Advances in Intelligent Systems Research, volume 133
2nd International Conference on Artificial Intelligence and Industrial Engineering (AIIE2016)

A Novel Teacher-Student Network for Sentiment Classification

Huajie Chen*, Eric Ke Wang, Feng Li and Wenli Yu
Department of Computer Science and Engineering, Harbin Institute of Technology Shenzhen Graduate School, Shenzhen, China
*Corresponding author

***Abstract*—Compared with traditional text classification, many sentiments online such as product reviews are not standard, which are concise with clear standpoints. Researchers on sentiment classification face tremendous challenges. Although various sentiment analysis systems are available, they have many operation restrictions and are still far from perfect. In this paper, we propose a novel approach, Teacher-Student Network (TSN), for automatically classifying the sentiment of reviews. Teacher-Student Network Model is composed of one teacher network and one student network. Teacher network is a Naïve Bayes model. Student network is deep neural networks model. Our approach can transfer knowledge between different models and requires less training data. Experimental results on different domain datasets show that when we employ full training data, our model can achieve similar performance to RNN(Recurrent Neural Network) model andwhen we reduce training data, our model achieve better performance than RNN.**

Keywords-sentiment classification; knowledge transfer; deep learning; RNN

I. INTRODUCTION

Sentiment analysis has become a hot research topic in thelast few decades. In sentiment analysistasks, sentiment classification is one of fundamental problems [1]. Sentiment classification task aims to classify a document or sentence into positive or negative view according to its sentiment. The applications of sentiment classification are widely spread in our life, such assentiment analysis on consumer review, social hot issues to public policy. Many researchers study on this task and have received a lot of achievements [2] [3].

Natural language processing has a long history. However, the research about sentiment classification began after 2000. Pang et al. [4] was the first paper toanalyze the performance of Naive Bayes, Maximum EntropyandSupport Vector Machines on moviereviews.The work of Pang et al. can be regarded as a baseline. The classifiers of Pang et al. belonged to supervised learning. Sometimes, labeling data is expensive. Therefore, some unsupervised learning approaches were considered. Turney [5] (1) used apart-of-speech tagger to identify phrases in the input text that contain adjectives or adverbs, (2) estimate the semanticorientation of each extracted phrase, (3) predicted the sentiment by theaverage semantic orientation of thephrases. Minqing Hu et al. [6] made further research. They created emotion vocabulary with an iteration process then identified opinion sentences. Recent year, researchers came up with many interested models. In order to deal with domain-dependent problem, Fangzhao Wu et al. [5] proposed a domain adaptation approach which can exploit sentiment knowledge from multiple source domains. They first extract both global and domain-specific sentiment knowledge from the data of multiple source domains using multi-task learning.Then,they transfer them to target domain with the help of words' sentiment polarity relations extracted from the unlabeled target domain data.This approachimproved the performance of cross-domain sentiment classification. Some researchers made contribution on fine-grain sentiment analysis. For instance, Linlin Wang et al. [8] proposed a sentiment and aspect extraction model based onRestricted Boltzmann Machines. It reflects the generation process of reviews by introducinga heterogeneous structure into the hiddenlayer and incorporating informative priors. Later on, with the rising of research on deep learning, many researchers employed deep learning method to deal with natural language processing tasks. The main models are based on conventional neural networks [9] and recurrent neural networks [10].

Although these approaches have achieved a great advancement, they have some shortcomings. Linguistics methods require deep-going research about language. It's hard for general scholars to join in. What is more, the features discovered in one language maybe difficultly generalizeto another language or other domains. Traditional machine learning method usual is based on bag of words model (BOW). BOW regards a sentence or document as a disordered word set which ignore the dependence and similarity among words. The position of word is a clue to predict sentiment. And different sentence can express same meaning by using similarity words. Due to the explosion of big data and the promotion of computing capability, deep learning models achieved success. On the contrary, these are barriers for deep learning. Not all domains have such full data and it is hard for human to understand the meaning of complicated model.

