ldots, c_n\}$, and the link length of entity e in category C is the sum of the path distances from the root node to that entity.

$$Path(e_i, e_j) = p_{e_i} + p_{e_j} \tag{1}$$

Where p_{e_i} is the path length as entity e_i and p_{e_j} is the path length as entity e_j.

Assuming that the entities e_i and e_j belong to the categories c_1 and c_2 respectively, the formula for calculating their semantic similarity between word sets can be obtained as shown below:

$$sim_{word}(e_i, e_j) = \frac{c_1 \cap c_2}{\max\{Path(e_i, e_j), 1\}} \tag{2}$$

3.2. **Similarity measure for fusing semantic and conceptual distances.** In the tree structure, the entity nodes in each layer represent a lexical entry, and the size of the distance between two entity nodes can be calculated using the word vector. Generally speaking, the words in a document may convey crucial data, and the semantic distances between entities inside the text should be close to one another within the same class cluster.

In this paper, to capture the semantic similarity of each entity in two documents, we create an undirected bipartite graph $G =< V, E >$, consisting of a vertex set V and an edge set E expressing semantic relations. In this paper, the Hungarian algorithm [27] is used to compute the above maximum matching problem between undirected bipartite graphs. It can be seen that the tree hierarchy in the knowledge graph is the key to the algorithm when considering the distance metric between entities. This hierarchy is the document hierarchy constructed by segmenting the acquired raw text, as the hierarchy is very useful for mining the internal structure of the document, which can be obtained either

by running a single-level text segmentation several times or by performing a hierarchical segmentation of the document.

Cosine similarity is used in the conventional AP clustering algorithm technique to determine how similar two text vectors are, and since each dimension of a text vector represents a word, this approach results in no connection between words that have the same meaning but different forms. For example, the connection between the words "tomato" and "tomatillo" will be ignored when calculating the cosine similarity. In order to solve this problem, after the text T_1 and T_2 are segmented, this paper uses named entity recognition and extracts the set of entities from the word set. In the hierarchical structure of the knowledge graph, given a node, each word in the content of the node is treated as a vertex in the vertex set V in the bipartite graph $G =< V, E >$.

For the dataset document D, it is divided into N parts, i.e., $D = \{p_1, p_2, \ldots, p_n\}$ using sentences or paragraphs as division boundaries. Clustering hierarchy is generated from actual text documents. In each hierarchy, a node is split into two, so the number of cluster classes increases at each level step by step [28]. The leaf level is the optimal clustering effect for text division, as shown in Figure 3.

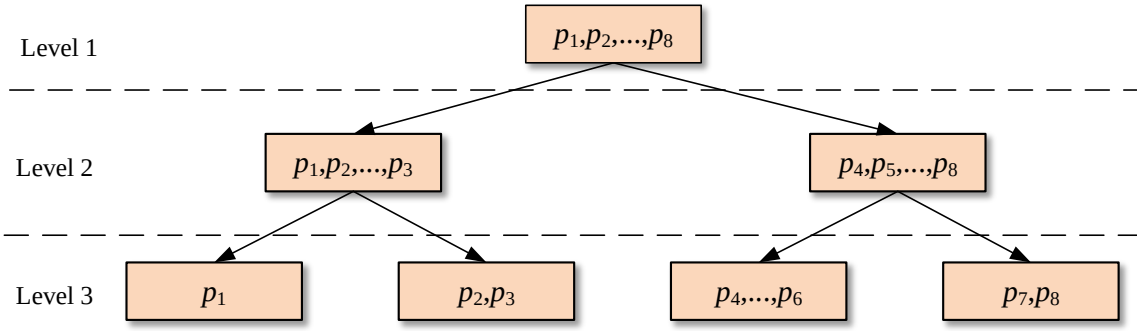


Figure 3. Examples of text partitions

Then the knowledge graph entity similarity is calculated in the dataset document D as shown below.

$$sim_{KG}(T_1, T_2) = \frac{|KB_1 \cap KB_2|}{|KB_1 \cup KB_2|} \tag{3}$$

Where KB_1 and KB_2 denote the set of knowledge graph entities corresponding to T_1 and T_2, respectively.

Combined with the word set similarity, the overall text similarity is obtained as follows.

$$sim(T_1, T_2) = \alpha sim_{KG}(T_1, T_2) + (1 - \alpha) sim_{word}(T_1, T_2) \tag{4}$$

Where α is the weight coefficient and sim_{word} is the word set similarity.

