International Journal of Machine Learning and Cybernetics (2021) 12:2517–2528
https://doi.org/10.1007/s13042-021-01335-3

ORIGINAL ARTICLE

Case2vec: joint variational autoencoder for case text embedding representation

Ran Song[1,2] · Shengxiang Gao[1,2] · Zhengtao Yu[1,2] · Yafei Zhang[1,2] · Gaofeng Zhou[1,2]

Received: 17 July 2020 / Accepted: 14 April 2021 / Published online: 7 July 2021

Abstract
The embedding representation of the case text represent text as vector which consist information of original texts abundantly. Text embedding representation usually uses text statistical features or content features alone. However, case texts have characteristics that include similar structure, repeated words, and different text lengths. And the statistical feature or content feature cannot represent case text efficiently. In this paper, we propose a joint variational autoencoder (VAE) to represent case text embedding representation. We consider the statistical features and content features of case texts together, and use VAE to align the two features into the same space. We compare our representations with existing methods in terms of quality, relationship, and efficiency. The experiment results show that our method has achieved good results, which have higher performance than the model using single feature.

Keywords Embedding representation · Statistical feature · Content feature · Variational autoencoder

1 Introduction

The case text is important information in the field of justice, and play an important role in multiple sections of judicial work. Case text embedding representation, which easy to compute and contain the relationship with each other, is significant for judicial work. This work has great help to judicial work such as case retrieval, case comparison and case correlation analysis. The key to embedding case text is how to use multiple text information more effectively.

The case text is *verdict form* that record the entire process of a case from occurrence to judgment. The case text usually consists of two parts: findings of fact and conclusions of crime. Findings of fact introduce process of crime event and people involved in whole case. Conclusions of crime is including crime, laws and judgment result. The main difference between the findings of fact of different cases lies in events and people. The difference in the conclusions of crime part mainly determine by the difference in sentencing and judgment.

For case text, it usually has following characteristics, mainly including fixed structure, fixed terminology, and large differences in text length. See Table 1 for example, the bold formatting part is findings of fact and other part is conclusions of crime. For case text: First, the conclusions of crime in case text must follow a fixed format; Second, the narrative requirements in the verdict document are detailed, and the facts are generally described in the order of time. Third, because the narrative process and the level of evidence of each case are different, the length of the text of each case is also different, as shown in Fig. 1. The above factors made case texts similar with each other in conclusions of crime and difficult understanding in findings of fact.

✉ Shengxiang Gao
gaoshengxiang.yn@foxmail.com

Ran Song
song_ransr@163.com

Zhengtao Yu
ztyu@hotmail.com

Yafei Zhang
zyfeimail@163.com

Gaofeng Zhou
zgf@stu.kust.edu.cn

[1] Faculty of Information Engineering and Automation, Kunming University of Science and Technology, Kunming 650500, Yunnan, China

[2] Yunnan Key Laboratory of Artificial Intelligence, Kunming University of Science and Technology, Kunming 650500, Yunnan, China

Table 1 Case text example

Case text example
People's procuratorates of Pingquan city charged: **The defendant Liu and the village resident Li were in the fields of Sanjia Village, a spat occurred because Liu's work with a hook machine damaged Li's field, Then Liu and Li tore, Liu rubbed Li's head and face with stones and escaped from the scene. At about 21 o'clock that day, Li was found unconscious by relatives in a ditch near the scene, and then sent to hospital for treatment. After appraisal, Li's injury was classified as minor injury. The prosecution consider, The defendant Liu illegally deprived others of his life.** His behavior violated the provisions of the ***Criminal Law of the People's Republic of China***. The criminal facts are clear and the evidence is true and sufficient. He should be held criminally responsible for the crime. The defendant Liu had already committed a crime, but he did not succeed for reasons other than his will. It was an attempted crime, which could be lighter or a lesser punishment. In the incidental civil lawsuit, the plaintiff requested the defendant to compensate medical expenses of 24,164.48 Yuan, lost work costs of 18,000.00 Yuan, nursing costs of 2200 yuan, food allowances of 2200.00 Yuan, appraisal costs of 800.00 Yuan, nutritional costs of 1100 yuan, transportation costs of 2000.00 yuan, and mental damage relief of 100,000.00 Yuan, the above sum totals RMB 150,464.48. The defendant Liu believed that it constituted crime.
平泉县人民检察院指控: 被告人刘某某与本村居民李某甲在三家村田地内, 因刘某某带钩机干活轧了李某甲家田地而发生口角, 后刘某某与李某甲发生撕打, 刘某某用石块将李某甲头面部打伤后逃离现场。当日21时许, 李某甲被亲属发现昏迷于现场附近一河沟内, 随后被送医院治疗。经鉴定, 李某甲伤情为轻伤一级。公诉机关认为, 被告人刘某某非法剥夺他人生命, 其行为触犯了《中华人民共和国刑法》之规定, 犯罪事实清楚, 证据确实、充分, 应当以罪追究其刑事责任。被告人刘某某已经着手实施犯罪, 由于意志以外的原因而未得逞, 是犯罪未遂, 可以比照既遂犯从轻或者减轻处罚。附带民事诉讼原告人要求被告人赔偿医疗费24164.48元, 误工费18000.00元, 护理费2200元, 伙食补助费2200.00元, 鉴定费800.00元, 营养费1100元, 交通费2000.00元, 精神损害抚慰金100000.00元, 上述款项合计人民币150464.48元。被告人刘某某认为其构成*****罪。

