Relation Aware Attention Model for Uncertainty Detection in Text

Manjira Sinha
IIT Kharagpur
manjiras@gmail.com

Nilesh Agarwal
IIIT Guwahati
agarwalnilesh@gmail.com

Tirthankar Dasgupta
TCS Research
iamtirthankar@gmail.com

ABSTRACT

Uncertainty in text is an important linguistic phenomenon that is relevant in many areas of natural language processing. In this paper, we present a neural approach towards detecting uncertainty cues in texts. We have explored a series of neural network architectures and evaluated the models with respect to three different data sources belonging to domains such as bio-medical texts, privacy policies, and product reviews. Our preliminary analysis showed that the relation aware attention models outperform the existing baseline systems across all the domains. We have also observed for domain specific texts incorporating character level embeddings significantly improves the performance.

KEYWORDS

Uncertainty detection, vagueness in text, neural networks

ACM Reference Format:
Manjira Sinha, Nilesh Agarwal, and Tirthankar Dasgupta. 2020. Relation Aware Attention Model for Uncertainty Detection in Text. In *Proceedings of the ACM/IEEE Joint Conference on Digital Libraries in 2020 (JCDL '20), August 1–5, 2020, Virtual Event, China.* ACM, New York, NY, USA, 4 pages. https://doi.org/10.1145/3383583.3398613

1 INTRODUCTION

For the past few decades *uncertainty* or *vagueness* has been a subject of vigorous debate in the philosophy of logic and language. Terms like 'weak', 'redish', and 'cold' are considered as vague due to their lack of well-defined extensions (someone may be neither weak nor not strong or there is no sharp distinction between strong people and the rest) **??**. This fuzzyness in semantics poses a fundamental challenge to classical symbolic logic that assumes propositions to be either true or false.

Uncertainty can be at both semantic level and discourse level [15] [18][1] [14]. Determining semantic uncertainty involves extracting lexical cues (such as epistemic model verbs and adverbs) in the form of hedges [6] [20], weasels [7] [4], and peacocks [1]. On the other hand, discourse-level uncertainty can be expressed by both lexical as well as syntactic cues (such as passive constructions). For example, the below sentence, taken from the Bioscope corpus, exhibits uncertain information:

< sentenceid = "S2.9" >
Interestingly, the MnlI – AluI fragment < xcopeid = "X2.9.1" >< cuetype = "speculation" ref = "X2.9.1" > could < /cue > suppress the basal – level activity of the conalbumin promoter in both Jurkat and HeLa cells </xcope>.</sentence>

There have been many attempts to detect uncertainty from texts. The existing works spans across different domain and genre of texts [8] [18] [15] [18] [14] [3] [4] [11] [7] [22] [12][23]. Applications of such works are many. Most of these works are usually based on supervised machine learning methods applied to a set of carefully designed complex linguistic features. Knowledge of such complex features has been shown to achieve well. However, due to the fact that multiple factors influence textual uncertainties, it is difficult to enumerate all these factors exhaustively.

In this paper, we have proposed a Relation aware Self-attention based Cascaded CNN-Tree-LSTM (Re-CNN-TLSTM) model for the automatic detection of uncertainty from textual documents belonging to different domains and genres. We have considered the task as a binary classification problem, where each sentence is classified into two classes namely, *uncertain* or *Not an Uncertain* sentence. We have also evaluated the proposed model with respect to a series of deep neural network architectures to establish its scalability. Further, we have also used the BERT-base model, fined-tuned over the given training corpus, for the same classification task. We evaluate our model with respect to three domains namely, Bio-medical texts, Review of movies, books and consumer products, and privacy policies. Our preliminary investigation shows that the proposed Re-CNN-TLSTM model surpasses the existing state-of-the-art BERT based models.

2 LINGUISTIC CONSTRUCTS IN UNCERTAINTY

In linguistics, uncertainty is broadly classified into two groups namely, semantic and discourse-level uncertainty. Semantic level uncertainty is usually associated with modal verbs (Palmer,1986), but the terms factuality [15], veridicality [4], evidentiality [2] and commitment [5] are also used. However, in this paper, we will specifically analyze the following discourse-level uncertainty phenomena.

