2017 International Conference on Security, Pattern Analysis,
and Cybernetics (SPAC)

Attention-based LSTM-CNNs for Uncertainty Identification on Chinese Social Media Texts

Binyang Li
School of Information Science and Technology
University of International Relations
Beijing, China
byli@uir.edu.cn

Kaiming Zhou
School of Information Science and Technology
University of International Relations
Beijing, China
kmzhou@uir.edu.cn

Wei Gao
School of Information Management
Victoria University of Wellington
Wellington, New Zealand
wei.gao@vuw.ac.nz

Xu Han*
College of Information Engineering
Capital Normal University
Beijing, China
hanxu@cnu.edu.cn

Linna Zhou
School of Information Science and Technology
University of International Relations
Beijing, China
lnzhou@uir.edu.cn

***Abstract*—Uncertainty identification is an important semantic processing task, which is crucial to the quality of information in terms of factuality in many techniques, e.g. topic detection, question answering. Especially in social media, the texts are written informally which are widely used in many applications, so the factuality has become a premier concern. However, existing approaches that still rely on lexical cues suffer greatly from the casual or word-of-mouth peculiarity of social media, in which the cue phrases are often expressed in sub-standard form or even omitted from sentences. To tackle these problems, this paper proposes the attention-based LSTM-CNNs for the uncertainty identification on social media texts, named ALUNI. ALUNI incorporates attention-based LSTM networks to represent the semantics of words, and convolutional neural networks to capture the most important semantics of uncertainty for identification. Experiments are conducted on both Chinese Weibo and news datasets, and 78.19% and 73.95% of F1-measure scores are achieved with 11% and 3% improvement over the baseline, respectively.**

Keywords—LSTM, CNN, uncertainty identification, social media

I. INTRODUCTION

“Uncertainty - in its most general sense - can be interpreted as lack of information: the receiver of the information (i.e., the hearer or the reader) cannot be certain about some pieces of information” [1]. The identification of uncertainty is significant to the trustworthiness of many natural language processing techniques and applications, such as question answering, information extraction, and so on [2].

The CoNLL-2010 Shared Task aimed at identifying uncertainty in biological papers and Wikipedia articles written in English [3] [4]. Most participants utilized linguistics features, e.g., lexical cues, Part-Of-Speech (POS), to detect the uncertain sentences from the texts.

Recently, with the growing popularity of social media, there exist more and more texts consisting of casual or word-of-mouth expressions. The quality of information in social media in terms of factuality has become a premier concern [5]. The generation and propagation of uncertain information will lead to rumor flooding among social media and even influence the real world. For example, the 2011 London Riots occurred owing to the spread of uncertain in-formation among social media, such as Twitter or Facebook. Therefore, uncertainty identification, i.e., identifying uncertain sentences is becoming increasingly critical for users to synthesize information to derive reliable interpretation.

However, unlike the biological papers and Wikipedia articles, the texts in social media are usually short and informal. Due to the word count limit and casual expression, many cue phrases are expressed in substandard shape or even omitted from sentences. In this case, the uncertain semantics will be implicitly conveyed by the whole sentence rather than explicitly by cue phrases. Existing approaches based on cue phrases for uncertainty identification are ineffective for social media texts, and they are also not good enough for formal text uncertain identification. It is noteworthy that in the CoNLL-2010 Shared Task, the participants all achieved better results on biological dataset than wiki dataset. It indicated the more formal the article is, the easier it is to judge the sentence uncertainty. As a result, uncertainty identification on Chinese social media texts has become a big challenge which needs more semantics information to solve.

We tried to judge the uncertainty of the Chinese text of social media based on semantics, so we turned to deep learning which could express the semantics of words and sentences well. Bahdanau et al. apply the RNN with attention mechanism to machine translation [6], their model makes the words’ semantics and the relation between words in both languages clearer. Kim utilizes CNNs to classify sentences and achieves good results [7], which shows CNNs have a unique advantage in both image and text classifying issues. Considering these above researches we decided to combine the two model structures to solve uncertainty identification problem.