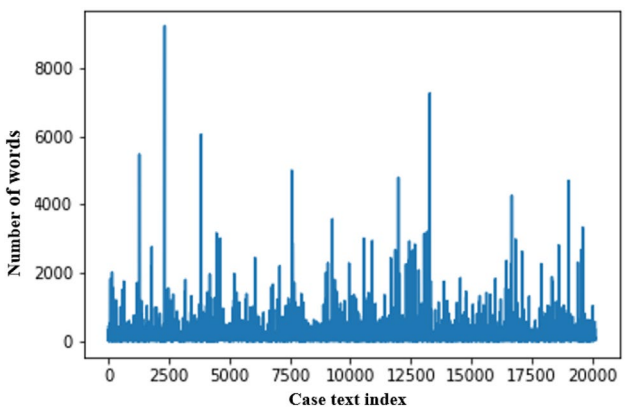

Fig. 1 Case text words count figure

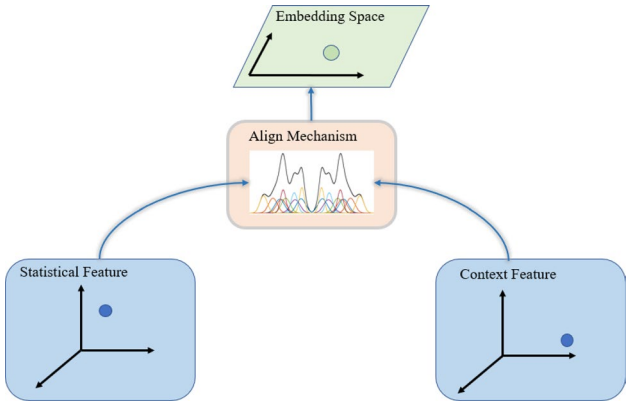

Fig. 2 The process figure

For text representation, traditional methods often use statistical features to represent text well, but it cannot completely represent case text that contain fixed and terminology. The reason is that these characteristics cause the statistical feature vectors to be very similar. Recently, neural network model has led to superior results in many NLP tasks and can obtain content features well. But for long case text, it is difficult that the neural network to cover long-distance text contexts. Therefore, when representing the case texts, different information should be considered, and statistical features and content features complement each other. However, statistical features and content features are not in a same space. How to effectively fuse the two features into the same space is the key to the case text representation. The process of fusing statistical features and content features into the same space can be regarded as the process of embedding representation of case text.

In this paper, we propose a case text representation model based on joint VAE. We propose a case text representation model based on joint VAE. The process is shown in Fig. 2, according to similar structure and repeated words, current deep learning model can learn the content features well, so we add self-attention mechanism to get content feature better. According to different text lengths, we use word frequency statistical features to make up for the fact that deep learning cannot deal with the large length span. However, the statistical features describe the word frequency of entire case text and content features is to represent context of case text. And tow features are extracted from different way, thus they are not in the same space and not the same distribution. Aiming at the problem, this paper regards both statistical features and content features as combination of multiple Gaussian distributions, and use VAE to align two features into the same space. The KL divergence loss in VAE can align the variational vector to certain distribution and the reconstruction loss allows model to understand the case text.

Finally, the variational vector that combines two features and aligns to an existing distribution is regarded as the representation of the case text.

This paper uses the above model to represent the case text and obtain a representation vector. This vector should contain the original information of the case text as much as possible. Therefore, in the experiment of this paper, the distance between the representation vectors is calculated to verify the quality of the representation. The closer vectors mean the more similar the cases that represent by our method. If there are more vectors around a representation vector and its kind, it means that the quality of the representation vector is higher.

In addition, this paper considers not only the quality of the representation vector, but also the utilization efficiency of the representation vector. This will improve the efficiency of the model when using vectors for downstream tasks. The code is released on github.[1]

2 Related work

2.1 Statistical feature extraction

For the statistical features of text, the main method is to encode the text by extracting the key statistic index information of the text through statistical model. The bag-of-words model represents text as a vector base on the word frequency information. Gururangan et al. [1] trained VAE using words frequency information to extract variational vectors for text, and proved that using variational vectors can improve low-resource classification tasks. Zhao et al. [2] propose a new document representation method named fuzzy Bag-of-Words, and address the aforementioned issues. Ma et al. [3] propose an approach that uses both the sentences and the bag-of-words as targets in the training stage, in order to encourage the model to generate the potentially correct sentences that are not appeared in the training set, and improve translation task. The TF-IDF algorithm considers both the word frequency and the inverse document frequency of the text, and adds the relationship between the texts. Trstenjak et al. [4] uses TF-IDF to represent documents, and uses KNN to achieve documents classification. Zhu et al. [5] propose a refined term frequency inversed document frequency to find hot terms, based on time distribution information and user attention. Blei et al. [6] proposed the LDA topic model to extract the topic representation of the article. The topic representation can also be regarded as the text representation. As an external knowledge, statistical features are often used to enrich the representation of text, Johnson et al. [7] used CNN to separately encode sequence information and word frequency statistics information, which improved the classification performance of the model. Naz et al. [8] proposed a robust feature extraction method. This method extracts statistical features based on a sliding window from right to left, and combines RNN to improve the accuracy of text classification.

2.2 Content feature extraction

For text content features, it is common to use deep learning models to learn the context information of the text, and encode the context information into a vector. Kim [9] used CNN networks to encode text into vectors to achieve text classification. Yang et al. [10] proposed a hierarchical attention mechanism network for document classification, which can encode long texts. Gupta et al. [11] claim that training word embeddings along with higher n-gram embeddings helps in the removal of the contextual information from the unigrams, resulting in better stand-alone word embeddings. Yang et al. [12] proposed three strategies to stabilize the dynamic routing process, and used capsule networks to encode text, which improved the performance of text classification. The language model has a good performance on the extraction of content features. Mikolov et al. [13] used the CBoW model and Skip-gram model to represent words as embedding vectors in the same space, and these vectors represented the words well. LSTM [14] is a neural network structure that can effectively time series information, and has a good performance in language models. Peters et al. [15] proposed the ELMo pre-trained language model, which is a bidirectional language model based on LSTM. It uses the task of forward prediction and backward prediction of the language model, so that it can fully find the context features of the text. Recently, Vaswani et al. [16] proposed the Transformer structure, which is a neural network structure that can capture longer distances and faster training time than LSTM. It consists of an encoder and a decoder, where the encoder can obtain text information in both directions. Transformer has achieved quite good results on the SEQ2SEQ task, and also proves that Transformer is a network structure with good ability to obtain context features. Radford [17] and others used the Transformer decoder to train GPT for generating tasks. Devlin et al. [18] the BERT pre-trained language model and used transformer encoder to train a pre-trained language model using a large amount of data and self-supervised methods, which performed well on many natural language understanding tasks. Bert proved to be effective in obtaining contextual information, bert-technology has become one of the hottest research directions in natural language processing.

[1] https://github.com/Maxpa1n/case2vec.

