# **Short text mining method based on sub-semantic space**

**Abstract:** In order to solve the problem of accurately identifying short text data, a short text strategy mining method based on sub-semantic space is proposed. This method first uses semantic space technology to solve the "vocabulary gap" and "data sparseness" in the analysis of short text; Then, the semantic space is divided into multiple sub-semantic spaces based on the clustering algorithm, and association rules are mined in each sub-semantic space in parallel, which improves the efficiency and quality of policy generation; Afterwards, the binary tree is used to merge strategies to generate a simple strategy set. Experiments show that, compared with the traditional classification model, the strategy set generated by this scheme can achieve an accuracy rate of **xx%** when the false positive rate is xx%. In the process of discovering and violating short messages, the strategy set mined by this technology has strong coverage, high accuracy, and strong practicability.

**Keywords:** sub-semantic space; strategy extraction; short text; association rule mining; clustering

**1 Introduction**

Short text strategy extraction algorithm is to extract keywords from massive short text data and generate association relationship. At present, the technical research on strategy extraction is mainly divided into 3 categories: Statistical-based methods, word network based methods and Natural language methods based on semantics and ontology.

* Statistics-based methods, such as the calculation methods based on TF-IDF features proposed in reference [1], this kind of method has the generalization of the model, but ignores the internal connection of lexical semantics, such as the N-gram model, which is limited to the computing power. At present, most of them are bi-gram and tri-gram. Reference [2] introduced the Markov model on the basis of the N-element model, adding improved model with added cache part, and the characteristics of the short text itself lead to a poor application effect.
* Based on the word network method, the document is mapped to the word network according to certain rules, and the word network is used to calculate the criticality of the word. The keyword extraction algorithm “TextRank” based on the co-occurrence network in reference [3] builds co-occurrence through the word co-occurrence window Network, and calculate the score of words according to the co-occurrence network; The word co-occurrence graph model in reference [4] uses the co-occurrence intensity between words as the basis for graph division, at this time, each cluster corresponds to a connected subgraph of a basic point of view, which constitutes a specific topic. In this way, the words in the cluster form several sets of strategic phrases. The disadvantage of this method is that in the case of massive data, the word network cannot get clear and effective clusters.
* The method of natural language understanding based on semantics and ontology, using the relationship between synonyms and synonyms, such as based on "HowNet" [5] and the "Synonym Cilin Extended Edition" revised by the Information Retrieval Laboratory of Harbin Institute of Technology. This method is limited by the completeness of the dictionary itself.

This paper proposes a method for mining short text strategies based on sub-semantic space, which is used in the process of identifying illegal SMS, constructing a semantic space and generating strategy sets based on this association rule mining. The co-occurrence relationship between vocabulary can mine the potential correlation between vocabulary and has good strategy expansion ability. At the same time, the algorithm based on machine learning can automatically learn and tune, which improves the accuracy of judgment and reduces the omission rate.

**2 Technical foundation**

***2.1 Semantic space***

In order to calculate the relationship between vocabulary, the traditional method uses "bag of words" to map the vocabulary into points in a vector space (VSM), the dimension of VSM is *equal* to the total number of words N, it will easily reach tens of thousands or millions of dimensions. Moreover, due to its own characteristics such as fewer words and random expressions, short text highlights two serious problems in the process of mining strategies: the "vocabulary gap" problem. During the analysis of the text, each word is converted into a "bag of words" and mapped to Points of the space vector, the expression is very random, and the similarity between words cannot be well portrayed; The problem of "data sparseness" is that the short text has a small number of words and when expressed by keywords, each data has only a few keywords, which will cause serious data sparseness during the calculation process. For example, "Apple mobile phone" and "iPhone" have similar meanings, and the vocabulary association becomes weaker after being mapped into the vector space:

* "Apple phone" is expressed as [0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0… 0] 1 × N;
* "IPhone" is represented as [0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0… 0] 1 × N.

In order to solve the above problem, the concept of semantic space, also called word vector space [6], is introduced, in which the semantic space can be set to a fixed M dimension, the vocabulary is converted into M-dimensional word vector representation, which not only greatly reduces the vector space dimension and reduces the amount of calculation, but also calculates the vocabulary relevance from the semantic level, which improves the accuracy of analysis. The essence of the semantic space [7-8] is: change each element of the vector from an integer to a floating-point number to a representation of the entire real number range; the original sparse, huge-dimensional vector space is compressed into a dense, low-dimensional semantic space. The above words "Apple phone" and "iPhone", the word vector in the M (eg M = 200) dimension semantic space is expressed as follows:

* "Apple phone" is expressed as [0.01 0.07 0.96 0.32 0.24…] 1 × M;
* "IPhone" is expressed as [0.02 0.24 0.84 0.45 0.01 ...] 1 × M.