This paper proposes Attention-based Long Short-Term Memory-Convolutional Neural Networks (LSTM-CNNs) for Uncertainty Identification on social media texts, named ALUNI. ALUNI incorporates attention mechanisms into LSTM networks to represent the semantics of the context in a sentence, and uses CNNs for the uncertainty identification. Benefitted from the attention mechanisms, the key elements of sentences can be highlighted and the hidden semantics can be captured, which will enable us to detect uncertainty based on the context of the whole sentence instead of depending on the cue-phrases.

*Corresponding author

The contribution of this paper is as follows:

- We propose attention-based LSTM-CNNs for uncertainty identification from social media texts, which can indiscriminately focus on the words regardless of cue-phrases or not that have decisive effect on uncertain semantics without using extra knowledge or external NLP components.
- We conduct experiments on both Chinese Weibo and news datasets, and 78.19% and 73.95% of F1-measure scores are achieved with about 11% and 3% improvement over the baseline, respectively.

The remainder of the paper is organized as follows: Section 2 summarizes the related work. Section 3 describes our proposed methods. Section 4 presents our experimental results, and Section 5 concludes the paper.

II. Related Work

Uncertainty identification has attracted lots of attention in the NLP area. The CoNLL-2010 Shared Task aimed at detecting uncertainty cues in biological papers and Wikipedia articles in English [3]. Recently, a special issue of the journal Computational Linguistics (Vol. 38, No. 2) was dedicated to detecting modality and negation in natural language texts [8]. Most of the existing approaches can be classified as rule-based ones [9][10] and machine learning methods, such as Medlock' s research on biomedical literature[11], Fernandes et al.'s work [12], Li et al.'s work [13], Tang et al.'s work [14] and Zhang et al.'s work [15] at CoNLL2010, which usually applied various supervised approaches on the annotated corpus to incorporate different types of linguistic features such as Part-Of-Speech (POS) tags, word stems, n-grams, and so on. Velldal in 2010 constructed a cue-lexicon to describe the context, which was applied into a binary classifier for detection [16].

The above approaches mainly focused on English, we are aware of one study aiming at Chinese texts which Ji et al. proposed in 2010 [17]. A supervised method with lexical features was proposed and evaluated by the annotated corpus consisting of Chinese news data.

Regarding the uncertainty identification from social media texts, Wei et al. firstly con-ducted an empirical study on uncertainty identification based on social media data [5], and the features beyond plain texts were accounted, such as tweets amount, relationship, etc. Veronika [18] [19] proposed to use lexical, morphological, syntactic, semantic and discourse-based features into a supervised classifier for detecting uncertainty in Hungarian social media texts.

Recently, deep learning has become a hotspot in natural language processing, especially in the text classification. CNNs, widely used in the field of machine vision at first, has also been used in Natural Language Processing in recent years. Zhang and Wallace conducted a sensitive analysis of one-layer CNNs [20]. The analysis showed though elaborate settings the CNNs model got the state-of-the-art in most of the sentence classification tasks. On the other hand, Yang et al. propose hierarchical attention networks for document classification [21] which impresses us a lot. Their work not only improves the accuracy of the document classification, but also makes people understand how the attention mechanism work though the attention weight visualization.

Our approach has three major differences from previous work: (1) Our model does not use any extra knowledge or external systems, but only word embedding; (2) We use the attention mechanism to represent the sentence and generate the semantic focus; and (3) Regarding the experimental datasets, we are the first to construct the Chinese social media corpus for evaluating uncertainty identification.

III. Attention-based LSTM-CNNs for Uncertainty Identification

In this section, we will introduce our model, namely ALUNI. Figure 1 illustrates the architecture of ALUNI, which consists of three components: word representation, words encoding, and convolutional classifier.