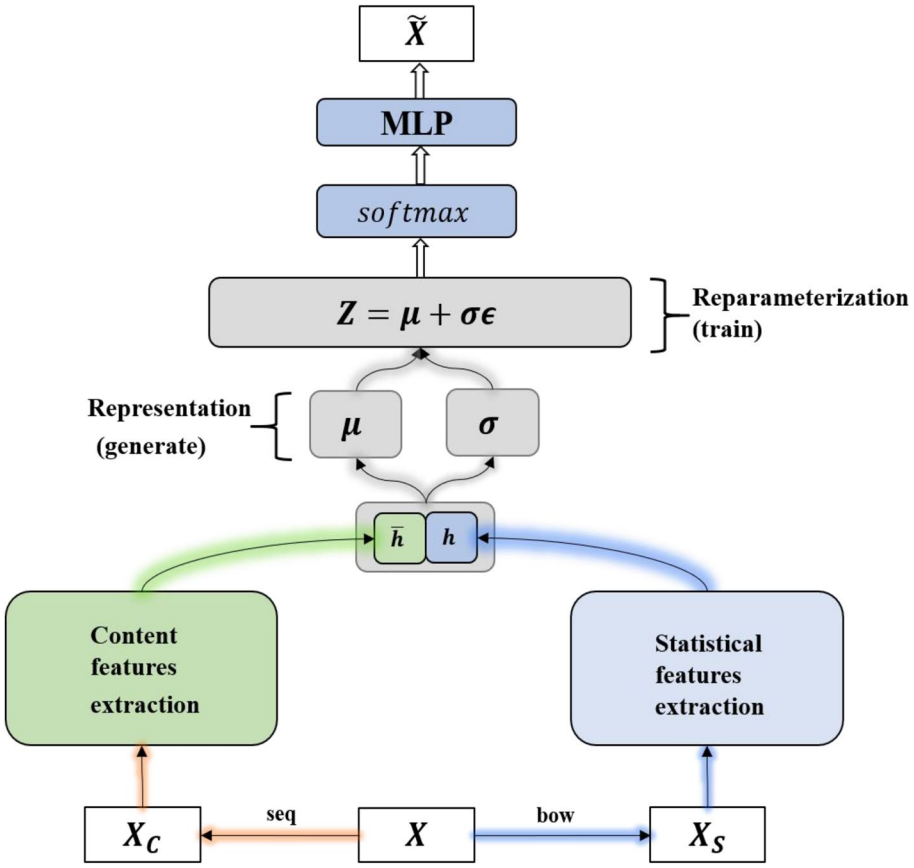

Fig. 3 Joint variational autoencoder model structure

2.3 Variational autoencoder

Recently, VAE has achieved good results in feature representation. The main idea of VAE is to treat real samples as Gaussian mixture models, normal sample from these Gaussian distributions, and then use normal samples to restore real samples. It can effectively mine the hidden features of text in a self-supervised channel, and has advantages in feature mining. Kingma and Welling [19] first proposed VAE and used it in the field of image generation. Bowman et al. [20] first applied VAE to natural language processing. It uses the bag of words vector to train the VAE model to obtain the latent representation of the document. Using this representation effectively improves the question answering system. Yishu et al. [21] used the bag-of-words model representation to represent documents as vectors. Using the bag-of-words vector to train the VAE to obtain the latent representation of the document, the use of this latent representation effectively improves the correct rate of correct answer matching in the question answering system. Yang et al. [22] used dilated convolutions as VAE decoder. It effectively reduces the impact of content features on language models, so that language models are more dependent on latent representations. And prove that using latent representations performs well on text classification tasks. Hoyle et al. [23] introduce a generative model of sentiment lexica to combine disparate scales into a common latent representation and realize this model with a novel multi-view VAE, improved sentiment text classification task. Zhao et al. [24] introduced the dialogue behavior as a condition on the traditional conditional VAE. A seq-to-seq conditional VAE network is constructed. The latent representation of dialog behavior improves the diversity of dialog tasks. Kusner et al. [25] represent context-free grammar as a syntactic parse tree, and propose a variant automatic encoder. It can directly encode and decode between these parse trees to ensure that the generated output is always valid. Li et al. [26] introduces a new VAE model, the latent tree VAE (VAE), is separate clustering are enforced on different subsets of the latent features.

3 Methods

This part includes the training process of the model and the process of encoding the case text. The model structure is shown in Fig. 3. For one case text $D = \{X, \mathrm{Y}\}$, where X is text and Y is the category of the case text. The case category Y is only used to test the result of the case text representation. The entire model training process and encoding representation process are unsupervised. X_s is pressed the word

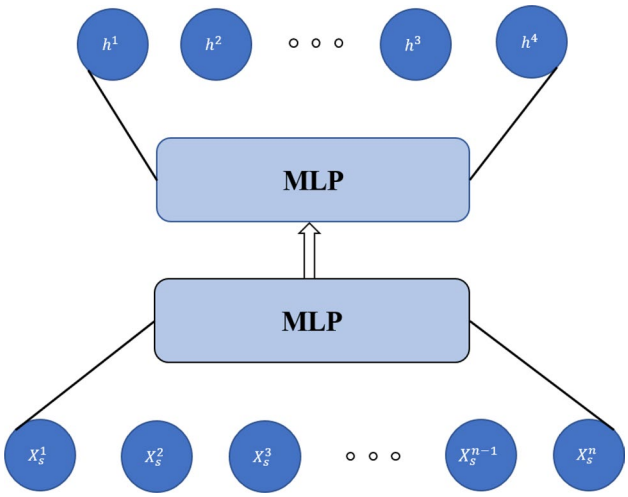

Fig. 4 Statistical feature extraction model

frequency of the case text, X_c is expressed as the sequence of case text. Both representations are based on the case text, but in two different ways.

3.1 Statistical feature for case text

This paper uses the bag of word model to obtain the word frequency statistical information of case text. Word frequency statistics is fixed length vector that can ignore the length of the case text. Since the word frequency statistical vector is sparse vector. And use two MLP to further extract the statistical features of the text, as shown in Fig. 4. The process can be described as:

$$X_S = \mathrm{BOW(X)} \tag{1}$$

$$\mathrm{c} = MLP\left(X_S\right) \tag{2}$$

$$\mathrm{h} = MLP(c) \tag{3}$$

where h is vector containing statistical features, and X represent the case text, c is hidden variable.

3.2 Content feature for case text

This paper uses the self-attention mechanism [10] to encode the text and obtain the content features from both directions. When input the case text, the Bert-like input method is used to ensure that one token represent the information of the sentence:

$$[CLS]case\ text[EOT] \tag{4}$$

$[CLS]$ means the beginning of the text, $[EOT]$ means the end of the text.