The pseudo-code of the knowledge graph-based text similarity metric is shown in Algorithm 1.

Among them, the Segment() function indicates that the text is segmented and returns the word collection; NER() function indicates that the named entity recognition is performed, and the entity is extracted from the word collection; WordSimilarity() function indicates that the similarity between the two word collections is computed, and it can be implemented by using the word vector model, etc. LinkToKG() function indicates that the entity is linked to the corresponding item in the knowledge graph, and the entity representation in the knowledge graph is returned. In this way, we can use the word similarity and the similarity of entity linking in the knowledge graph at the same time

Algorithm 1 Text similarity metrics based on knowledge graphs

Input: Text T_1, T_2, Knowledge graph KG
Output: $sim(T_1, T_2)$

$words_1 = Segment(T_1)$
$words_2 = Segment(T_2)$
$entities_1 = NER(words_1)$
$entities_2 = NER(words_2)$
$KB_1 = LinkToKG(entities_1, KG)$
$KB_2 = LinkToKG(entities_2, KG)$
$sim_{KG} = \frac{|KB_1 \cap KB_2|}{|KB_1 \cup KB_2|}$
$sim_{word} = WordSimilarity(words_1, words_2)$
$sim = \alpha * sim_{KG} + (1 - \alpha) * sim_{word}$
return sim

to get the text similarity which integrates multiple information. α value can be set by validation.

4. Knowledge graph-based text data mining.

4.1. AP clustering algorithm. AP algorithm is a parameter-free clustering algorithm based on a message passing mechanism.It iteratively updates the suitability of each sample to be a clustering centre by passing the two messages of responsibility and availability between the samples.It automatically decides the number of clusters and finds the best clustering centre during the iterative process to cluster all the samples.AP algorithm is a parameter-free clustering algorithm based on a message passing mechanism.AP algorithm is a parameter-free clustering algorithm based on a message passing mechanism.

The mathematical description of AP clustering is as follows [29, 30]: let 2 samples i and j to be clustered, the degree of similarity between them can be expressed as:

$$S(i,j) = -\|x_i - x_j\|^2 \tag{5}$$

The similarity values of any two sample points are solved according to the above equation to form the sample similarity matrix, where the diagonal elements are called the bias parameter P. The value of P has a large impact on the number of clustering categories and is highly sensitive to the impact on the clustering performance in practical applications.

Let $r(i,j)$ and $a(i,j)$ denote the attractiveness function and affiliation function of samples i and j respectively. The degree of similarity between samples i and j is proportional to the value of $r(i,j)+a(i,j)$. Solve for the attractiveness and affiliation of any 2 samples by forming the matrices $R = [r(i,j)]_{N\times N}$ and $A = [a(i,j)]_{N\times N}$.

$$r(i,j) = s(i,j) - \max_{j\neq i}\{a(i,j) + s(i,j)\} \tag{6}$$

$$a(i,j) = \min\{0, r(j,j) + \sum_{i\neq j}\max\{0, r(i,j)\}\} \tag{7}$$

Where $r(i,j)$ is the self-attractiveness of node j, j denotes other sample points except j, and i denotes other sample points.

When $i = j$, Equation (7) becomes:

$$a(j,j) = \sum_{i\neq j}\max\{0, r(i,j)\} \tag{8}$$

Adding $a(i,j)$ to both sides of Equation (6) gives:

$$r(i,j) + a(i,j) = s(i,j) + a(i,j) - \max_{j \neq i}\{a(i,j) + s(i,j)\} \tag{9}$$

Let $E = [e(i,j)]_{N\times N} = [r(i,j) + a(i,j)]_{N\times N}$ be called the decision matrix, then $\Gamma = [\tau(i,j)]_{N\times N} = [s(i,j) + a(i,j)]_{N\times N}$ is the potential matrix.

$$e(i,j) = \tau(i,j) - \max_{j \neq i}\{\tau(i,j)\} \tag{10}$$

Solve for the maximum value of $e(i,j)$ to obtain the maximum degree of similarity between the sample points. The higher the degree of similarity between the sample points and the center of a cluster, the more the node belongs to that cluster. Solving the $e(i,j)$ maximum value of each node to each cluster center point one by one, the clustering category of all the points is obtained.