Weasels: are kind of uncertainty markers that deal with the source and reliability of information conveyed [13, 23]. Wikipedia defines Weasels as events with no obvious source or is specified only vaguely or too generally, hence, it cannot be exactly determined who the holder of the opinion is. For example, the sentence *"**Some** people claim that not enough of the waste from homes is recycled."* is weasel since the source of the information is represented

[1] http : //en.wikipedia.org/wiki/Wikipedia : Manual_of_Style/Words_to_watch

JCDL '20, August 1–5, 2020, Virtual Event, China

ACM ISBN 978-1-4503-7585-6/20/06…$15.00
https://doi.org/10.1145/3383583.3398613

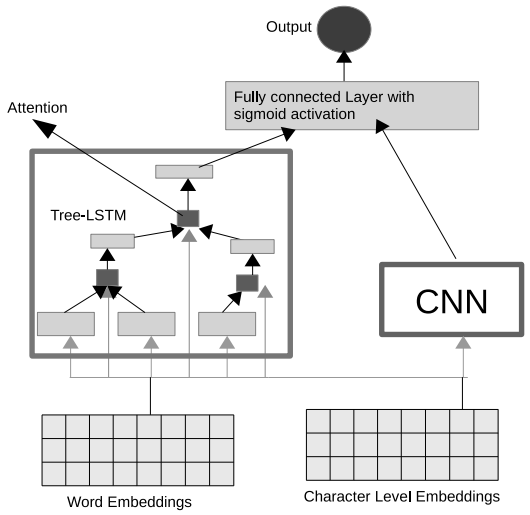

Figure 1: Overview of the neural network architecture (A) and a partial view of the relation aware self attention based Tree-LSTM (B).

by a pronoun *some*. Thus, the provider of the information is not known making the statement uncertain whether this is a reliable piece of information. Similarly, passive constructs like, "*As has been suggested **[by whom?]**, I must take a leave.*" which do not identifies the agent, also belongs to the weasel group.

Hedges: are the language constructs that make meaning of a statement fuzzy [6, 20]. It includes speculations, approximators, and passive voices. Moreover, circumscribers (approximately, probably), intensifiers (extremely, many), and deintensifiers (a little bit, low, small) also belong to this category.

Peacocks: are polarity induced expressions where words express unprovable qualifications or exaggerations [7]. Examples of Peacock terms include, brilliant, awesome, excellent, poor, and astonishing.

3 THE RELATION AWARE SELF-ATTENTION BASED CNN-TREE-LSTM MODEL

The Input Embeddings: In approach-1 the Re-CNN-Tree-LSTM model uses static embeddings such as FastText, Word2Vec and ELMo. Words occurring more than five times in the corpus are included and the rest denoted as UNK. To mitigate the loss due to the unknown words, we have also used character level embeddings. For ELMo, the context-based embedding is obtained through a two-layer Bi-LSTM language model (bi-LM) along with a stacked layer of a character-based Convolutional Neural Network (char-CNN) followed by a low-dimensional projection and a fully-connected layer. Thus, the contextual word embedding is formed with a trainable aggregation of highly-connected bi-LM. The new embeddings are then fed into the proposed neural network framework. The overall architecture of the proposed model is depicted in Figure 1.

In approach-2, we use the BERT-base architecture having 12 layers of transformer blocks, 768 hidden units, and 12 self-attention heads. The model is fine-tuned on training dataset. The fine-tuning approach adjusts the entire language model and is is integrated into the downstream classification task. We used BERT models for both general domain and biomedical domain (BIO-BERT).

Tree-Structured LSTM: In a standard Bi-LSTM model, information propagates sequentially. This often ignores the complex dependency structures among the linguistic entities. Thus, following the work of [19] we have used the *Dependency Tree-LSTM*. Such Tree-LSTM models allows richer network topologies. Here, the gating vectors and memory cell updates are dependent on the states of multiple child units.