Because the self-attention mechanism can simultaneously obtain text information from both directions, the text sequence is out of order for the model. To enable the model to obtain sequence information, it is necessary to add position embedding information to each word in the sequence, as shown in Fig. 5. The process can be described as:

$$\mathrm{X_c} = token(\mathrm{X}) \tag{5}$$

$$E_w = \mathrm{X_c} W_{em} \tag{6}$$

$$E_p = \mathrm{X_c} W_{pos} \tag{7}$$

$$E = E_w + E_p \tag{8}$$

where W_{em} is the word embedding matrix and W_{pos} is the position embedding matrix. The *token* is tokenizer base on law vocabulary.

The model need to pay more attention to the important information of the text, self-attention is used to distinguish the importance of the words in the sequence, and use the multi-head self-attention mechanism to make the model learn information in different representation subspaces, as shown in Fig. 6.

$$\mathrm{Q} = MLP_q(E) \tag{9}$$

$$\mathrm{K} = MLP_k(E) \tag{10}$$

$$\mathrm{V} = MLP_v(E) \tag{11}$$

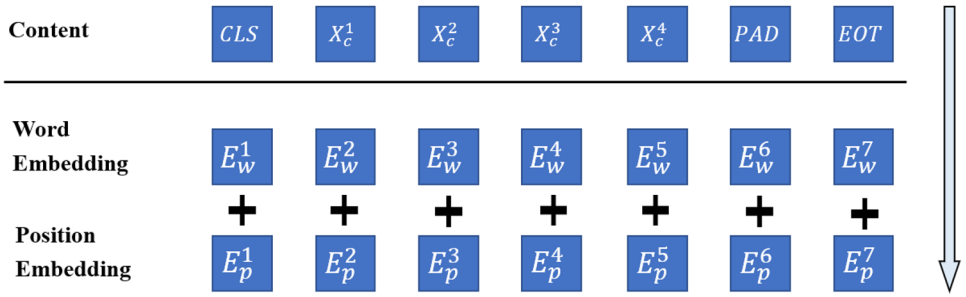

Fig. 5 Content feature embedding

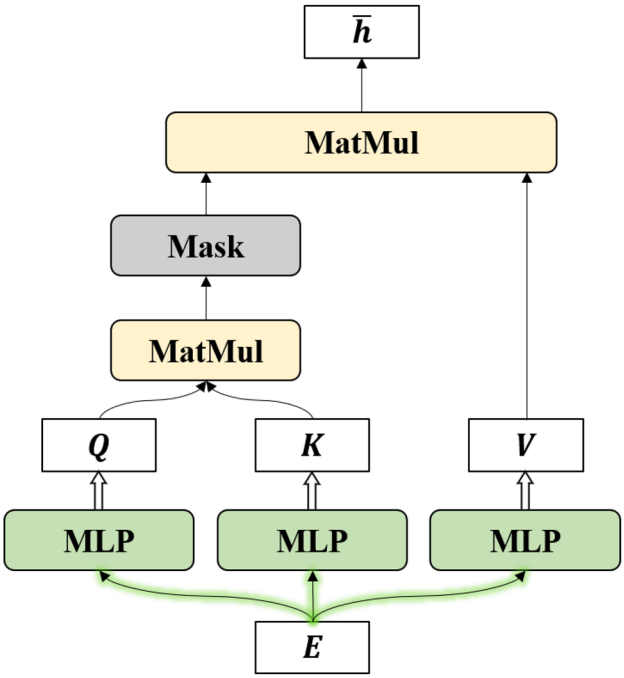

Fig. 6 self-attention mechanism

$$MultiHead(Q,K,V) = Concat\left(head_1,\ldots,head_h\right)W^0 \quad (12)$$

$$\overline{h}_i = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}} * MASK\right)\text{V} \quad (13)$$

where $\sqrt{d_k}$ is the dimension of K, $\overline{h}_i$ is the vector containing the content features, Q, K, V are all nonlinear transformations of E_i, W^0 is combination matrix, and MASK is the matrix described below.

For the data that does not meet the batch length, if it is simply replaced with 0, it will cause the position information to be retained. So uses the attention MASK approach, so that the word information and position information can be removed at the same time. There are two forms of MASK information for each word:

$$\text{MASK} = \begin{cases} 1 & nomask \\ -\infty & mask \end{cases} \quad (14)$$

After the softmax function, the attention weight of the MASK part is 0, and the information at that position will be completely hidden.

3.3 Variational autoencoder

VAE structure is used to align different distributions of information into the same space. First, assume that all case texts $X = \{x_i\}$ are independent and identically distributed variables.

VAE is autoencoder structure consisting of an encoder and a decoder, where $q_\phi(\cdot)$ is the encoder and ϕ is the parameter of the encoder, and the model encoding process can be described as:

$$Encoder \sim q_\phi(z|x) \quad (15)$$

where z is the latent vector.

And $p_\theta(\cdot)$ is the decoder, where θ is the parameter of the decoder, and its decoding process can be described as:

$$Dncoder \sim p_\theta(\tilde{x}|z) \quad (16)$$

where $\tilde{x}$ is reconstruction vector by decoder.

- For autoencoder structure, the encoder transform x into latent vector z and allows the decoder to restore $\tilde{x}$ from z. The goal of model training is to make x and $\tilde{x}$ as similar as possible. The loss of the process is:

$$\mathcal{L}_{re} = \min\{D(x,\tilde{x})\} \quad (17)$$

$$\mathcal{L}_{re} = E_{q_\phi(z|x_i)}\left[logp\left(x_i|z\right)\right] \quad (18)$$

where $D(\cdot)$ is a function that calculates the distance between x and $\tilde{x}$.