We think that traditional machine learning models and deep learning models are not independent. The advantage of traditional machine learning algorithm is easy to understand and require less training data than that in deep learning. Deep learning algorithm is good at describing the objective world.Why not combine them together? So we propose a teacher-student networks model. First, build a teacher network based on Naïve Bayes. Then, leverage a teacher network to teach a student network which is based on c deep neural networks model (DNNs). The final goal is to attain an efficient student networks. Our experiment approved that this way can reduce the demand of huge training data and can promote the performance of classification.

II. RELATED WORK

Our model is consisted of a teacher network based on Naïve Bayes and a student network based on DNNs. The model transfersthe knowledge learned from teacher network into student networks. Our work is related to the following threeparts. We will introduce them briefly.

DNNs were responsible for major breakthroughs in image classification.Kaiming He et al. [11] achieved 4.94% top-5 test error on the ImageNet 2012 classificationdataset.This result even beat human-level performance (5.1%, [12]) on this dataset.Inspired by the outstanding performance of DNNs in image classification, researchers try to use DNNs to deal withtext classification tasks.

A. Word Embedding

In classification of image task, the input of DNNs is an image which is composed of pixels. When it comes to sentiment classification, we should change word into word embedding. This idea can date back to 2003. YoshuaBengio [13] proposed aneural probabilistic language model. The input of the network is a sequence $w_{t-n+1} \cdots w_{t-1}$ of words ($w_k \in$ V, V is a set of words). The objective is to learn a model to predict the probability of next word w_t . C can be regarded as vocabulary which records embedding of each word. The model is train by gradient descent.After training, we not only get a prediction model but also learn a distributed representation for words. Tomas Mikolov et al. [14][15][16] proposed CBOW model and Skip-gram model to train word embedding.

- In CBOW model, the idea is using context words to predict current word. The objective function is:

$$\max(\mathcal{L}) = \max\left[\sum_{\mathrm{w}_i \in C} log\, P(\mathrm{w}_i | w_{i-2} w_{i-1} w_{i+1} w_{i+2})\right]$$

- In Skip-gram model, the ideal is using current to predict context words.The objective function is:

$$\max(\mathcal{L}) = \max\left[\sum_{\mathrm{w}_i \in C} log \prod_{j=-1, j\neq 0}^{2} P(\mathrm{w}_{i+j} | w_i)\right]$$

Ciscorpus. $\mathrm{w}_i$is current word. $\mathrm{w}_{i+j}$is neighbor word of w_i.

In our model, we will use CBOW model and Skip-gram model to pre-train word embedding. When we train our DNNs model, we adjust the word embedding simultaneously in the same way with YoshuaBengio [13]. By this way, the word embedding can capture syntactic, semantic and sentimentalrelationships.

B. Deep Networks

The main deep neural networks model contains two types, convolution neural networks [9] and recurrent neural networks.

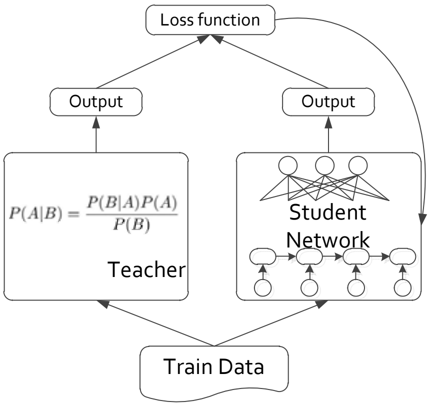

FIGURE I. THE STRUCTURE OF TEACHER-STUDENT NETWORKS.

In the CNNs model, a convolutional layer is applied to the sentence matrix s. $s = [v(w_1), \cdots, v(w_k), \cdots, v(w_L)]$, $\mathrm{s} \in R^{d \times L}$(d is the dimension of word embedding, L is the length of sentence). $v(w_k)$is the word embedding of w_k, $v(w_k) \in R^d$. The third layer is a max-pooling operation following by softmax output with full connected. The model was valued on various classification datasets, including question dataset, subjectivity dataset and sentiment dataset. The CNN architecture achieves very good performance across datasets.

Recurrent Neural Networks (RNNs) are popular models that have shown great promises in many NLP tasks.The most commonly used type of RNNs is LSTM, which are much better at capturing long-term dependencies [17]. The same with CNNs, the input for RNNs is word embedding.