The CNN Layer: Along with the Tree-LSTM units we also fed the input to the convolution networks. Each of the convolution network assumes a pre-defined sequence of n-gram words as input and performs a series of linear transformation [10]. The width of the convolution is k over the sentence. Accordingly, the convolution layer applies a linear transformation to all K windows in the given sequence of vectors. In order to ensure a uniform dimension over all sequences of input and output vectors, we perform a zero padding.

Relation aware Self-attention: Following [16] [17] we extend the self-attention unit attached with the hidden layers to consider the pairwise relationships between input elements. The pair between the input elements are formed according to the dependency based relations. Thus, we model the input as a labeled, directed, fully-connected graph. The edge between input elements x_i and x_j is represented by vectors $a_{ij}^V, a_{ij}^K \in R^{d_a}$. These representations can be shared across attention heads. Thus, we can modify the traditional self attention representation to propagate edge information to the sub-layer output as: $Z_i = \sum_{j=1}^{N} \alpha_{ij}(x_j.W^V + a_{ij}^V)$.

This modification is important for tasks where information about the edge types selected by a given attention head is useful to the classification engine. We further modify the edge vectors to consider edges when determining compatibility as:

$$e_{ij} = \frac{x_i W^Q (x_j W^K + a_{ij}^K)^T}{\sqrt{d_z}} \quad (1)$$

4 EVALUATION AND RESULTS

4.1 The Dataset

Dataset-1: Website Privacy Policies: The corpus was formed by including 100 website privacy policies from the collection gathered by [12]. The documents used are quite extensive and they contain 2.3K words on a average. Vagueness scores assigned by human annotators were averaged and further discretized into four buckets: [1,2), [2,3), [3,4), [4,5], respectively corresponding to "clear," "somewhat clear," "vague," and "extremely vague" categories. The sentences in the four buckets respectively consist of 26.9%, 50.8%, 20.5%, and 1.8% of the total annotated sentences.

Dataset-2: SFU Review Corpus: We choose The Simon Fraser University Review corpus [11] consisting of 400 documents (50 of each type) of movies, books, cars, hotels and consumer product reviews from the website Epinions.com. Sentences: 17263 (13351 non-speculative and 3912 speculative)

Dataset-3: BioScope Corpus: The corpus that we have used, includes text extracts from 3 distinct sources and 2 distinct types so that we can ensure that it captures the heterogeneity of the

language that is used in the biomedical domain. Biological full papers and Biological abstracts (texts from Genia) have been added to the corpus. [9, 21]. Documents: 1282, Sentences:- 14496 (11511 non-speculative and 2685 speculative)

The dataset-2 and 3 are collected from the The CoNLL-2010 shared task on learning to detect hedges and their scope in natural language text [6].

4.2 Fine-tuning the Models

We apply the following hyperparameter settings for fastText: window size of 15, minimum word count of 5, 15 iterations, and embedding size of 300. For ELMo, we follow the same setting for pre-training as reported in Peters et al., (2018). For UNK, a char-CNN embedding layer is applied with 16-dimension character embeddings, filter widths of [1, 2, 3, 4, 5] with respective [32, 64, 128, 256, 512] number of filters. After that, a two-layer Bi-LSTM with 4,096 hidden units in each layer is added. The output of the final bi-LSTM language model is projected to 512 dimensions with a highway connection. The pre-training step is performed for 15 epochs.

Fine-tuning BERT Depending on the type of the dataset we have used both the pre-trained Bio-BERT and General-BERT models. We use Xavier initialization rather than initializing the Bi-LSTM output weights (Glo rot and Bengio, 2010). Without this the fine-tuning was failed to converge. The early stopping of fine-tuning is set to 800 steps without improvement to prevent over-fitting. Finally, post-processing steps are conducted to align the BERT output with the concept gold standard, including handling truncated sentences and word-pieced tokenization.

4.3 Experiments

Based on the given datasets, we have conducted three different experiments.

In **Experiment-I:** We take each of the individual datasets and we divided them into three groups 70%, 20% and 10% for training, validation, and testing respectively. We have performed several experiments to identify the best model architecture for our task.