After building the autoencoder, a new training goal is to align the encoder to a known distribution. The purpose of this step is to take the input x as Gaussian mixture model and fit its distribution, which can be described as:

$$\mathcal{L}_{kl} = -\text{KL}(q_\phi(\text{z}|x_i)||p_\theta(\text{z})) \quad (19)$$

The loss function $\mathcal{L}_{kl}$ is the KL divergence of the distribution $q_\phi(\text{z}|x_i)$ and the distribution $p_\theta(z)$. The goal is to make the distribution $q_\phi(\text{z}|x_i)$ as close as possible to the distribution $p_\theta(z)$. In order to ensure that $p_\theta(z)$ follows the standard normal distribution, the reparameterization trick of Kingm [19] is used:

3.4 Joint VAE for feature alignment

According to the previous steps, the bag of word model and the self-attention mechanism are used to obtain the word frequency statistical information h_i and content information $\overline{h}_i$. We use the MLP network as encoder, and concat the statistical information and content information. The steps can be described as:

$$H_i = [h_i : \overline{h}_i] \quad (22)$$

$$\mu_i = MLP_\mu(H_i) \quad (23)$$

Table 2 The number of cases table

Case type	Number of cases
Production and Sales of Fake Medicines (生产、销售假药)	1940
Falsely Making out Specialized VAT Invoices (虚开增值税专用发票、用于骗取出口退税、抵扣税款发票)	1983
Rape (强奸)	2067
Voluntary Manslaughter (故意杀人)	2100
Embezzlement (贪污)	2116
Illegal [manufacture, sale, transport, post][firearms, ammunition, explosive] 非法[制造、买卖、运输、邮寄][储存枪支、弹药、爆炸物]	1932
Illegal Possessing or Concealing firearms or Ammunition) 非法[持有、私藏][枪支、弹药]	2094
Illegal Occupies Farmland (非法占用农用地)	1964
Illegal Funds-raising (非法吸收公众存款)	2010
Illegal Business Operations (非法经营)	1920

$$\sigma_i = MLP_\sigma(H_i) \tag{24}$$

$$z_i = \mu_i + \sigma_i \epsilon_i \tag{25}$$

where network parameters $\{\mu, \sigma\}_i \in \phi$, MLP_μ is the mean extraction layer, MLP_σ is the variance extraction layer, μ_i is the mean vector of H_i, σ_i is the variance vector of H_i, z_i is the hidden vector, ϵ_i is a vector sampled from the normal distribution.

For the decoder:

$$\beta_i = softmax(z_i) \tag{26}$$

$$\tilde{x}_i = MLP_d(\beta_i) \tag{27}$$

where the network parameter $\{d\}_i \in \theta$, MLP_d is the decoder layer, $\tilde{x}_i$ is the reconstruction vector, and the distribution of β_i variational vector.

The above is the construction process of the joint VAE. According to the previous definition of VAE, the loss function is:

$$\mathcal{L}_{re} = -(x \cdot \log\_softmax(\tilde{x})) \tag{28}$$

$$\mathcal{L}_{kl} = -\frac{1}{2}\left(1 + \mu - \sigma^2 - e^{\mu}\right) \tag{29}$$

$$\mathcal{L} = \mathcal{L}_{re} + \mathcal{L}_{kl} \tag{30}$$

where $\mathcal{L}_{re}$ is the reconstruction loss of VAE, $\mathcal{L}_{kl}$ is the KL divergence loss. For the reconstruction loss, because x is sparse vector, the *softmax* will cause the loss to be too small, it would be cause the gradient disappear easily. Therefore, we use log_*softmax* to amplify the loss to increase gradient. Since the length of case text is very different, the longer text may cause gradient explosion, so the gradient truncation technique is used to ensure the stability of the training process.

After the model training, to stabilize the output representation vector, no reparameterization trick when generating the representation vector. Because $p(z|x) = \mathcal{N}(\mu, \sigma^2)$, the encoder has the ability to parse out the potential feature distribution of the case text, so use μ to represent the characterization vector of the text:

$$\mu_i = MLP_\mu(H_i) \tag{31}$$

The above is the training and generation steps of the model in this paper.

4 Experiment

To experiment, this paper constructed 20,126 real case, the data is judgment documents sourced from *China Judgement Online*. The *verdict form* records the trial process and results of the court. All the texts of the cases come from real cases. Because our experiment should judge the quality of representation by case category, we remove some strong information for case classification. Like" His behavior violated the provisions of Article 234 of the Criminal Law of the People's Republic of China and constituted the crime of intentional injury." processed as "His behavior violated the provisions of Article of the Criminal Law of the People's Republic of China and constituted the crime."

We collect case texts of 10 classifications, each case text corresponds to one crime entry, and the number of cases of each classification is shown in Table 2. Some case texts correspond to multiple crime entries, and one of them is selected as the label of the case text to ensure that there is only one label for a case text.

4.1 Hyperparameters

In this paper, the vocabulary size is a 35,281, that constructed by selecting words with frequency greater than 3 and removing stop words. Implementing the experiments in Pytorch [27], Using the optimizer Adam [28] with a learning rate of 3e-3, $\beta_1$is0.9andβ_2 is0.999. The statistical feature encoder uses MLP, the hidden layer dimension is 512, and the activation function uses ReLu. The content feature encoder uses transformer encoder, the hidden layer dimension is 256, and the number of attention heads is 8, and the sequence length is 100. For variation autoencoder, MLP_{μ} size is 20 and MLP_{σ} size is 20, the activation function uses softmax. The sampling vector in reparameterization obeys normal distribution $\mathcal{N}(0,1)$. So that the variation vector z dimension is 20. Batch size is set to 1000 during training.

For the problem of gradient explosion, we have two options. First, using the gradient cropping technique, the threshold for gradient cropping is 20. Second, the model does not shuffle the input during training, because this paper uses the big batchsize training strategy, random shuffle will cause multiple noise data to appear at the same time and cause gradient explosion. After experiments, the two methods can be used separately or together to avoid gradient explosion and have little effect on the results.

4.2 Evaluation

This paper proposes a method to evaluate the quality of case text representation. Intuitively, for case texts, if cases texts are more similar, the distance of the representation vectors of case texts is also closer. Therefore, this paper selects the case closest to it for a target case text and calculates the number of the same classification as the target case.

For each case text of classα, it is encoded as a vector z, and α case has a total of N case texts. Calculate the cosine similarity between each case text, and select K cases that are most similar to the target case. M represents the number of K cases in the same classification as the target case.

$$S_{one\ case} = \frac{M\ same\ classification}{K\ closest\ vectors} \tag{32}$$

$$S_i^{\alpha} = \frac{M}{K} \tag{33}$$

$$S^{\alpha} = \frac{1}{N}\sum_{1}^{N} S_i \tag{34}$$

S^{α} is the accuracy in class α. K takes different values to detect different interval accuracy.