C. Transfer Learning

We propose transfer knowledge method between different networks. This idea is the same with that in transfer learning domain. With the exploding increase of data, we can mine the knowledge behind big data. Many classification algorithms can be regarded as learning how to divide future space. It assumes that the training data and the testing data have the same distribution. This limits the capability of algorithms. For instance, we have a classification task in one domain of interest, but we only have sufficient training data in another domain. Where the latter data may be in a different feature space or follow a different data distribution. The objective of transfer learning is to transfer knowledge learned from one data set to another data set. In such cases, knowledge transfer, if done successfully, would greatly improve the performance of learning by avoiding much expensive data labeling efforts [18]. We think knowledge cannot only be transfer among different data but also among different algorithms. So, we proposed teacher-student model. In the III section, we will introduce how to transfer knowledge from Naïve Bayes to RNNs by teacher-student model.

III. TEACHAER-STUDENTNETWORKS MODEL FOR KNOWLEDGE TRANSFER

The structure of teacher-student networks is shown on figure I. The advantage of traditional machine learning algorithm is easy to understand and requrie less training data than that in deep learning. Deep learning algorithm is good at describing the objective world. It's proved that neural nets can compute any function. The advantages of deep learning models are the weaknessof traditional machine learning models. The opposite is also true. So we propose a teacher-student networks model to combine them together.

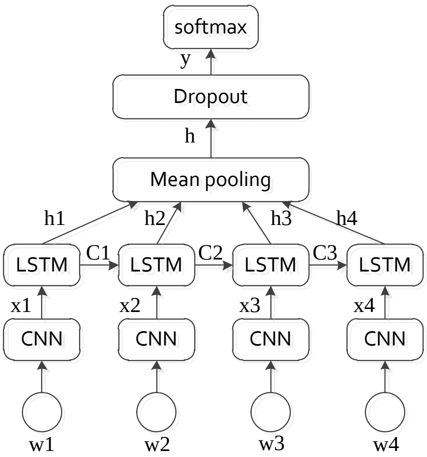

FIGURE II. THE STRUCTURE OF STUDENT NETWORKS.

A. *Teacher Network*

In teacher network, we chose Naïve Bayes model. Although its conditional independence assumption is not always true in real life, Naïve Bayes take a good performance in classification task. First, we learned a feature set $\{f_1, f_2, \cdots, f_m\}$ from train set by chi-square analysis, m is the number of features. Chi-square analysis can be used to evaluate the relationship between word and class. It assumes that if a word often occurs in class i, the word will have high relationship with class i.

$$\chi^2(w,i) = \frac{n * F(w)^2(F(w,i) - F(i))^2}{F(w)(1 - F(w))F(i)(1 - F(i))}$$

n is the number of total document. $F(w)$is the number of document that contain word w. $F(w,i)$ is the number of document in class *i* that contain word w. $F(i)$ is the number of document in class *i*.

Let $n_i(d)$ represent the frequency of feature f_i in document d. Then, each document d can be represent by a document vector $\vec{d} = (n_1(d), n_2(d), \cdots, n_m(d))$. Naïve Bayes model assign document d the class $c^* = arg\max_c P(c|d)$. $P(c|d)$is calculate by Bayes' rule.

$$P(c|d) = \frac{P(c)P(d|c)}{P(d)}$$

In sentiment classification:

$$P(c|d) = \frac{P(c)P(\prod_{i=1}^{m} p(f_i|c)^{n_i(d)})}{P(d)}$$

B. *Student Network*

In student network, we build DNN model that is composed of a CNN feature extraction part and a LSTM memory part. The CNN part is similar with the work of [9]. The LSTM part plays a role on memory. As we know, the meaning of word is related to its context. 'Short' is a negative word when we are talking about battery life. However, it a positive word for the shutter of camera. So LSTM part is designed to remember context information. The structure of student networks is shown on figure II.

w_t is the t-th input. The activation of CNN layer x_t is calculated as follows:

$$x_t = [x_t^1, x_t^2, \cdots x_t^n]$$

$$x_i^k = f\left(U_{cnn} \cdot w_i^{k\sim k+l} + b_{cnn}\right)$$

U_{cnn} isconvolution kernel, $U_{cnn} \in R^l$. *l*is the size of convolution kernel. $w_i^{k\sim k+l}$ is the subembedding of w_i from index *k* to *k*+*l*. b_{cnn} is bias.$x_t^k \in R$.