***2.2 Association rule mining***

Association rule mining [9] is a classic unsupervised learning algorithm for data mining. It can find the interdependence and association between item-sets in transactional data sets. Attributes can be predicted from other attribute values. For example, in the classic case "beer and diapers", people who buy beer will buy diapers with a certain probability, {beer} → {diaper} is an association rule.

* Item set: The indivisible small unit information in the database is called item, which is represented by symbol i; the item set is called item set, which is represented by I.
* Transactions: All data are collectively called transactions, and each transaction is a non-empty item set.
* Association rule: the implication of association rule x → y, where x and y are the true subsets of I, and x ∩ y = φ, x is called the premise, and y is the result. The association rule reflects the rule that the items in x appear, and the items in y follow.
* Item set support: The number of transactions including the item set is called the item set support count (the frequency of the item set).
* Min support: The small number of items that appear in all training tuples at the same time.
* Support for association rules (support): The support for association rules is the ratio of the union of transactions containing x and y in the transaction set to all transactions:

(1)

* Confidence of association rules: The confidence of association rules is the ratio of the transaction set containing both x and y to all transactions containing x:

(2)

the association rule mining problem can be divided into two sub-problems:

* Find all frequent itemsets in the database that are greater than or equal to the specified small support level;
* The association rules required for frequent item set generation are used, and strong association rules are selected according to the set small credibility.

**3. Illegal SMS words Mining strategy based on sub-semantic space**

Based on sub-semantic space mining strategy method, short text strategy extraction consists of two steps: keyword extraction and association relationship combination. First extract keywords from short text samples, and then determine the association relationship (AND, OR, and non-relationship) of these keywords to form a strategy, use these mining strategies to identify more short texts targets, Herein, which was used in the process of identifying violations of messages.

The main process is divided into 5 steps, and the system flow is shown in Figure 1.

SMS data

SMS preprocessing

Strategic Assessment

Keyword extraction

Word3Vec Calculate word vector

Clustering

Subsemantic space 2

Subsemantic space N

Subsemantic space 1

Mining Association Rules

Mining Association Rules

Mining Association Rules

Generating Association Rules

Strategically fit

Final Strategy set

Abandon

***Strategy set generation system flow***

* Step 1 Pre-process the SMS sample and extract keywords;
* Step 2 Construct a semantic space of keywords, and cluster based on the semantic space to generate a sub-semantic space;
* Step 3 Mining each sub-semantic space in parallel to generate association rules with obvious relationships and potential relationships;
* Step 4 Use training corpus to evaluate association rule generation strategy;
* Step 5 Build a binary tree compression strategy and merge strategy to form a streamlined strategy set.

***3.1 Step 1***

*(1) SMS preprocessing*

The main function of SMS pre-processing [10] is to normalize the SMS samples containing interference items, such as traditional characters, numeric sequences, interference symbols, etc., and convert the original information samples into unified text according to the matching conversion rules sequence.

The sample text message is:

This # Company \* issues various Fa tickets on behalf of the company, engraves the official seal on behalf of it, ITU 13② 88057 & 898;

Your mobile number has been selected as the winner of "दीपावली भाग्योदय", Get 1Lac Rs. bonus and log in to tryyourluck.com Verification code: 7768 to receive the price.

The processing method is as follows:

Handling numbers: convert according to the number comparison table, and then extract the relatively continuous number sequence according to the context, such as Lac → 6, ② → 2

Dealing with interfering characters: remove the special symbols such as #, &,%, α, β, я, ы, щ, ■, ●, ★, and ※ in the message to be tested;

Dealing with Local Language: Provide a comparison table of local language to convert to English, such as भाग्योदय → Lucky Draw.

*(2) Word segmentation and remove stop words and useless words*

Perform the following operations on the preprocessed SMS:

First use the word segmentation tool to generate word segmentation and part-of-speech tagging results; then delete the stop words and useless words according to the stop word list. The words that can really express the meaning of the sentence are nouns, nominal phrases, verbs and verb phrases, etc. Phone numbers, bank card numbers, etc. have less significance in strategy extraction, and can be identified and used in a unified vocabulary through regular methods after replacement, only the above four parts of speech vocabulary will be retained.