4.3 Case representation classification experiment

In order to verify the representation quality of the case text representation vector in the embedded space, this paper did the accuracy experiment of case representation, Table 3 uses a series of models to encode case text as embedding vectors. Then use the evaluation method proposed above to obtain the accuracy of each type of case, K takes the total number of types of cases, guarantee to cover all cases. BOW is to use the bag of words model to represent the case text, TF-IDF is to encode case text using term frequency-inverse document frequency, CBOW uses case text data for training and uses the mean of word vectors to represent case text, BERT (DIRECTLY) is to directly use BERT to encode the case text, BERT + MLM is to use BERT to fine-tune the mask language model task on the case text before encoding the case text. Both ROBERTA [29] and ALBERT [30] are high-performance pre-trained language models, which obtain state-of-the-art result on some NLP tasks. The VAE + BOW only considers the statistical features of word frequency.

Analysis Table 3 shows, it can be seen that neither statistical information nor content information can completely represent the case text. The accuracy of the vectors represented by BOW and TF-IDF is lower than the method proposed in this paper, and statistical information has a better effect on some cases with outstanding characteristics, Illegal Occupies Farmland and Falsely Making out Specialized VAT Invoices et al. Because such cases have strong characteristics and domain, often use some special vocabulary, and that vocabulary basically will not be described in other classification. Therefore, such cases can be well distinguished from other cases. Moreover, most of these cases belong to the financial case, which has strong domain and professionalism.

For content information, BERT-like language model can capture sequence information well and has achieved good results in many natural language processing tasks. However, from the experimental results, the original BERT does not have a good understanding of the case text in the judicial field, and BERT still has some deficiencies in the field content. And using the case text to perform the mask language mode task on the native BERT and fine-tune it. It can be seen that a great improvement has been obtained, but it is difficult for BERT to process text with a large sequence length difference at the same time. BERT-like language models have limit for text length that sustain 512 tokens in one sentence. Regardless of content or length, BERT is not applicable to this task.

The method proposed in this paper combine both the statistical and content features, and get good results in most case. However, the traditional method is less effective for some cases that has similar findings and conclusions. Cases such as Illegal Manufacture Firearms and Possessing or

Table 3 Case representation accuracy

	BOW	TF-IDF	CBOW	BERT (DIRECTLY)	BERT + MLM	ROB-ERTA + MLM	ALBERT + MLM + SOP	VAE + BOW	VAE + BOW + ATT
	Statistical feature			Content feature			Ours		
Sales fake medicines	0.471	0.511	0.352	0.319	0.635	0.685	0.831	0.855	**0.874**
Making out VAT invoices	0.567	0.570	0.633	0.460	0.602	0.668	0.821	0.894	**0.947**
Rape	0.334	0.210	0.240	0.262	0.345	0.412	0.671	0.637	**0.768**
Voluntary man-slaughter	0.346	0.219	0.206	0.372	0.511	0.564	0.735	0.691	**0.772**
Embezzle-ment	0.307	0.292	0.192	0.261	0.503	0.511	0.752	0.675	**0.817**
Manufacture firearms	0.342	0.303	0.229	0.226	0.357	0.351	0.412	0.404	**0.430**
Possessing firearms	0.338	0.291	0.252	0.204	0.379	0.409	0.491	0.466	**0.502**
Occupies farmland	0.608	0.534	0.332	0.308	0.605	0.691	0.845	0.929	**0.940**
Funds-raising	0.405	0.422	0.287	0.311	0.652	0.704	0.712	0.701	**0.810**
Business opera-tions	0.345	0.260	0.177	0.160	0.347	0.349	0.353	0.350	**0.356**
Average	0.406	0.361	0.290	0.288	0.493	0.534	0.662	0.660	**0.721**

Concealing Firearms are often accompanied, so the model performs poorly.

4.4 Case text representation visualization experiment

In order to observe the positional relationship between the representation of case in classifications space, this paper visualizes the case text representation. We use the TSNE algorithm to reduce the text representation to two-dimensional. Then, according to different case classifications, using different colored points, all 20,126 cases are drawn in the coordinate system. On the right is the color bar to the case classification.

As shown in Fig. 7a, the BOW model that have no normalization for data is very sensitive to noisy data. Some representations of long text that carry information more than short text is too big in value. It can be seen that some point away from the cluster center. As shown in Fig. 7b, the vector represented by the TF-IDF model has a certain anti-interference ability, but the distribution of TF-IDF's case representation is very scattered. Repeated wording and similar structure lead to very similar statistical features of different cases.

We use BERT to perform mask language model task on the case text and represent the case text. As can be seen from Fig. 7c, the sequence information of the model can well distinguish the differences between cases. But some case texts, the case representation in the upper left corner of Fig. 7c, have poor performance. Due to the length limitation of the neural network model, the sequence feature representation is incomplete, resulting in a single region in space. As shown in Fig. 7d, The VAE + BOW model has good effects on noise and boundaries, but lacks the relationship between cases.

Figure 8 shows the method proposed in this paper to get the case text representation in the same embedded space, the method has good results for noise and boundaries. In this representation space also shows the relationship between different classification cases. As shown by the black line in Fig. 8, the cases to the left of the black line are Rape, Voluntary Manslaughter, Illegal Possessing Firearms and Illegal Manufacture Firearms, which all involve violent crimes, and