4.2. **Knowledge graph-based AP clustering algorithm.** This paper proposes to combine knowledge graph with AP clustering algorithm for text data mining. The knowledge of knowledge graph is fully utilised to achieve relationship-based sample representation, clustering interpretation, and iterative knowledge discovery. The semantic content is taken as a consideration for distance calculation [31], and the similarity distance between two entities is used to represent the semantic closeness between entities by using the structure function of the knowledge graph diagram. The steps of knowledge graph based AP clustering algorithm are shown below:

Step 1: Construct the knowledge graph, including entities, attributes, and relationships. Entity can be the object of sample data.

Step 2: Use the knowledge graph for relational reasoning to expand the feature representation of each sample into enhanced samples.

Step 3: Calculate the similarity between the augmented samples and construct the similarity matrix. We can consider fusing the similarity measures based on attributes and relationships.

Step 4: Input the similarity matrix into AP algorithm for cluster centre selection and sample clustering.

Step 5: Obtain which types of entities in the knowledge graph the samples of each cluster belong to, as a clustering explanation.

Step 6: According to the clustering results, construct class-instance relationship in the knowledge graph, and integrate the results into the knowledge graph.

Step 7: Iteratively refine the knowledge graph and similarity matrix, repeat Step3 to Step6 to achieve progressive clustering.

Step 8: Construct sample-clustering relationship in the knowledge graph to form enhanced knowledge graph.

This can make full use of the knowledge of the knowledge graph to achieve relationship-based sample representation, cluster interpretation, and iterative knowledge discovery. The pseudo-code of AP clustering algorithm based on knowledge graph is shown in Algorithm 2.

Where A_i denotes the set of added attributes obtained after knowledge graph inference for sample x_i and R_i denotes the set of added relationships obtained after knowledge graph inference for sample x_i.

For the extraction of the sample matrix, a data structure based on the dissimilarity matrix is used in the AP clustering algorithm. The dissimilarity matrix represents the

Algorithm 2 Knowledge graph-based AP clustering algorithm

Input: sample set X, Knowledge graph KG
Output: sample clustering results $clusters$; cluster centre samples $exemplars$

```
# Building the Knowledge Graph KG
KG = BuildKG(entities, attributes, relations)
# Intellectual reasoning
for sample x_i in X do
    (A_i, R_i) = KGReasoning(x_i, KG)
end for
# Calculate the similarity
for x_i, x_j in X do
    sim_ij = ω * Sim_attr(A_i, A_j) + (1 − ω) * Sim_rel(R_i, R_j)
    sim_matrix[i, j] = sim_ij
end for
# AP clustering
(exemplars, clusters) = AP(sim_matrix)
# Typological reasoning
for c_i in clusters do
    tc_i = ArgmaxType(c_i, KG)
    # Increase knowledge
    KG.add((tc_i, 'has_instance', x_k), x_k ∈ c_i)
    # Update the similarity
    sim = λ * sim + (1 − λ) * sim_prev
end for
# Knowledge representation
for x_j in X do
    KG.add((x_j, 'belongs_to', tc_i))
end for
```

matrix of the degree of dissimilarity between two of the n data sample points.

$$\begin{bmatrix} 0 & & & \\ d(2,1) & 0 & & \\ d(3,1) & d(3,2) & 0 & \\ \vdots & \vdots & \vdots & \vdots \\ d(n,1) & d(n,2) & \cdots & 0 \end{bmatrix} \tag{11}$$

Where $d(i,j)$ indicates the numerical magnitude of the dissimilarity between object i and object j. The more similar object i is to object j, the closer this value is to 0. If object i is more dissimilar to object j, then the value of $d(i,j)$ will be large.

5. Experimental results and analyses.

5.1. **Experimental dataset.** The experimental dataset of this paper is from Wikipedia Chinese text dataset. The dataset is divided into five categories: public health, social welfare, accidents and disasters, types of complaints and natural disasters. In this paper, about 200 texts from each category are randomly selected as the experimental sample set. When performing clustering training, the ratio of the number of training and testing numbers for the four sample sets is 3:1. The number of texts in each category is shown in Table 1.