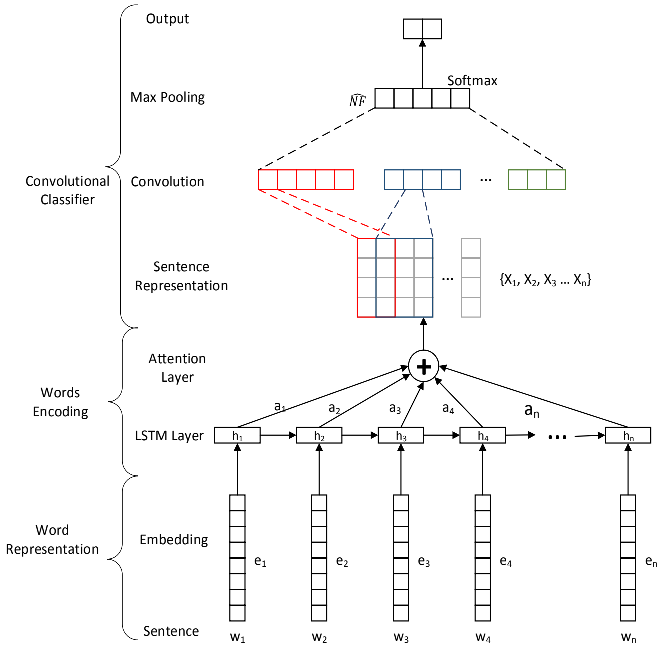

Fig. 1. Architecture of our model.

- Word representation component: It inputs sentence to this model and map each word into a k-dimensional vector.
- Words encoding component: It utilizes LSTM to get high-level features for a more accurate word semantics representation, and produces a weight vector based on attention mechanisms, then multiplies word-level features by the weight vectors to highlight the words which are important for uncertainty identification.
- Convolutional classifier component: It extracts N-gram features of the sentence and picks the most important part for uncertainty identification. After a full connection layer, there is a softmax function to output the prediction result.

These components will be introduced in detail in this section.

A. Word Representation

We pre-trained an embedding matrix $M \in \mathbb{R}^{k|V|}$ to translate words into word vectors, where Va fixed-sized vocabulary, and k is the dimension of word embedding. Suppose there is an input sentence $S = \{w_1,\ w_2, \dots,\ w_n\}$ with n words, when searching for word vectors, via looking up embedding matrix, each word w_i is converted into a k-dimensional real-value vector representation $e_i \in \mathbb{R}^k$. But there may be a little words out of our vocabulary (OOV). To solve the problem that rare words cannot be represented by vectors, we discarded these words. As OOV rarely occurs and it's also very common for people skip the difficult words which makes little effect on understanding the whole sentence. Converting a sentence word into a vector, we set the maximum length of a sentence, the sentences less than the maximum length will be completed by zero vector

B. Words encoding

The sentence representation component consists of two layers: LSTM layer and attention layer.

1. LSTM Networks

LSTM networks were firstly proposed to overcome gradient vanishing problem, and an adaptive gating mechanism is introduced to decide the degree to which LSTM units keep the previous state and memorize the extracted features of the current data input [22]. Lots of LSTM variants have been proposed, like Sundermeyer's improvement applied to language model [23] and Yao et al.'s research about depth-gated recurrent neural networks [24].

We apply a variant of LSTM networks, proposed by Graves [25], to represent the complete semantics of a sentence. In our LSTM-based recurrent neural networks, the inputs are word vectors $\{e_1,\ e_2, \dots,\ e_n\}$ and the outputs are hidden states $\{h_1, h_2, \dots,\ h_n\}$. There are three types of gates: one input gate i, one forget gate f, and one output gate o. Given the current input e_i together with the cell state generated by previous cells, all these gates will decide to what degree we should take the current input and forget the memory stored before. Our LSTM can be computed by the following equations:

$$\begin{pmatrix} i \\ f \\ o \\ g \end{pmatrix} = \begin{pmatrix} sigm \\ sigm \\ sigm \\ tanh \end{pmatrix} (W \cdot [D(h_{i-1}), e_i] + b) \tag{1}$$

$$c_i = f \odot c_{i-1} + i \odot g \tag{2}$$

$$h_i = o \odot \tanh(c_i) \tag{3}$$

where D is a dropout operation, $sigm$ is the sigmoid function, $tanh$ is the tanh function, W, b are the parameters that need to learn, $\odot$ is the elementwise multiplication.

In this way, the current cell state c_i will be generated by calculating the weighted sum using both previous cell state and current information generated by the cell. As we know the same word in different contexts may have different meanings, only by incorporating the contextual information into the representation of the word can we express the word meaning exactly. LSTM networks encode every words and takes previous information into those words, so that each hidden state h_i can represent the meaning of a word in the specific sentence more accurately.