Concretely, each stepof LSTM takes inputx_t, C_{t-1}, h_{t-1} and produce h_t, C_tt via the following calculations [19]:

$$i_t = \sigma(W^i x_t + U^i h_{t-1} + b^i)$$

$$f_t == \sigma(W^f x_t + U^f h_{t-1} + b^f)$$

$$o_t = \sigma(W^o x_t + U^o h_{t-1} + b^o)$$

$$\hat{c}_t = \tanh(W^c x_t + U^c h_{t-1} + b^c)$$

$$c_t = f_t \cdot c_{t-1} + i_t \cdot \hat{c}_t$$

$$h_t = o_t \cdot \tanh(c_t)$$

W^i, U^i, W^f, U^f,W^c, $U^c$are weight matrices. b^i, b^f, b^o, b^c are bias vectors.

We also add dropout operation. This architecture makes model more universal. Dropout is a simple way to prevent neural networks from overfitting. Our TSN model has a large number of parameters which make it very powerful to capture information. On the contrary, it's more possible to be overfitting. Dropout is a technique for addressing this problem. The input of Dropout layer is h and the output is y.

$$r_j = Bernoulli(p)$$

$$\hat{h} = r \cdot h$$

$$z = W\hat{h} + b$$

$$y = f(z)$$

$p$is the dropout rate. W is weight matrices. b is bias. f is aviation functions

C. Transfer Learning

A very simple way to improve the performance of classifier is train many different classifiers then chose the class that most classifiers vote as final result. This way can reduce the error effectively. For example, we train 21 independent classifiers. Each classifier's error rate is 0.3. More than 11 classifiers make mistakes then the final result is incorrect. So the error rate is $\sum_{i=11}^{21} C_{21}^{i}(0.3)^{i}(0.7)^{21-i} \approx 0.026$. It's smaller than individual classifier. This is the based idea of ensemble classifiers. Howevermore classifier means more computational overhead, especially if the individual classifier is a complex model. Online application of sentiment analysis would require high time limited and low computational overhead. So we propose a novel way to combine different classifiers.

As showing on Figure I, our model is composed of two parts. First, we use training data to train a Naïve Bayes model. This model likes a teacher. It will teacher the student network how to classify. The input of student work is the same with student network. When training student work, we not only concern about the true label, called hard target, but also concern about the output of teacher, called soft target. Make the student network learning the output of teacher. In order to learning both targets simultaneously, we design two cost functions for each part then combine them together.

$$\text{Cost Function} = (1 - \pi) \times \text{Cost Funtion}_1 + \pi \times \text{Cost Funtion}_2$$

$$\text{Cost Funtion}_i = -\frac{1}{n}\sum_x y\ln a + (1 - y)\ln(1 - a)$$

for i=1, 2

In equation (5), π is used to keep balance between hard target and soft target, called weight coefficient. We use cross entry as cost function in detail. In equation (6), n is the number of train instance. a represents the predict value. y is soft target or soft target for i=1, 2 respectively.

IV. Experiments

A. Dataset and Experimental Measure

We tested our model on various domain sentiment datasets. Summary statistics of the datasets are in Table I.

TABLE I. SUMMARY STATISTICS FOR THE DATASETS

Data	***c***	***l***	**|V|**	***Train***		***Test***	
				Pos	*Neg*	*Pos*	*Neg*
Hotel	2	88	23762	2500	2500	500	500
Car	2	24	17731	2428	2840	810	935
Food	2	15	11959	7000	7000	500	500

L：Language of dataset.c: Number of target class. l: the average length of s entence. |V|: Number of vocabulary. Train Pos: Number of positive train ins tance. Train Neg: Number of negative train instance. Test Pos: Number of p ositive test instance. Test Neg: Number of negativetest instance.

- Hotel: Hotel review with one sentence per review.Classification involves detecting positive or negative reviews.
- Car: Hotel review with one sentence per review. This dataset was collected from news, weibo and forum about car. Classification involves detecting positive or negative reviews.
- Food: Food review with one sentence per review. This dataset was collected from some platform for ordering a meal. Classification involves detecting positive or negative reviews.