It is worth mentioning here that in dataset-1 the training set is prepared in such a way that there are four output classes unlike the two classes as presented in dataset-2 and 3. Therefore, in such a case we have specifically modified our neural network architecture to output four different classes instead of two binary classes. Due to the increase number of classes we have observed significant difference in results for this dataset as compared to the other two dataset.

In **Experiment-II:** We combine all the datasets together and formed a combined annotated corpus of 8K sentences. We then divided the entire corpus into 70%, 20% and 10% for training, validation, and testing respectively. The entire training set is then used to evaluate the proposed models.

In **Experiment-III:** We selectively choose some of the individual dataset, train our models over the given dataset and finally tested them over other datasets. For example, we train our models on the SFU dataset and tested the models using the *BioScope* and *Privacy Policy* dataset.

Additionally, we have explored a number of other deep neural network-based models as a baseline system. Some of these models are the bi-directional LSTM model (BILSTM), convolution network (CNN), and CNN with BILSTM. Each of the above neural network model were individually trained and tested with FastText vectors, Word2Vec embedding and ELMo embeddings. Results of each of the experiment are reported in Table 1.

Table 1: Results of experiments demonstrating F1 Scores for each model across different dataset. '*' represent models developed using both word and character level embeddings

	Dataset-1			Dataset-2			Dataset-3		
	FT	W2V	EL	FT	W2V	EL	FT	W2V	EL
CNN	0.40	0.32	0.40	0.61	0.61	0.63	0.67	0.67	0.69
BiLSTM	0.41	0.37	0.43	0.56	0.59	0.58	0.53	0.54	0.58
CNN-LSTM	0.41	0.36	0.41	0.65	0.64	0.66	0.61	0.63	0.65
Re-CNN-LSTM	0.50	0.51	0.53	0.75	0.71	0.77	0.69	0.68	0.77
Re-CNN-LSTM*	0.51	0.54	0.55	0.77	0.74	0.75	0.74	0.75	0.78
Re-CNN-TLSTM	0.53	0.55	0.58	0.85	0.82	0.90	0.79	0.78	0.83
Re-CNN-TLSTM*	0.54	0.55	**0.59**	0.87	0.80	**0.92**	0.78	0.77	0.84

BERT	DS-1	DS-2	DS-3
BERT-base	**0.57**	0.85	**0.85**[2]

Table 2: Results of Experiment-II demonstrating F1 Scores for each model across the combined dataset.

	FastText	Word2Vec	ELMo
CNN	0.64	0.66	0.69
BiLSTM	0.61	0.67	0.69
CNN-BiLSTM	0.65	0.69	0.71
Re-CNN-LSTM	0.71	0.73	0.75
Re-CNN-LSTM*	0.72	0.75	0.79
Re-CNN-TLSTM	0.74	0.76	0.81
Re-CNN-TLSTM*	0.77	0.78	0.84

Table 3: Results of Experiment-III demonstrating F1 Scores for the CNN-BiLSTM-att model when trained over a given dataset D_i and tested over other datasets D_j such that $i \neq j$.

	Dataset-1	Dataset-2	Dataset-3
Dataset-1	X	0.45	0.51
Dataset-2	0.49	X	0.73
Dataset-3	0.44	0.75	X

4.4 Results

We first tried to evaluate the performance of the individual models. We found that in most of the cases the performance of the proposed Re-CNN-TLSTM* model along with the word and character level ELMo embedding is higher than the individual baseline models across all the datasets.

During the analysis of the individual datasets we have observed that for Dataset-2, we have achieved an F1 score of 0.89 using the Re-CNN-TLSTM* model with ELMo embedding. This is the highest accuracy that we have achieved among all other models. We also observe the difference in performance with respect to the attention mechanism used. The relation aware self attention mechanism is performing significantly well when using the same neural network architecture. Thus, the pair wise dependency relationship between words shows to plays an important role.