Table 1. Sample set of experimental data

Set	Number of documents	Number of entries	Category
public health	202	120659	10
social welfare	380	251125	11
accident and disaster	477	173182	12
Type of complaint	336	27262	10
natural disaster	206	82665	14

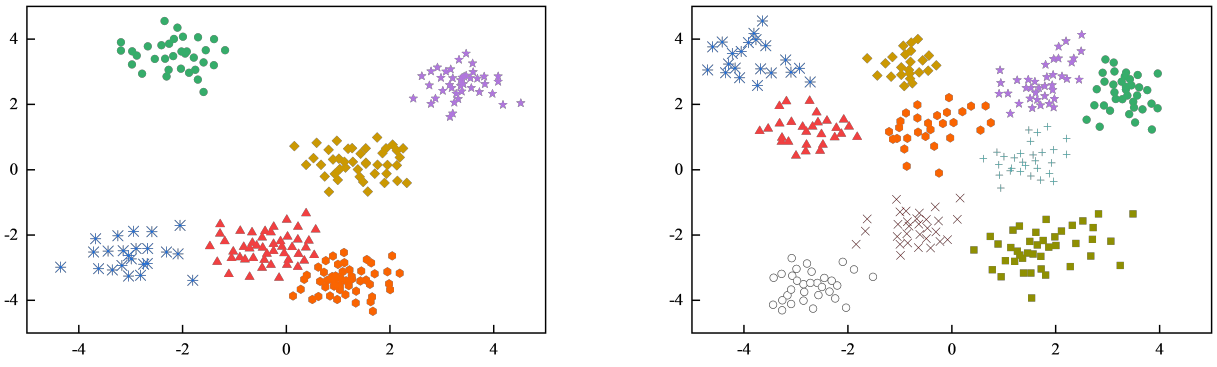
5.2. **Selection of Weighting Parameter α.** In this paper, a set of comparative experiments is used to get the determination of the weighting parameter α to find the parameter value that gets the best clustering effect, and the experimental results are shown in Table 2.

Table 2. Clustering results for different values of the parameter α

α	0.30	0.35	0.40	0.45	0.50	0.55	0.60	0.65	0.70
RI	0.411	0.457	0.503	0.464	0.590	0.760	0.642	0.525	0.481
MI	0.640	0.550	0.641	0.515	0.821	0.820	0.763	0.633	0.543

As can be seen from Table 2, when the value of parameter α is in the range of [0.30, 0.70], the evaluation indexes RI (Rand index) and MI (Mutual Information) of AP clustering results will get different results. Better results can be achieved at $\alpha = [0.45, 0.60]$. At $\alpha = 0.55$, the mean value of RI and MI obtained is optimal, so in the subsequent experiments this paper takes the value of the weight parameter α as 0.55 when calculating the fusion distance between entities.

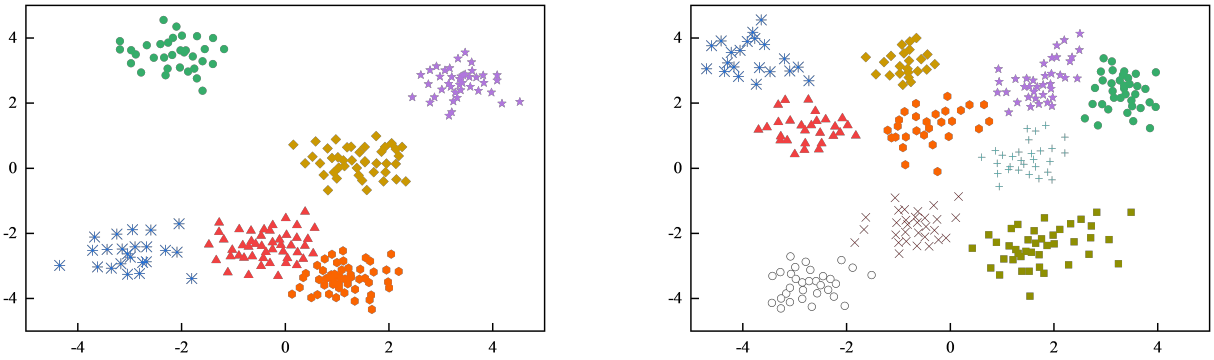
5.3. **Effect of knowledge Graph on AP clustering performance.** In order to verify the effect of knowledge graph on Ap algorithm, AP algorithm and KG-AP algorithm were used to simulate the performance of the dataset in Table 1 and visualise the clustering results, respectively. Here, only the visualisation results of public health and social welfare are selected for presentation, as shown in Figure 4 and Figure 5.