We measure the performance of different models in different dataset by accuracy (Acc) and F_1 value.

$$Acc = \frac{PP + NN}{PP + PN + NP + NN}$$

$$F_1 = \frac{2precision \times recall}{precision + recall}$$

$$precision = \frac{PP}{PP + PN}$$

$$recall = \frac{PP}{PP + NP}$$

TABLE II. THE CONFUSION MATRIX FOR CLASSIFICATION TASK

		Gold Standard	
		Pos	Neg
Predicted	Pos	PP	PN
	Neg	NP	NN

B. Performance Evaluation

We experiment with several models.

- Naïve Bayes: Our baseline model. As introduce in section III, use chi-square analysis to select feature words. Then calculate the probabilities of a instance according to different class.

- RNN: This model contains It contains 6 layers. Except for input layer and output layer, it contains one LSTM layer, one pooling layer, one dropout layer and one softmax layer.
- TSN: This is our model. The teacher network is based on Naïve Bayes. The student network is based on DNN.

TABLE III. RESULTS OF OUR TSN MODELS AGAINST OTHER METHODS

		Bayes	RNN	TSN
Hotel	Acc	0.631	0.862	**0.87**
	F_1	0.4349	0.8547	**0.8752**
Car	Acc	0.9014	**0.962**	0.958
	F_1	0.9038	**0.9597**	0.9546
Food	Acc	0.631	**0.872**	0.87
	F_1	0.4349	**0.8735**	0.8687

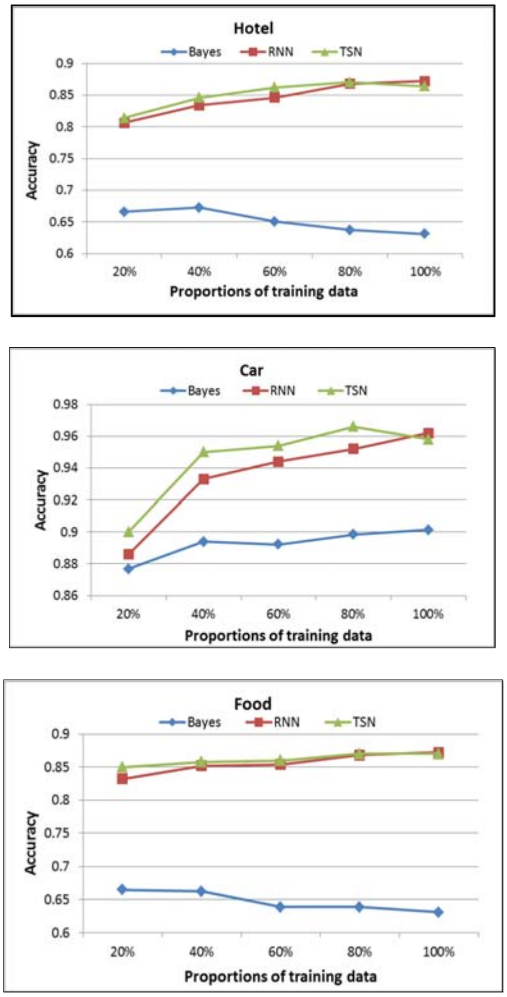

FIGURE III. RESULTOF OUR TSN MODELS AGAINST OTHER METHODS USING DIFFERENT PROPORTIONS OF TRAINING DATA

Result was shown on Table III. Our model got the best performance in Hotel dataset and got comparable performance on Car and Food dataset. What's more, Teacher-Student Network model requires less training data. As shown in Figure III, we employed different proportionsof training data to train models,our model can keep better performance than recurrent neural networks model and Naïve Bayes model.

V. CONCLUSION

This paper presents a knowledge transfer method for sentiment classification from traditional machine learning algorithm to deep learning algorithm. Our approach consists of two steps. First, we use Naïve Bayes to train a teacher model. Second, we transfer the knowledge learned from teacher model to student model. Student model is built based on CNNs and RNNs. CNNs is used for extracting feature and RNNs is used for recording memory.Experimental results on different domain datasets show that when using full training data, our model can get comparable performance than recurrent neural networks model and when we reduce training data, our model can keep best performance. These results indicate that the Teacher Student Network model scheme is both effective and feasible.