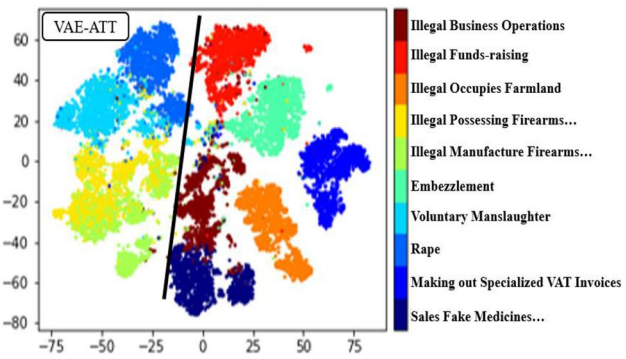

Fig. 7 VAE + ATT visualization

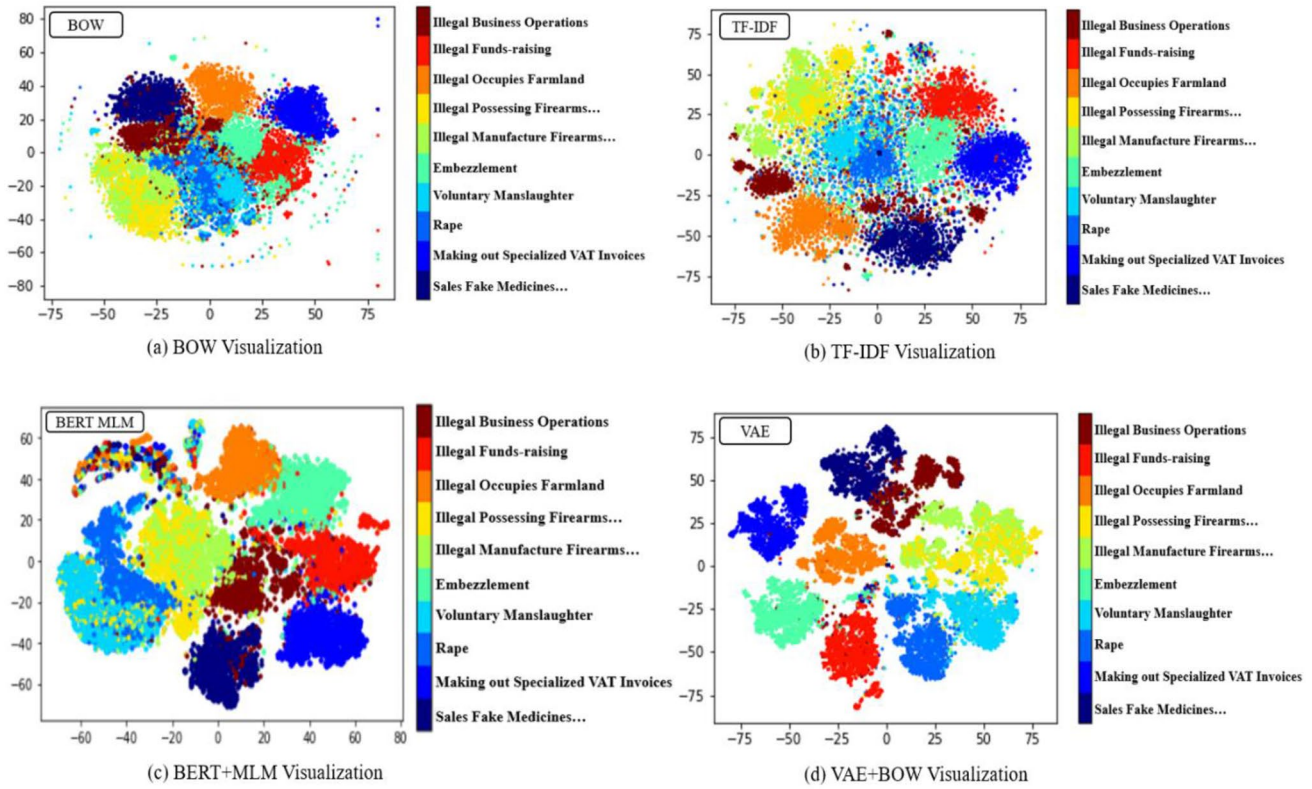

Fig. 8 Other model visualization

have similar the findings. The Illegal Possessing Firearms and Illegal Manufacture Firearms have high coincidence. The reason is that these two cases often occur together. The cases on the right of the black line are all cases of economic crimes. The representations from our method are in line with the actual case. In this embedded space can explain the relationship between cases, it shows that the method can well represent the case text in the same space.

4.5 Case text representation efficiency experiment

For the case representation vector, not only to consider the representation quality of the vector, but also the dimension of the representation vector of the case, lower dimensions can be easier when used in downstream tasks. So lower dimensions and better representation should be considered at the same time. In this experiment, we use the representation vector to perform the TSNE dimensionality reduction algorithm and record the time.

As shown in Table 4, for the statistical features of BOW and TF-IDF, the dimension of the case representation vector is the size of the vocabulary. The 35,281-dimensional vector is too long for most downstream tasks to difficult use. For coding such as BERT pre-training language models, recent research shows, the larger the representation dimension used by the pre-training model, the better the representation effect, 768-dimensional vector is acceptable to downstream tasks. The case representation vector proposed by the method presented in this paper is only 20-dimensional, which is far lower than other models.

4.6 Case correlation experiment

Because the similarity between cases is difficult to distinguish, so the text manually judges whether the two cases are similar according to the key plot of the case. Give a target case text and use the method of this paper to encode, select the 10 cases closest to the target case representation vector, determine whether there is a key plot. The results are shown in Table 3:

Table 4 Representation efficiency experiment table

	Dimension	Cost time (s)
TFIDF/BOW	35,281	3.81E+04
BERT/ALBERT/ROBERTA	768	680
VAE	20	90
VAE + ATT	20	90

Table 5 Correlation experiment table

Whether to include the plot	yes	none
Selling mark six to others	7	3
Attempted rape	8	2
Voluntary manslaughter caused by emotional dispute	8	2
Selling imported drugs illegally	7	3

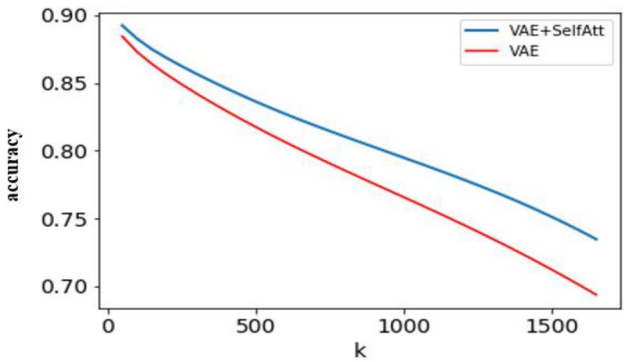

Fig. 9 Comparison of robustness

It can be seen from Table 5, the model can mine the latent features of the case text, especially for cases with special criminal circumstances. Based on the above results, case texts with the same characteristics will have a shorter distance.

4.7 Robustness experiments

To study the robustness and generalization ability of the model, according to the evaluate method of 3.2, the accuracy rate depends on how many of the closest K cases and the target case are similar cases. As the value of K increases, the higher the accuracy, the stronger the robustness of the model.

As shown in Fig. 9, as the value of K increases, the average accuracy of VAE decreases faster than VAE + ATT. So VAE + ATT is more robust than VAE. It shows that the generalization ability of this method is stronger.

5 Conclusion and future work

This paper proposes case text embedding representation method based on joint VAE. Aiming at the similarity of structure and the repeated words, we use self-attention mechanism to get content feature. For different lengths of case texts, we use bag of words get statistical features of case text. For the problem that the content features and statistical features are not in the same space, we use VAE to align the two feature pairs into the same embedding space, and representing the case text as a computable vector embedded in the space. This paper designs experiments to explore the quality, utilization efficiency and generalization ability of case text representation. Finally, the experimental results show that the proposed method has a good effect on three indicators. This paper proves that using VAE with statistical features and content features can well represent the case text.