(a) AP algorithm (b) KG-AP algorithm

Figure 4. Cluster visualisation of public health sample sets

It can be seen that after the knowledge graph processing, the number of categories mined is closer to the actual value for the public health sample set and the social welfare sample set. Before the knowledge graph analysis, the number of categories for the public health sample set and the social welfare sample set are 6 and 6 respectively, which are



(a) AP algorithm (b) KG-AP algorithm

Figure 5. Cluster visualisation of a sample set of social benefits

much smaller than the actual categories of 10 and 11. Therefore, the addition of the knowledge graph greatly reduces the error of the AP clustering, which indicates that the clustering accuracy of the AP algorithm is significantly improved after the textual data is analysed by the knowledge graph.

5.4. **Comparative analysis of clustering results.** The commonly used text clustering algorithms K-means [32], K-modes [33], AP and KG-AP are used to simulate the data in Table 1 respectively, and their average clustering accuracies are counted, and the results are shown in Table 3.

Table 3. Accuracy of 4 clustering algorithms

Sample set	K-means	k-modes	AP	DE-AP
public health	0.7329	0.7833	0.8847	0.9437
social welfare	0.7247	0.7217	0.8414	0.9213
accident and disaster	0.7271	0.7634	0.8611	0.9241
Type of complaint	0.7512	0.7875	0.8917	0.9616
natural disaster	0.7175	0.7129	0.8339	0.9147

It can be seen that in terms of the clustering accuracy of the four sample sets, the KG-AP algorithm has the best performance, with clustering accuracy above 0.9 for all of them, followed by the AP algorithm, while the K-means and K-modes algorithms are worse, with both below 0.8. While comparing the data sets horizontally, four of these algorithms have the highest text clustering accuracy in the complaint type sample set, while the natural disaster sample set has the worst clustering accuracy, which may be related to the number of attributes contained in the sample set. When there are more attributes, the difficulty of data clustering climbs and the accuracy rate decreases accordingly, and the number of attributes in the natural disaster sample set is significantly higher than that in the complaint type sample set.

In order to validate the effectiveness of the KG-AP clustering algorithm proposed in this paper, four common evaluation metrics are taken to comprehensively analyse the performance of the four clustering algorithms, and the results are shown in Tables 4, 5, 6 and 7, respectively.

In these five types of document clustering, the overall trend of RI tends to be closer to 1, showing better clustering results. Although the results of AP and KG-AP are close to each other in the categories of public health and natural disasters where the number of

Table 4. Rand index (RI) for the four clustering algorithms

Sample set	K-means	k-modes	AP	KG-AP
public health	0.505	0.394	0.505	0.691
social welfare	0.611	0.448	0.454	0.702
accident and disaster	0.597	0.535	0.477	0.751
Type of complaint	0.808	0.495	0.515	0.854
natural disaster	0.711	0.380	0.571	0.712

Table 5. Silhouette Coefficient (SC) for the 4 clustering algorithms

Sample set	K-means	k-modes	AP	KG-AP
public health	0.294	0.186	0.297	0.611
social welfare	0.344	0.194	0.396	0.526
accident and disaster	0.292	0.335	0.277	0.612
Type of complaint	0.208	0.223	0.275	0.654
natural disaster	0.211	0.265	0.320	0.748

Table 6. Mutual Information (MI) of the 4 clustering algorithms

Sample set	K-means	k-modes	AP	KG-AP
public health	0.611	0.386	0.507	0.801
social welfare	0.635	0.404	0.598	0.816
accident and disaster	0.692	0.396	0.531	0.871
Type of complaint	0.508	0.493	0.591	0.871
natural disaster	0.593	0.365	0.569	0.865

Table 7. V-measure of the four clustering algorithms

Sample set	K-means	k-modes	AP	KG-AP
public health	0.691	0.644	0.704	0.885
social welfare	0.735	0.660	0.712	0.873
accident and disaster	0.592	0.680	0.742	0.899
Type of complaint	0.608	0.672	0.597	0.847
natural disaster	0.683	0.677	0.665	0.891

documents is small, KG-AP still achieves good accuracy and is slightly higher than AP when confronted with a larger number of data word counts.

In a data set, the value of SC is in the range of [-1,1]. If the result is closer to the value 1, then it means that entities of the same category are closer together and entities between different categories are more distant. In the evaluation results of this metric, the results of the four clustering methods are significantly lower, reflecting the unsatisfactory effect of cohesion and separation between clusters. In the two metrics of mutual information and V-measure, the value of KG-AP algorithm is higher than the other previous distance metric algorithms, which achieves a better clustering effect.