2. Attention

Since not all the words in a sentence contribute equally to uncertainty identification, we adopt attention mechanisms to generate better sentences representation with semantic focus.

We calculate the attention α_i for each word w_i as follows:

$$\alpha_i = \frac{\exp(v^{\mathrm{T}} \tanh(W_r h_i + b_r))}{\sum_i \exp(v^{\mathrm{T}} \tanh(W_r h_i + b_r))} \tag{4}$$

where v, W_r, b_r are model parameters that need to learn. Compared to other attention models that sum up the product of the hidden states and their respective weights [21], we concatenate them so that all the hidden states sequences generated by word vectors can be maintained, which can be used in the following CNNs component to obtain the most important features from all the words hidden states vectors.

$$X_{1:n} = \alpha_1 h_1 \oplus \alpha_2 h_2 \oplus \dots \oplus \alpha_n h_n \tag{5}$$

Benefitted from the attention mechanisms, the key elements in sentences can be highlighted and richer semantics can be conveyed by the encoded words (an instance will be shown in our experiment). Then $X_{1:n}$ will be as the input of the convolutional neural networks.

C. Convolutional Classifier

Convolution neural networks (CNNs) are widely used and have achieved state-of-the-art results in many classification tasks. Collobert et al. [26] proposed a sentence-based network with CNNs. Inspired by the CNN architecture of Collobert et al. [26], we design our convolution neural networks to determine whether a sentence is certain or not.

We take the sentence representation $X_{1:n}$ that carries the hidden state of each word as the input, and the result of the uncertainty identification, i.e., $R \in \{certain,\ uncertain\}$, as the output.

Our convolution neural networks involve a filter $f \in \mathbb{R}^{l \times k}$, which is applied to a window of length l sliding over the hidden states of LSTM networks to produce a new feature. For example, the i^{th} new feature NF_i can be computed by the following equation:

$$NF_{l_i} = relu(f \cdot X_{i:i+l-1} + b_{nf}) \tag{6}$$

where b_{nf} is a bias parameter, $relu$ is called rectified linear units following the specified transformation.

After we calculate all the new features in turn, we can get a NF_l sequence:

$$NF_l = \{NF_{l_1}, NF_{l_2}, \dots, NF_{l_{n-l+1}}\} \tag{7}$$

Then we adopt a max pooling operation over the NF_l sequence to get the maximum value of the new feature $\widehat{NF}_l = \max\{NF_l\}$. In our experiment, we also tried an average pooling, but the results proved that the max pooling was better. It is because that the average pooling can capture more comprehensive features, while the max pooling can capture the most important features that are more useful for our task. We then use filters with different windows sizes of sentence l to get multiple features, and connect all $\widehat{NF}_l$ to get $\widehat{NF}$. With the CNNs, we extracts N-gram features of the sentence, *N* is the size of the sliding window. As the words for judging uncertainty are given more weight in the attention mechanism, max pooling can easily help us to focus on these words. So $\widehat{NF}$, the output of CNNs, is an effective representation for uncertainty identification.

Finally, after a full connection neural network layer, we apply a softmax layer to produce the output.

$$p = \text{softmax}\left(W_p\widehat{NF} + b_p\right) \tag{8}$$

We use the cross-entropy against the correct labels as training loss.

$$L = -\sum_{S}\sum_{C} T_c(S) \cdot \log\left(p_c(S)\right) + \lambda l_2 \tag{9}$$

where C is the binary class of the sentence S, $T_c(S)$ is the binary value indicates whether the sentence S belongs to class c, while $p_c(S)$ is the prediction result of sentence S. l_2 is the $L2$ norm for regularization, it is a squares sum of all parameters, λ is a parameter to decide how much l_2 should be calculated into the loss.

When we constructed our model, we tried to simulate people's thought process to identify the sentence uncertainty. Firstly, facing such a problem, people usually read through the sentence to understand each word and then the whole sentence. In our model, LSTM networks finish this job. Then, to identify the uncertainty of the sentence, people often pay attention to some useful words, it looks like our attention mechanism. Finally, people generally give out the result based on several prominent short sentences which carry certain or uncertain semantics. In our model, it is considered more comprehensively, as CNNs' different size sliding windows scan all-over the sentence. We think that the model is reasonable, and expect that this model will achieve or even beyond the performance of mankind in uncertainty identification.