*(3) Keyword extraction*

The purpose of this method is to eliminate the words of ultra-high frequency and ultra-low frequency, and the extraction accuracy of keywords need not be too high. You can use TF-IDF to calculate the lexical weight. TF-IDF is a statistical method used to evaluate the importance of vocabulary to the corpus. The vocabulary appears frequently in one article and rarely appears in other articles. The word or phrase is considered to have a good category distinguishing ability, suitable for classification. According to the lexical weight from high to low, according to the number of keywords required or threshold to obtain the specified keywords, The TF-IDF lexical weight calculation formula is as follows: (3)

Among them, is the frequency of vocabulary , . is the frequency of inverted documents, , N is the total number of samples, and is the total number of articles containing .

All the short message data in the training set is processed according to the above three steps, and the keywords are used for characterization. Even if the vocabulary appears many times in the short text, the impact on the sentence is very small, so it is only retained once.

***3.2 Step 2***

Use the Word2Vec idea to convert the keywords extracted in step 1 into word vectors and construct a semantic space. At this time, if you directly mine association rules for the entire semantic space, not only the amount of data is large, but also all word vectors in the space do not have association relationships, easily causing the problems such as slow mining speed or memory overflow. Therefore, the idea of cutting the semantic space is adopted to "divide and conquer" big data. This paper uses a clustering algorithm to divide the semantic space, cluster semantically similar word vectors into a sub-semantic space, and reduce the similarity in the sub-semantic space to not less than the threshold θ ,( θ = 0.98) vocabulary generates "or" relationship (such as A1 | A2 | A3 ... | An), that is, a dictionary of synonyms is automatically generated. See Table 1 for some vocabulary of sub-semantic space.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  | | --- | --- | | Table 1 Sub-semantic space vocabulary | | | Category | Vocabulary | | Promotion | Good Gifts | Gifts, Big Broadcasting, Losing Money | Loss Losses | Bargains, Fiery | Hot, Joy, Losing Money | Sales| Loss, Losses | Losses, Lost | Missing, Communicating | Delivering | | Fraud | Rent | Landlord Rent | Room Fee, Card | Account , Lucky,  Selection | Appraisal, Issuing materials, obtaining certificates, engraving seals, invoices, points | | Illegal | Death, Lies, Opportunities, Hesitation, Tribulation | Disaster, Dharma, Sin | Crime, Suppression, Kung Fu, Blessing, Communist Party | Central Committee | | BlackMail | Forget about arranging, losing money, gambling | playing mahjong, anytime, anywhere | without leaving home | without going out, gambling | games, interruptions, ideas | luck to,  http|www.|.com|.c0m|.cn|.cc|.hk|.net|.apk|m.|.lt | | |  |  |  | | --- | --- | --- | | Table 2 Order of strategy words in sub-semantic space | | | | Strategy word | Feature ID | Support | | Betting | Playing Mahjong | T1 | 60 | | Gambling | Game | T2 | 60 | | http|www.|.com|.c0m|.cn|.cc|.hk|.net|.apk|m.|.lt | T3 | 60 | | Idea|Luck | T4 | 40 | | Parting | T5 | 20 | | Arrange | T6 | 20 | | Lose money | T7 | 20 | | Prophet | T8 | 20 | | Anytime, anywhere | without leaving home | without going out | T9 | 10 | |

***3.3 Step 3***

The FP-Growth association rule algorithm is used for each sub-semantic space to mine frequent itemset generation strategies in parallel.

*(1) Statistics first-level frequent itemsets*

Scan the illegal SMS training data, count the frequency of keywords in each sub-semantic space, and calculate the first-level frequent item set. One of the differences between illegal and normal text messages is that: within the specified time window, the number of repeated illegal text messages is at least a dozen times that of normal text messages, so set a small level of frequent itemsets with small support min\_support = 10, confidence = 80%. Reduce the probability of accidentally hitting the normal short message. Among them, the keywords that are larger than the small support degree are used as strategy words, and they are sorted according to the word frequency. The order is shown in Table 2.

*(2) Convert training message data to form a data record table.*

Each message is converted into a strategy word and its corresponding feature ID. The support of each record is the small value of the support degree of each strategy word in the message. See Table 3.