6. **Conclusion.** In order to solve the situation that AP clustering algorithm is prone to bias when dealing with high-dimensional sparsity data, this work introduces knowledge graph into AP clustering algorithm to optimise the method of distance calculation between text objects. Taking semantic content as a consideration for distance calculation, the similarity distance between two entities is used to represent the semantic closeness

between entities using the structure function of the knowledge graph graph. The evaluation function was used to compare and analyse the KG-AP clustering algorithm with other clustering algorithms in terms of clustering results. Four evaluation metrics such as RI, SC, MI and V-measure were used to compare and evaluate the experimental results. The experimental results show that the KG-AP clustering algorithm has a significant improvement in terms of accuracy and performance. A follow-up study will attempt to analyse the interpretability of the clustering results and justify the parameter selection and convergence.

Acknowledgement. This work supported by ministry of education's college student department's supply and demand matching employment and education project (No. 20230103588), project of the informationization education guidance committee of the ministry of education (No. KT22620), and science and technology research project of chongqing education commission (No. KJQN202203213).

REFERENCES

[1] L. Hickman, S. Thapa, L. Tay, M. Cao, and P. Srinivasan, "Text preprocessing for text mining in organizational research: Review and recommendations," *Organizational Research Methods*, vol. 25, no. 1, pp. 114-146, 2022.

[2] T.-Y. Wu, J. C.-W. Lin, U. Yun, C.-H. Chen, G. Srivastava, and X. Lv, "An efficient algorithm for fuzzy frequent itemset mining," *Journal of Intelligent & Fuzzy Systems*, vol. 38, no. 5, pp. 5787-5797, 2020.

[3] T.-Y. Wu, J. Lin, Y. Zhang, and C.-H. Chen, "A Grid-Based Swarm Intelligence Algorithm for Privacy-Preserving Data Mining," *Applied Sciences*, vol. 9, no. 4, pp. 774, 2019.

[4] A. K. Kushwaha, A. K. Kar, and Y. K. Dwivedi, "Applications of big data in emerging management disciplines: A literature review using text mining," *International Journal of Information Management Data Insights*, vol. 1, no. 2, 100017, 2021.

[5] L. Chen, W. Gan, Q. Lin, S. Huang, and C.-M. Chen, "OHUQI: Mining on-shelf high-utility quantitative itemsets," *The Journal of Supercomputing*, vol. 78, no. 6, pp. 8321-8345, 2022.

[6] C.-M. Chen, L. Chen, W. Gan, L. Qiu, and W. Ding, "Discovering high utility-occupancy patterns from uncertain data," *Information Sciences*, vol. 546, pp. 1208-1229, 2021.

[7] W. Gan, L. Chen, S. Wan, J. Chen, and C.-M. Chen, "Anomaly Rule Detection in Sequence Data," *IEEE Transactions on Knowledge and Data Engineering*, 2022. [Online]. Available: https://doi.org/10.1109/TKDE.2021.3139086

[8] F. Galati, and B. Bigliardi, "Industry 4.0: Emerging themes and future research avenues using a text mining approach," *Computers in Industry*, vol. 109, pp. 100-113, 2019.

[9] S. Chen, X. Guo, T. Wu, and X. Ju, "Exploring the online doctor-patient interaction on patient satisfaction based on text mining and empirical analysis," *Information Processing & Management*, vol. 57, no. 5, 102253, 2020.

[10] M. Pejić Bach, Ž. Krstić, S. Seljan, and L. Turulja, "Text mining for big data analysis in financial sector: A literature review," *Sustainability*, vol. 11, no. 5, pp. 1277, 2019.

[11] U. Bodenhofer, A. Kothmeier, and S. Hochreiter, "APCluster: an R package for affinity propagation clustering," *Bioinformatics*, vol. 27, no. 17, pp. 2463-2464, 2011.

[12] C.-D. Wang, J.-H. Lai, C. Y. Suen, and J.-Y. Zhu, "Multi-exemplar affinity propagation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 9, pp. 2223-2237, 2013.

[13] J. Wang, Y. Gao, K. Wang, A. K. Sangaiah, and S.-J. Lim, "An affinity propagation-based self-adaptive clustering method for wireless sensor networks," *Sensors*, vol. 19, no. 11, 2579, 2019.

[14] Z. Geng, R. Zeng, Y. Han, Y. Zhong, and H. Fu, "Energy efficiency evaluation and energy saving based on DEA integrated affinity propagation clustering: Case study of complex petrochemical industries," *Energy*, vol. 179, pp. 863-875, 2019.