IV. Experiments

In this section, we will introduce our experiments, including the experiment setup, dataset description, experimental results display, and the analysis.

A. Datasets and Experiment Setup

Experiments are conducted on both Chinese social media dataset and news dataset.

The Chinese news dataset is the experiment data in Ji, et al.'s research, these data was collect form Baidu News. They searched and selected the first 50 news reports and then they picked out the paragraphs with cue words. After sorting out the data, the paragraphs were cut into sentences. The sentences were annotated following the annotation schema of CoNLL-2010 Shared Task dataset, according to the clue words rule. It's the only available Chinese corpus for uncertainty identification, which is consisted of 10,000 sentences, including 2,858 uncertain sentences.

The social media dataset was collected from Sina Weibo during the Shanghai Expo. We cleaned the data and extracted plain text in the tweets. We randomly selected 30,000 sentences as our experiment dataset, then we manually annotated these sentences whether they were certain or not. The standard of the annotation is fully according to the semantics. In order to make the annotation labels credible, each sentence was judged by two people. If the results were different, the third person would make a final decision. We calculated the kappa value of label results, the value was 0.7568 which showed the data markers basically agreed with each other and the dataset was reliable.

TABLE I. Overview of the Datasets.

	News dataset	Social media dataset
# of Sentences	10,000	30,000
Average words	30.72	21.14
Uncertain sentences	2,858	11,071
Uncertain percentage	28.58%	36.90%
Uncertain sentence cues	5084	11618
Average cues	1.79	1.05

The overview of our experiment datasets shows in Table I, social media dataset has more sentences than News dataset and more uncertain sentences with fewer cue words. The average length of Social media sentences is far shorter than News sentences. It comes out that the informal expression of social media makes sentences incomplete and complex. In our experiment, we randomly chose 8,000 sentences from news corpora with 2,248 uncertain sentences, and 24,000 sentences from Weibo dataset with 8,798 uncertain sentences as training set, respectively, and the others were used as test set.

In the experiment, we used jieba[1] for the Chinese word segmentation. Jieba is one of the best Python Chinese word segmentation module, which is accurate and easy to use. We used gensim[2], a python package, to produce word vectors with deep learning via word2vec's skip-gram model presented by Mikolov et al. in 2013 [27]. We used 30 GB Chinese texts, including Shanghai Expo Weibo and Ji, et al.'s Chinese news, to train word vector of 100-dimension. We also randomly selected 1,000 sentences as development set, on which the hyper-parameters of our model were tuned. Note that we do not remove the punctuations, as the punctuations are also meaningful. For example, the "?" in a sentence usually indicates uncertain.

During training, we used a two-layer RNN with attention mechanism, (We didn't use deeper networks because the two-layer perform well and it was also faster) and set the length of window as 3, 4, 5, and 6 in our CNNs to extract the features. (as 3 to 6 consecutive Chinese words usually express a clear semantics.) To avoid overfitting, we set the dropout parameter as 0.5.

In our experiment, we chose the work of Ji et al. [17] as our Baseline1, which was the first uncertainty detection method on Chinese texts. We also selected and redesigned Veronika's work [19] as Baseline2, which was originally proposed for Hungarian social media texts. Both of on the cue-phrase features. To better investigate our system, we also evaluated various configurations of our neural networks. We adopt the official evaluation metrics of CoNLL 2010.

B. Results and Analysis

In our experiment, we trained our model for a long time and verified instantly to get the accuracies of the test set, it showed in Fig 3. Our model achieved the best results quickly without

1. https://pypi.python.org/pypi/jieba/
2. http://radimrehurek.com/gensim/

Chinese Sentence	相近 (similar)	赔率 (odds)	下 (under)	西甲 (Liga)	主队 (home team)	客胜 (away win)	概率 (prob.)	很小 (very small)	。 (.)
English Translation	Under the similar odds, away winning probability of the home team is very small.								

Fig. 2. Visualization of attention over words.