(4)

Table 3 Data record table

|  |  |  |  |
| --- | --- | --- | --- |
| SMS ID | Strategy word representation | Feature ID representation | Support |
| M1 | Gambling | Playing Mahjong, Gambling | Game, http | www. | .Com | .c0m | .cn | .cc | .hk | .net | .apk | m. | .Lt, Anytime, anywhere | without leaving home | without going out | T1,T2,T3,T4 | 40 |
| M2 | Gambling | Playing mahjong, points, ranking | T1,T5,T6 | 10 |
| M3 | Lottery | Betting | Mahjong, http | www. | .Com | .c0m | .cn | .cc | .hk | .net | .apk | m. | .Lt | T1,T3,T5,T6 | 10 |
| M4 | Gambling | Games, Losing Money, Prophets, Anytime, Anywhere | Without Home | Do Not Go Out | T2,T7,T8,T9 | 10 |
| M5 | Gambling | Game, http | www. | .Com | .c0m | .cn | .cc | .hk | .net | .apk | m. | .Lt, lose money, prophet | T2,T3,T7,T8 | 10 |

*(3) Use FP\_Growth to mine association rules*

Scan the data record table to generate the FP-tree multi-fork tree. The structure is shown in Figure 2. The FP-Growth algorithm is used to mine the frequent item set of the tree. The association rules are shown in Table 4.

Figure 2 FP-tree tree structure

Table 4 Association Rules

|  |  |  |  |
| --- | --- | --- | --- |
| ITEM ID | Model basis | Condition FP\_TREE | Association rules |
| T9 | {(T2 T7 T8:10)} | <T2:10 ,T7:10 ,T8:10> | T2T9:10,T7T9:10,T8T9:10,T2T7T9:10,T2T8T9:10,T2T7T8 T9:10 |
| T8 | { (T2 T7:10),(T2 T3 T7:10)} | <T2:10, T7:10>  <T2:10, T3:10, T7:10> | T2T8:20,T7T8:10,T2T7T8:20,T3T8:10,T2T3T7T8 |
| T7 | {(T2:20)} | < T2:20 > | T2T7:20 |
| T6 | {(T1 T2 T5:60) , (T1 T2 T3 T5:10)} | <T1:60, T2:60 ,T5:60>  <T1:10,T2:10,T3:10,  T5:10> | T1T6:70,T2T6:70,T5T6:70,T1T2T6:70,T1T5T6:70,T2T5T6:70,T1T2T5T6:70,  T1T2T3T6:10,T2T3T5:10,T1T3T5:10,T1T2T3T5T6:10 |
| T5 | {(T1 T2:10) , (T1 T2 T3:10)} | <T1:20,T2:20,T3:10> | T1T5:20,T2T5:20,T3T5:10,T1T2T5:20,T1T3T5:10,T2T3T5:10,T1T2T3T5:10 |
| T4 | {(T1 T2 T3:40)} | <T1:40,T2:40,T3:40> | T1T4:40,T2T4:40,T3T4:40,T1T2T4:40,T1T3T4:40,T2T3T4:40,T1T2T3T4:40 |
| T3 | {(T1 T2:40) , (T1 T2:10)(T2:20)} | <T1:50,T2:50><T2:20> | T1T3:50,T2T3:70,T1T2T3:50 |
| T2 | {(T1:60)} | <T1:60> | T2T1:60 |

***3.4 Step 4***

*(1) Assess association rules*

  The generated association rule contains 2 ~ N strategy words, as shown below:

"<lottery| Betting | Playing Mahjong, Arrangement>,

  <Lotery | Gambling | Play Mahjong, Gambling | Game, Arrangement>,… <Lottery | Gambling | Play Mahjong, Gambling | Game, http | www. | .Com | .c0m | .cn | .cc | .hk | .net | .apk | m. | .lt, idea| luck |>

The following issues should be noted during the evaluation of rules:

Too few policy words in the rules will cause many normal short messages to be missed

Excessive number of policy words in the rules will lead to violations of short messages. Although the hits are accurate but the hits are small, the effect of the rules is suppressed;

Some strategy phrases are neutral vocabulary, such as "<Prophet, anytime, anywhere | not leaving home | not going out>", which is easy to accidentally hit normal text messages and is not suitable as a strategy for filtering spam messages.