Table 4: Distribution of uncertainty cues across dataset(Ds)

Ds-1	May, other, some, most, certain, third parties, time to time, generally, others, might, services various
Ds-2	May, think, can, could, would, should, seems, perhaps, probably, believe, either, thought, whether, maybe, likely, suggest
Ds-3	Might, likely, potential, appeared, seems, approximately, suggest, indicated that, unclear, possible, significant

For *Dataset-1*, we found both the BERT based classification (F1 = 0.58) and Re-CNN-TLSTM* (F1 = 0.57) performs comparatively well. We also observe a improvement in performance with respect to models employing the plain LSTM and Tree-LSTM (see Table 1. This may be accounted for due to the fact that different website privacy policies uses different type of language constructs and are having a restrictive vocabulary of terms. Thus, there might be significant variation in dependency structures that are captured by the model. It is worth mentioning here that our proposed model surpasses the existing performance of the AC-GAN model (F1 = 0.52) reported in the literature [12].

In case of *Dataset-3*, F1 score of 0.85 is achieved in BERT-base model as compared to the 0.84 F-score in ReMC-CNN-TLSTM model. The reason behind better performance of BERT over ReMC-CNN-TLSTM is the fact that Dataset-3 belongs to biomedical domain. Thus, we have used the pre-trained Bio-BERT model and finedtuned over the Bioscope dataset.

We have also explored the performance of the different word embedding strategies used in the experiments. We observe that ELMO embeddings are very effective in capturing solely contextual information and thus surpasses the performance of the other embedding models.

In **Experiment-II:** Table 2 report the results obtained after combining all the training datasets together and testing the individual models. Similar to the observations reported for Experiment-I we can see that the performance of the Re-CNN-TLSTM* model far surpasses the performance of the other models.

For **Experiment-III:** Here we have trained the models on one single dataset and tested over other datasets. The results are depicted in Table 4. Due to page limitations we report only the results of the ReMC-CNN-TLSTM with the ELMo embedding model since it has shown to be reporting the best results among all the other models.

Apart from the above results, we have noted that depending upon the domain and genre of the text uncertainty cues do vary among each other. Table 4 reports some of the frequently used uncertainty cues across different domain.

5 CONCLUSION

Identifying uncertainty in texts is an important yet non-trivial problem. A lot of computational analysis based on different linguistic features have been performed in the past. In this paper, we aim to study the effectiveness of deep neural network for uncertainty detection. Accordingly, we have proposed a relation aware self-attention based multi-channel CNN-Tree-LSTM based model and compared its performance with other standard deep neural network based models including the BERT architecture. We have evaluated the models with respect to three different data sources. We showed that the proposed network architecture is more effective in obtaining an higher recall with fewer false positives compared to simple n-gram shifting context window features. However, a detailed analysis of the results is yet to be done which remains a future scope of this work.

REFERENCES

[1] Cem Akkaya, Janyce Wiebe, and Rada Mihalcea. 2009. Subjectivity word sense disambiguation. In *Proceedings of the 2009 conference on empirical methods in natural language processing*. 190–199.

[2] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci. 2002. Wireless Sensor Networks: A Survey. *Comm. ACM* 38, 4 (2002), 393–422.

[3] Ferdinand De Haan. 1999. Evidentiality and epistemic modality: Setting boundaries. *Southwest journal of linguistics* 18, 1 (1999), 83–101.

[4] Marie-Catherine De Marneffe, Christopher D Manning, and Christopher Potts. 2012. Did it happen? The pragmatic complexity of veridicality assessment. *Computational linguistics* 38, 2 (2012), 301–333.

[5] Mona T Diab, Lori Levin, Teruko Mitamura, Owen Rambow, Vinodkumar Prabhakaran, and Weiwei Guo. 2009. Committed belief annotation and tagging. In *Proceedings of the Third Linguistic Annotation Workshop*. Association for Computational Linguistics, 68–73.

[6] Richárd Farkas, Veronika Vincze, György Móra, János Csirik, and György Szarvas. 2010. The CoNLL-2010 shared task: learning to detect hedges and their scope in natural language text. In *Proceedings of the Fourteenth Conference on Computational Natural Language Learning—Shared Task*. Association for Computational Linguistics, 1–12.