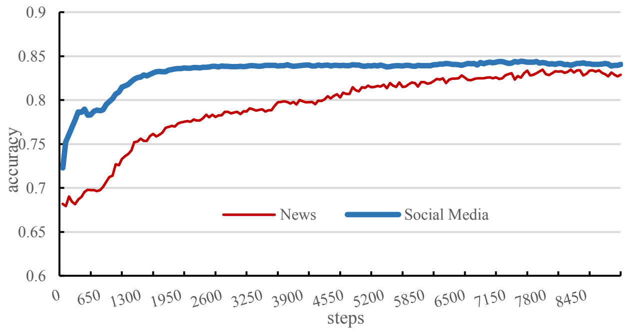

Fig. 3. Accuracy of different training steps.

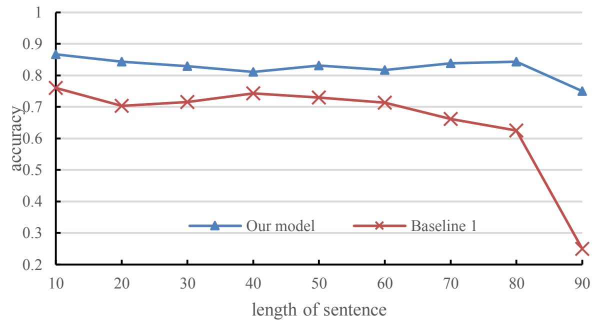

Fig. 4. Accuracy for different sentence lengths.

TABLE II. RESULTS ON CHINESE SOCIAL MEDIA DATASET.

Model	Precision	Recall	F1
Baseline1	0.6754	0.6502	0.6625
Baseline2	0.5271	0.5425	0.5347
CNN	0.7081	0.6286	0.6659
RNN	0.7127	0.6800	0.6959
RNN+ATT	0.7681	0.7186	0.7425
CNN+RNN	0.7662	0.7029	0.7331
ALUNI	**0.7784**	**0.7856**	**0.7819**

TABLE III. RESULTS ON CHINESE NEWS DATASET.

Model	Precision	Recall	F1
Baseline1	0.7024	0.7082	0.7053
Baseline2	0.6235	0.6620	0.6422
CNN	0.5928	0.6230	0.6075
RNN	0.6343	0.6808	0.6567
RNN+ATT	0.7090	0.7289	0.7188
CNN+RNN	0.7010	0.7157	0.7083
ALUNI	**0.7414**	**0.7377**	**0.7395**

obvious overfitting. The model was more stable on social media data set which was a bigger dataset.

Table II and Table III showed the experimental results of the comparison on Chinese news dataset and Chinese social media dataset, respectively.

From Table II and Table III, we found that ALUNI (CNN+RNN+ATT) outperformed both baselines, and yielded an F1-measure scores of 78.19% and 73.95% on social media dataset and news dataset, respectively, with 11% and 3% improvement over the Baseline1, respectively. Both of the baselines performed bad on the social media dataset, and about 4% of F1-measure score was decreased comparing to news dataset, while ALUNI increased more than 4%. It proved that ALUNI could perform well on both news data and social media data.

On the social media dataset, our model had a better performance than baselines, as there were 972 sentences in our social media dataset, which were uncertain but the 388 cue words provided by baseline 1 did not appear in these sentences. As for the baseline 2, we also used the cue words provided by baseline 1. Experiment result showed the cue words methods were not complete and not common. The reason why our model did better on the social media dataset than the Chinese news dataset was that larger amount of data was helpful to deep learning and deep learning was good at deal with the dataset annotated according to semantics.

What's more, the results also showed that the attention mechanisms were helpful, since both RNN+ATT and ALUNI outperformed their counterparts without attention mechanisms. We analyzed the different results of different models, we considered that RNNs could capture more global features, so it had a high recall by understand the general idea of the sentence. While CNNs and attention mechanisms could grasp the most accurate features for uncertainty identification to get high precision, the attention mechanism was more effective and the combine of CNNs and attention mechanism was the best.