*(2) Conversion of association rules into strategies*

A strategy is composed of N strategy words and 3 kinds of associations {“AND (&)”, “or (|)”, “NO (!)”}, And its expression is <(A1 | A2) & (B1 | B2 | B3) &…! (F1 | F2 | F3)>, where the converted form is as follows: “<(gambling | gaming) & points>… <(Lottery | Betting | Mahjong) & (Gambling | game) & ranking> ”

***3.5 Step 5***

After evaluating the strategies, it is found that there are a large number of strategy inclusions, intersections, etc., such as: "0001 T1 & T6, 0002 T2 & T6, 0003 T1 & T2 & T6, 0004 T1 & T5 & T6, 0005 T1 & T2 & T5 & T6".

By constructing a strategy binary tree, strategies are effectively merged and compressed to form a streamlined strategy set. The process of constructing a binary tree is as follows.

*(1) Frequency of statistical elements*: First, the frequency of each element in each strategy is counted, sorted from high to low {T6: 5; T1: 4; T2: 3; T5: 2}.

*(2) Strategies are adjusted in accordance with the order of element frequency*. Each element is adjusted and compared in accordance with the order of strategy after frequency order, see Table 5.

Table 5 Strategy comparison results in order of frequency adjustment

|  |  |  |
| --- | --- | --- |
| Strategy ID | Before sorting | After sorting |
| 0001 | T1&T6 | T6&T1 |
| 0002 | T2&T6 | T6&T2 |
| 0003 | T1&T2&T6 | T6&T1&T2 |
| 0004 | T1&T5&T6 | T6&T1&T5 |
| 0005 | T1&T2&T5&T6 | T6&T1&T2&T5 |

*(3) Construct a binary tree*

Scan the strategy table according to the frequency adjustment order again to construct an ordered binary tree. The construction principle is that all strategies start from the root node, where the child node is on the left side of the tree, the sibling node is on the right side of the tree, the last element of each strategy is accompanied by the strategy ID, and each node is a triplet <strategy word, ID, state>, where the last node as a policy has an ID and a state. There are two states: {0: not extracted, 1: extracted}, the tuple that has been merged by the policy, the node status is marked as 1. In order to avoid the second generation strategy, the strategy binary tree structure is shown in Figure 3.

root

T6

T1|0001|0

T2|0003|0

T2|0002|0

T5|0004|0

T5|0005|0

Figure 3 Strategy binary tree structure

(4) *Traversing the binary tree merge strategy*,

the merge principle is:

  Recursive mid-order traversal of binary tree;

  After traversing and discovering that a node contains an ID,

Recursive mid-order traversal of binary tree;

After traversing, it is found that the next node contains an ID, and from the root node to all nodes on this path form a "and" relationship strategy;

Once it is found that the node has a right subtree, it always scans the right subtree. If it contains an ID, it becomes an "or" relationship and marks the node as 1.

The resulted merge scheme based on the binary tree achieves 1/3 ~ 1/2 compression, and obtains a good compression and merge effect. The effect is as follows:

0003 T6 & T1 & (T2 | T5)

0001 T6 & (T1 | T2)

**4. Experimental results and analysis**

**Reference**

YIH W, GOODMAN J, CARVALHO V R. Finding advertising keywords on Web pages[C]//Proceedings of the 15th International Conference on World Wide Web. New York: ACM Press, 2006: 213-222.

KUHN R, DE MORI R. A cache-based natural language model for speech recognition[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 1990, 12(6): 570-583.

MIHALCEA R, TARAU P. TextRank: bringing order into texts[Z]. 2004.

WANG Q, CHEN Z Y, GUO J, et al. Project keyword lexicon and keyword semantic network based on word co-occurrence matrix[J]. Journal of Computer Applications, 2015, 35(6): 1649-1653.

DONG Z D, DONG Q, HAO C L. Theoretical findings of HowNet[J]. Journal of Chinese Information Processing, 2007, 21(4): 3-9.

BENGIO Y, DUCHARME R, VINCENT P, et al. A neural probabilistic language model[J]. Journal of Machine Learning Research, 2003(3): 1137-1155．

MIKOLOV T, CHEN K, CORRADO G, et al. Efficient estimation of word representations in vector space[J]. Computer Science, 2013.

BENGIO Y, DUCHARME R, WINCENT P. Neural probabilistic language model neural probabilistic language model[Z]. 2003.

HANM J W, KAMBER M, PEI J. Data mining concepts and techniques[M]. Translated by FAN M, MENG X F. Beijing: Machinery Industry Press, 2012: 157-179.