[15] S. Subedi, H.-S. Gang, N. Y. Ko, S.-S. Hwang, and J.-Y. Pyun, "Improving indoor fingerprinting positioning with affinity propagation clustering and weighted centroid fingerprint," *IEEE Access*, vol. 7, pp. 31738-31750, 2019.

[16] X. Chen, S. Jia, and Y. Xiang, "A review: Knowledge reasoning over knowledge graph," *Expert Systems with Applications*, vol. 141, 112948, 2020.

[17] S. Ji, S. Pan, E. Cambria, P. Marttinen, and S. Y. Philip, "A survey on knowledge graphs: Representation, acquisition, and applications," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 33, no. 2, pp. 494-514, 2021.
[18] Q. Guo, F. Zhuang, C. Qin, H. Zhu, X. Xie, H. Xiong, and Q. He, "A survey on knowledge graph-based recommender systems," *IEEE Transactions on Knowledge and Data Engineering*, vol. 34, no. 8, pp. 3549-3568, 2020.
[19] Y. Dai, S. Wang, N. N. Xiong, and W. Guo, "A survey on knowledge graph embedding: Approaches, applications and benchmarks," *Electronics*, vol. 9, no. 5, pp. 750, 2020.
[20] C. Dreisbach, T. A. Koleck, P. E. Bourne, and S. Bakken, "A systematic review of natural language processing and text mining of symptoms from electronic patient-authored text data," *International Journal of Medical Informatics*, vol. 125, pp. 37-46, 2019.
[21] A. Onan, "Sentiment analysis on massive open online course evaluations: a text mining and deep learning approach," *Computer Applications in Engineering Education*, vol. 29, no. 3, pp. 572-589, 2021.
[22] F. Greco, and A. Polli, "Emotional Text Mining: Customer profiling in brand management," *International Journal of Information Management*, vol. 51, 101934, 2020.
[23] X. Xie, Y. Fu, H. Jin, Y. Zhao, and W. Cao, "A novel text mining approach for scholar information extraction from web content in Chinese," *Future Generation Computer Systems*, vol. 111, pp. 859-872, 2020.
[24] A. Subakti, H. Murfi, and N. Hariadi, "The performance of BERT as data representation of text clustering," *Journal of Big Data*, vol. 9, no. 1, pp. 1-21, 2022.
[25] R. Janani, and S. Vijayarani, "Text document clustering using spectral clustering algorithm with particle swarm optimization," *Expert Systems with Applications*, vol. 134, pp. 192-200, 2019.
[26] D. Wu, R. Yang, and C. Shen, "Sentiment word co-occurrence and knowledge pair feature extraction based LDA short text clustering algorithm," *Journal of Intelligent Information Systems*, vol. 56, pp. 1-23, 2021.
[27] Z. Li, H. Liu, Z. Zhang, T. Liu, and N. N. Xiong, "Learning knowledge graph embedding with heterogeneous relation attention networks," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 33, no. 8, pp. 3961-3973, 2021.
[28] A. Rossi, D. Barbosa, D. Firmani, A. Matinata, and P. Merialdo, "Knowledge graph embedding for link prediction: A comparative analysis," *ACM Transactions on Knowledge Discovery from Data*, vol. 15, no. 2, pp. 1-49, 2021.
[29] S. K. Mohamed, V. Nováček, and A. Nounu, "Discovering protein drug targets using knowledge graph embeddings," *Bioinformatics*, vol. 36, no. 2, pp. 603-610, 2020.
[30] D. N. Nicholson, and C. S. Greene, "Constructing knowledge graphs and their biomedical applications," *Computational and Structural Biotechnology Journal*, vol. 18, pp. 1414-1428, 2020.
[31] B. Abu-Salih, "Domain-specific knowledge graphs: A survey," *Journal of Network and Computer Applications*, vol. 185, 103076, 2021.
[32] T. M. Ghazal, "Performances of K-means clustering algorithm with different distance metrics," *Intelligent Automation & Soft Computing*, vol. 30, no. 2, pp. 735-742, 2021.
[33] M. Mahmoodi, S. H. Tavassoli, O. Takayama, J. Sukham, R. Malureanu, and A. V. Lavrinenko, "Existence conditions of high-k modes in finite hyperbolic metamaterials," *Laser & Photonics Reviews*, vol. 13, no. 3, 1800253, 2019.