To illustrate the effectiveness of our attention mechanisms, the sentence "Under the similar odds, the away winning probability of the home team is very small." taken from the social media dataset is visualized with attention weights in Figure 2, where darker color means higher weight indicating uncertainty. In this instance, there is no cue-phrase in the sentence, so this sentence can hardly be identified by both baseline methods. In ALUNI, however, the word "odds" with its implicit uncertain semantics can be captured by attention mechanisms, even if it is not a cue-phrase, and hence the sentence is determined as uncertain.

Finally, we examined the effect of sentence length on our model. We calculated the accuracy of different lengths sentences. The sentence length here is the number of words, not characters. We divided the sentences into nine groups at intervals of 10 words. The accuracy value points were drawn in the end of each group, such as the position at 10, 20 and etc. We compared the results of baseline 1 with the results of our model. As we could see from the Fig 4, the accuracy of our model didn't decrease with the increase of sentence length, but the accuracy only decreased a little in very long sentences. The accuracy of Baseline 1 dropped at the sentence length 40, it had a very poor performance in very long sentences. Although there are only a few long sentences in social media text, our model can cope with long sentence problems so that our model can be used trustingly.

V. CONCLUSIONS AND FUTURE WORKS

This paper proposes attention-based LSTM-CNNs for uncertainty identification on Chinese social media texts, named ALUNI. ALUNI uses attention-based LSTM networks to focus on the words regardless of cue-phrases or not that have decisive effect on the uncertain semantics of the sentence without resorting to extra knowledge or external NLP components. The convolutional neural networks of ALUNI capture the most important semantic information for uncertainty identification.

Experiments are conducted on both Chinese social media and news datasets, and 78.19% and 73.95% of F1-measure scores are achieved with about 11% and 3% improvement, respectively.

In the future, we will extend our work from two aspects: we will first expand the social media dataset and make further classification of the uncertain sentences into different uncertain types. We will also evaluate our proposed method in other languages.

ACKNOWLEDGMENT

This work is partially supported by the Key Lab of High Confidence Software Technologies(CUHK) MoE, and funded by the National Natural Science Foundation of China (61502115, 61602326, 61572043, U1636103, U1536207, 61672361), and the fundamental research fund for the central universities (3262015T70, 3262017T12, 3262017T18). We would also like to thank Nuo Chen, Le Chen, Ning Jiang, Lingchen Lian, Haotian Ren, and Jun Xiang for their work on corpus.

REFERENCES

[1] G.Szarvas, V.Vincze, R.Farkas, G.Móra, and I.Gurevych. "Cross-genre and cross-domain detection of se-mantic uncertainty". Computational Linguistics, 38(2), 335-367. 2012.

[2] X J. Li, W.Gao, and J.W.Shavlik. "Detecting Semantic Uncertainty by Learning Hedge Cues in Sentences Using an HMM. In Proceeding of SIGIR". 2014.

[3] R.Farkas, V.Vincze, G.Móra, J.Csirik, and G.Szarvas. "The conll-2010 shared task: learning to detect hedges and their scope in natural language text". In Proceedings of the 14 Conference on Com-putational Natural Language Learning-Shared Task, CoNLL '10: Shared Task, pages 1-12. As-sociation for Computational Linguistics. 2010.

[4] Conference on Computational Natural Lan-guage Learning-Shared Task, CoNLL '10: Shared Task, pages 126-131. Association for Computational Linguistic.

[5] Z Y. Wei, J W. Chen, W.Gao, B Y. Li, L J. Zhou. "An Empirical Study on Uncertainty Identification in Social Media Context". In Proceedings of the Association for Computa-tional Linguistics pages 58-62. 2013.

[6] D. Bahdanau, K. Cho, and Y. Bengio. "Neural machine translation by jointly learning to align and translate". arXiv preprint arXiv:1409.0473. 2014.

[7] Y. Kim. "Convolutional neural networks for sentence classification." arXiv preprint arXiv:1408.5882, 2014.

[8] R. Morante and C. Sporleder. "Mo-dality and negation: An introduction to the spe-cial issue". Computational Linguistics, 38(2):223-260. 2012.

[9] M. Light, X Y. Qiu, and P. Srinivasan. "The language of bioscience: Facts, specu-lations, and statements in between". In Proceed-ings of BioLink 2004 workshop on linking bio-logical literature, ontologies and databases: tools for users, pages 17-24. 2004.