During the experiment, it was found that the existing pre-trained language model BERT does not have a good understanding in the judicial field. Therefore, the subsequent work will design the BERT pre-training task according to the characteristics of the text in the judicial field, and propose a pre-trained judicial text representation method.

Acknowledgements The work was supported by National Key Research and Development Plan (Grant Nos. 2018YFC0830101, 2018YFC0830105, 2018YFC0830100), National Natural Science Foundation of China (Grant Nos. 61972186, 61761026, 61732005, 61672271 and 61762056), Yunnan high-tech industry development project (Grant No. 201606), Yunnan provincial major science and technology special plan projects: digitization research and application demonstration of Yunnan characteristic industry (Grant No. 202002AD080001-5), Yunnan Basic Research Project (Grant Nos. 202001AS070014, 2018FB104), and Talent Fund for Kunming University of Science and Technology (Grant No. KKSY201703005).

References

1. Gururangan S, Dang T, Card D et al (2019) Variational pretraining for semi-supervised text classification. In: Proceedings of the 57th annual meeting of the association for computational linguistics. pp 5880–5894
2. Zhao R, Mao K (2017) Fuzzy bag-of-words model for document representation. IEEE Trans Fuzzy Syst 26(2):794–804
3. Ma S, Sun X, Wang Y et al (2018) Bag-of-words as target for neural machine translation. In: Proceedings of the 56th annual meeting of the association for computational linguistics, vol 2 (Short Papers). pp 332–338
4. Trstenjak B, Mikac S, Donko D (2014) KNN with TF-IDF based framework for text categorization. Proc Eng 69:1356–1364
5. Zhu Z, Liang J, Li D et al (2019) Hot topic detection based on a refined TF-IDF algorithm. IEEE Access 7:26996–27007
6. Blei DM, Ng AY, Jordan MI et al (2003) Latent dirichlet allocation. J Mach Learn Res 3:993–1022
7. Johnson R, Zhang T (2015) Effective use of word order for text categorization with convolutional neural networks. In: Proceedings of the 2015 conference of the North American chapter of the association for computational linguistics: human language technologies, NAACL HLT 2015
8. Naz S, Umar AI, Ahmad R et al (2017) Urdu Nasta'liq text recognition system based on multi-dimensional recurrent neural network and statistical features. Neural Comput Appl 28(2):219–231
9. Kim Y (2014) Convolutional neural networks for sentence classification. In: Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP). pp 1746–1751
10. Yang Z, Yang D, Dyer C et al (2016) Hierarchical attention networks for document classification. In: Proceedings of the 2016 conference of the North American chapter of the association

for computational linguistics: human language technologies. pp 1480–1489
11. Gupta P, Pagliardini M, Jaggi M (2019) Better word embeddings by disentangling contextual n-gram information. In: Proceedings of the 2019 conference of the North American chapter of the association for computational linguistics: human language technologies, vol 1 (Long and Short Papers). pp 933–939
12. Yang M et al (2018) Investigating capsule networks with dynamic routing for text classification. In: Proceedings of the 2018 conference on empirical methods in natural language processing
13. Mikolov T, Sutskever I, Chen K, Corrado GS, Dean J (2013) Distributed representations of words and phrases and their compositionality. In NeurIPS
14. Hochreiter S, Schmidhuber J (1997) Long short-term memory. Neural Comput 9(8):1735–1780
15. Peters ME, Neumann M, Iyyer M, Gardner M, Clark C, Lee K, Zettlemoyer L (2018) Deep contextualized word representations. In NAACL-HLT
16. Vaswani A, Shazeer N, Parmar N, Uszkoreit J, Jones L, Gomez AN, Kaiser L, Polosukhin I (2017) Attention is all you need. In NeurIPS
17. Radford A, Narasimhan K, Salimans T, Sutskever I (2018) Improving language understanding by generative pre-training
18. Devlin J, Chang M-W, Lee K, Toutanova K (2019). BERT: pre-training of deep bidirectional transformers for language understanding. In NAACL-HLT
19. Kingma DP, Welling M (2014) Auto-encoding variational bayes. Stat 1050:1
20. Bowman S, Vilnis L, Vinyals O, et al (2016) Generating sentences from a continuous space[C]. In: Proceedings of the 20th SIGNLL conference on computational natural language learning, p 10–21
21. Yishu M, Yu L, Blunsom P (2016) Neural variational inference for text processing. In: International conference on machine learning
22. Yang Z, Hu Z, Salakhutdinov R et al (2017) Improved variational autoencoders for text modeling using dilated convolutions. In: Proceedings of the 34th international conference on machine learning, vol 70. JMLR. org, pp 3881–3890
23. Hoyle AM, Wolf-Sonkin L, Wallach H et al (2019) combining sentiment lexica with a multi-view variational autoencoder. In: Proceedings of the 2019 conference of the North American chapter of the association for computational linguistics: human language technologies, vol 1 (Long and Short Papers). pp 635–640
24. Zhao T, Zhao R, Eskenazi M (2017) Learning discourse-level diversity for neural dialog models using conditional variational autoencoders. In: Proceedings of the 55th annual meeting of the association for computational linguistics, vol 1 (Long Papers). pp 654–664
25. Kusner MJ, Paige B, Hernández-Lobato JM (2017) Grammar variational autoencoder. In: Proceedings of the 34th international conference on machine learning, vol 70. JMLR. org, pp 1945–1954
26. Li X, Chen Z, Poon LKM et al (2019) Learning latent superstructures in variational autoencoders for deep multidimensional clustering. In: Proceedings of international conference on learning representations
27. Paszke A, Gross S, Chintala S (2017) Automatic differentiation in PyTorch. In: Proceedings of the NIPS auto diff workshop. MIT Press
28. Kingma DP, Ba J (2014) Adam: a method for stochastic optimization. arXiv preprint arXiv:1412.6980
29. Yinhan L et al (2019) Roberta: a robustly optimized bert pretraining approach. arXiv preprint arXiv:1907.11692
30. Zhenzhong L et al (2020) ALBERT: a lite BERT for self-supervised learning of language representations. In: Proceedings of international conference on learning representations