[7] Viola Ganter and Michael Strube. 2009. Finding hedges by chasing weasels: Hedge detection using Wikipedia tags and shallow linguistic features. In *Proceedings of the ACL-IJCNLP 2009 Conference Short Papers*. 173–176.

[8] Pierre-Antoine Jean, Sébastien Harispe, Sylvie Ranwez, Patrice Bellot, and Jacky Montmain. 2016. Uncertainty detection in natural language: A probabilistic model. In *Proceedings of the 6th International Conference on Web Intelligence, Mining and Semantics*. ACM, 10.

[9] J-D Kim, Tomoko Ohta, Yuka Tateisi, and Jun'ichi Tsujii. 2003. GENIA corpus—a semantically annotated corpus for bio-textmining. *Bioinformatics* 19, suppl_1 (2003), i180–i182.

[10] Yoon Kim. 2014. Convolutional neural networks for sentence classification. *arXiv preprint arXiv:1408.5882* (2014).

[11] Natalia Konstantinova, Sheila CM De Sousa, Noa P Cruz Díaz, Manuel J Mana López, Maite Taboada, and Ruslan Mitkov. 2012. A review corpus annotated for negation, speculation and their scope.. In *Lrec*. 3190–3195.

[12] Logan Lebanoff and Fei Liu. 2018. Automatic detection of vague words and sentences in privacy policies. *arXiv preprint arXiv:1808.06219* (2018).

[13] Raheel Nawaz, Paul Thompson, and Sophia Ananiadou. 2010. Evaluating a meta-knowledge annotation scheme for bio-events. In *Proceedings of the Workshop on Negation and Speculation in Natural Language Processing*. Association for Computational Linguistics, 69–77.

[14] Frank Robert Palmer. 2001. *Mood and modality*. Cambridge University Press.

[15] Roser Saurí and James Pustejovsky. 2012. Are you sure that this happened? assessing the factuality degree of events in text. *Computational Linguistics* 38, 2 (2012), 261–299.

[16] Peter Shaw, Jakob Uszkoreit, and Ashish Vaswani. 2018. Self-attention with relative position representations. *arXiv preprint arXiv:1803.02155* (2018).

[17] Yusuke Shido, Yasuaki Kobayashi, Akihiro Yamamoto, Atsushi Miyamoto, and Tadayuki Matsumura. 2019. Automatic Source Code Summarization with Extended Tree-LSTM. *arXiv preprint arXiv:1906.08094* (2019).

[18] György Szarvas, Veronika Vincze, Richárd Farkas, György Móra, and Iryna Gurevych. 2012. Cross-genre and cross-domain detection of semantic uncertainty. *Computational Linguistics* 38, 2 (2012), 335–367.

[19] Kai Sheng Tai, Richard Socher, and Christopher D Manning. 2015. Improved semantic representations from tree-structured long short-term memory networks. *arXiv preprint arXiv:1503.00075* (2015).

[20] Veronika Vincze. 2013. Weasels, hedges and peacocks: Discourse-level uncertainty in wikipedia articles. In *Proceedings of the Sixth International Joint Conference on Natural Language Processing*. 383–391.

[21] Veronika Vincze, György Szarvas, Richárd Farkas, György Móra, and János Csirik. 2008. The BioScope corpus: biomedical texts annotated for uncertainty, negation and their scopes. *BMC bioinformatics* 9, 11 (2008), S9.

[22] Zhongyu Wei, Junwen Chen, Wei Gao, Binyang Li, Lanjun Zhou, Yulan He, and Kam-Fai Wong. 2013. An Empirical Study on Uncertainty Identification in Social Media Context.. In *ACL (2)*. World Scientific, 58–62.

[23] Janyce Wiebe, Theresa Wilson, and Claire Cardie. 2005. Annotating expressions of opinions and emotions in language. *Language resources and evaluation* 39, 2-3 (2005), 165–210.