ACKNOWLEDGMENTS

This research was supported in part by National Natural Science Foundation of China (No.61572157), Guangdong Province Natural Science Foundation (grant No.2016A030313660), Shenzhen Municipal Science and Technology Innovation Project (No.JCYJ20160428092427867, No.JCY2015040316192350, No.JCYJ20150617155357681and No.JSGG20141017150830428). The authors thank the reviewers for their comments.

REFERENCES

[1] Xinjie Zhou, Xianjun Wan and Jianguo Xiao."Cross-Lingual sentiment classification with bilingual document". In Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics,Berlin, Germany, pp. 1403–1412,August 2016.

[2] Bo Pang and Lillian Lee. "Opinion mining and sentiment analysis". Foundations and trends in information retrieval, 2008(2), pp. 1–135.

[3] Bing Liu. "Sentiment analysis and opinion mining". Synthesis Lectures on Human Language Technologies, 2012, pp. 1–167.

[4] Pang B, Lee L, Vaithyanathan S. Thumbs up?: sentiment classification using machine learning techniques. In Proceedings of the 2002 Conference on Empirical Methods in Natural Language Processing. 2002, pp. 79-86.

[5] Peter D. Turney. Thumbs Up or Thumbs Down? Semantic Orientation Applied to Unsupervised Classification of Reviews. In Proceedings of the 40nd Annual Meeting of the Association for Computational Linguistics. 2002, pp. 417-424.

[6] Minqing Hu and Bing Liu. Mining and Summarizing Customer Reviews. In Proceeding KDD'04 Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining. 2004, pp. 168-177.

[7] Fangzhao Wu and Yongfeng Huang. Sentiment Domain Adaptation with Multiple Sources. In Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics,Berlin, Germany, pp.301–310,August 2016.

[8] Linlin Wang, Kang Liu, Zhu Cao, Jun Zhao and Gerard de Melo."Sentiment-Aspect Extraction based on Restricted Boltzmann Machines". In Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics. 2014, pp. 616–625.

[9] Yoon Kim. “Convolutional Neural Networks for Sentence Classification”. In Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP). 2014, pp. 1746-1751.

[10] G Mesnil, T Mikolov, MA Ranzato, Y Bengio. “Ensemble of generative and discriminative techniques for sentiment analysis of movie reviews”. In the 3rd International Conference on Learning Representations (ICLR2015). 2015.

[11] Kaiming He, Xiangyu Zhang, Shaoqing Ren, Jian Sun. “Delving deep into rectifiers: Surpassing human-level performance on imagenet classification”. In The IEEE International Conference on Computer Vision (ICCV), 2015, pp. 1026-1034.

[12] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh,S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein,et al. “Imagenet large scale visual recognition challenge”.arXiv:1409.0575, 2014.

[13] Yoshua Bengio, Réjean Ducharme, “Pascal Vincent and Christian Jauvin. A Neural Probabilistic Language Model”. Journal of Machine Learning Research. 3(2003), pp. 1137–1155.

[14] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. “Efficient Estimation of Word Representations in Vector Space”. In Proceedings of Workshop at ICLR, 2013.

[15] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg Corrado, and Jeffrey Dean. “Distributed Representations of Words and Phrases and their Compositionality”. In Proceedings of NIPS, 2013.

[16] Tomas Mikolov, Wen-tau Yih, and Geoffrey Zweig. “Linguistic Regularities in Continuous Space Word Representations”. In Proceedings of NAACL HLT, 2013.

[17] Shujie Liu, Nan Yang, Mu Li and Ming Zhou. “A Convolutional Neural Network for Modelling Sentences”. In Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics,USA, Maryland, pp: 655–665, June 2014.

[18] Sinno Jialin Pan, Qiang Yang. “A Survey on Transfer Learning”. IEEE Transactions on Knowledge and Data Engineering. 22(2010), pp: 1345–1359.

[19] Deng Cai , Hai Zhao. “Neural Word Segmentation Learning for Chinese ”. In Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics, pages 409–420, Berlin, Germany, August 7-12, 2016.