[10] W.W. Chapman, D. Chu, and J.N. Dowling. "Context: An algorithm for iden-tifying contextual features from clinical text". In Proceedings of the ACL Workshop on BioNLP 2007, page 81-88. 2007.

[11] B.Medlock. "Exploring hedge identification in biomedical literature". Journal of biomedical informatics 41.4 (2008): 636-654. 2008.

[12] E. Fernandes, C. Crestana, and R. Mi-lidiu. "Hedge detection using the RelHunter approach". In Proceedings of the 14 Conference on Computational Natural Lan-guage Learning-Shared Task, CoNLL '10: Shared Task, pages 64-69. Association for Computational Linguistics. 2010.

[13] X X. Li, J P. Shen, X. Gao, and X. Wang. "Exploiting rich features for detect-ing hedges and their scope". In Proceedings of the 14 Conference on Computational Natural Language Learning-Shared Task, CoNLL '10: Shared Task, pages 78-83. Association for Computational Linguistics. 2010.

[14] Buzhou Tang, Xiaolong Wang, Xuan Wang, Bo Yuan, and Shixi Fan. 2010. A cascade method for detecting hedges and their scope in natural language text. In Proceedings of the 14 Confer-ence on Computational Natural Language Learning-Shared Task, CoNLL '10: Shared Task, pages 13-17. Association for Computa-tional Linguistics.

[15] S D. Zhang, H. Zhao, G D. Zhou, and B L. Lu. "Hedge detection and scope finding by sequence labeling with nor-malized feature selection." In Proceedings of the 14 Conference on Computational Natural Lan-guage Learning-Shared Task, CoNLL '10: Shared Task, pages 92-99. Association for Computational Linguistics. 2010.

[16] E. Velldal. "Detecting uncertainty in bio-medical literature: A simple disambiguation approach using sparse random indexing". In Proceedings of SMBM 2010, pages 75-83, Cambridge, UK. 2010.

[17] F. Ji, X P. Qiu and X J. Huang. "A Research on Chinese Uncertainty Sentence Recognition". In Proceedings of the 16th China Conference on Information Retrieval, pages 595-601. 2010.

[18] Vincze,Veronika. "Uncertainty detection in natural language texts(Doctoral dissertation, szte)". 2015.

[19] Vincze, Veronika. "Detecting uncertainty cues in Hungarian social media texts". In Pro-ceedings of the Workshop on Extra-Propositional Aspects of Meaning in Computa-tional Linguistics. 2016.

[20] Y. Zhang and B.C. Wallace. A "sensitivity analysis of (and practitioners' guide to) convolutional neural networks for sentence classification". arXiv preprint arXiv:1510.03820. 2015.

[21] Z C. Yang, D Y. Yang, C. Dyer, X D. He, A. Smola, and E. Hovy. "Hierarchical Attention Networks for Document Classification". In Proceeding of Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies pages: 1480-1489. 2016.

[22] S. Hochreiter and J. Schmidhuber. "Flat minima". Neural Computation, 9(1), 1-42. 1997.

[23] M. Sundermeyer, R. Schlüter, and H. Ney. "LSTM Neural Networks for Language Modeling". In Interspeech pages 194-197. 2012.

[24] K S. Yao, T. Cohn, K. Vylomova, K. Duh, and C. Dyer. "Depth-Gated Recurrent Neural Networks". https://arxiv.org/abs/1508.03790v1. 2015.

[25] A. Graves. "Generating sequences with re-current neural networks". Computer Science, 2013: 2048-2057. 2013.

[26] R. Collobert, J. Weston, L. Bottou, M. Karlen, K. Kavukcuoglu et al. "Natural language processing (almost) from scratch". Journal of Machine Learning Re-search, 12(1), 2493-2537. 2011.

[27] T. Mikolov, K. Chen, G. Corrado, and J. Dean. "Efficient estimation of word representa-tions in vector space". arXiv preprint arXiv:1301.3781. 2013.

[28] Z. He, X. Li, D.Tao, X. You, and Y. Tang, "Connected Component Model for Multi-Object Tracking, IEEE Trans. on Image Processing", 2016, vol.25, no.8, pp.3698-3711.