Directed Literature Reading

on the topic of Sentiment Analysis

Weixiang Zheng
Department of Computer Science
Nanyang Technological University
Singapore, Singapore
wzheng014@e.ntu.edu.sg

ABSTRACT

In this directed reading assignment, I am going to present to you the literature review of three selected papers on the topic of sentiment analysis: 1) Sentiment Analysis of Twitter Data [1] by Apoorv Agarwal, Boyi Xie, Ilia Vovsha, Owen Rambow and Rebecca Passonneau. 2) Sentiment Analysis by Capsules [2] by Yequan Wang, Aixin Sun, Jialong Han, Ying Liu and Xiaoyan Zhu. 3) Aspect-level Sentiment Analysis using AS-Capsules [3] by Yequan Wang, Aixin Sun, Minlie Huang, Xiaoyan Zhu. The first paper is a relatively old paper published in 2011 [4], the other two papers are relatively new papers that Prof Sun has participated. After that, I am going to talk about some of my thoughts on the topic of Sentiment Analysis.

1 Sentiment Analysis of Twitter Data

This paper was published at the early age of microblogging services. The purpose of this paper was to build a technology to detect and summarize an overall sentiment for every tweet. In this paper, authors build models for two types of classification tasks: 1) a binary classification that classifies sentiment into positive or negative class, 2) a 3-way classification that classifies sentiment into positive, negative and neutral classes. For each of the classifications, they experiment with three types of models: 1) a baseline unigram model that was shown to work well for sentiment analysis task, 2) a feature-based model that proposes new features in the past literature, 3) a new tree kernel-based model that is newly proposed in this paper. The result is that newly proposed tree kernel-based model outperforms the other two in the following:1) number of features: tree kernel-based model uses 100 features to achieve the baseline model with 10000 features performance, 2) the tree kernel-based model has better accuracy when keep other the same, 3) combined models also result in better performance. Some other discoveries include: Twitter-specific features such as hashtags add very limited value to the classifier, on the other hand features that combine prior polarity of words with their POS tags are very important, i.e. if you can understand the sentiment of a newswire, you are very likely to comprehend the sentiment of Twitter as well.

The experiment data is 11875 tweets from a commercial source. They collect data by archiving real-time stream without any restriction and manually annotating sentiment based on human knowledge. Among these data, some are in foreign language, they use Google Translate and eliminate those not translated well. The total data left are 8753 tweets. They further stratified the sample to get 1709 tweets for each of positive, negative and neutral classes.

The preprocessing of the data includes 1) dealing with emoticon, 2) dealing with URLs, 3) dealing with different negation, e.g. not, never, cannot, n't and so on. They addressed 1) by replacing the emoticon directly by the sentiment labels such as extremely-positive, positive, neutral, negative and extremely negative. Then they replace the URLs by tag || U ||, targets such as "@John" by tag || T || to solve 2). Finally, they unify the negation by replacing terms with negation meaning to NOT.

Next, they introduced a new tree kernel-based model to represent tweet. From the introduction, we can see many advantages, for example, it largely keeps the syntactic structure and semantic meaning of the original tweet, but at the same time, also unifies some of the difficult terms such as emoticon, targets in tweet, and negation terms. It adds pos tagging to every leaf of the tree and also keeps the order of the sentence. It minimizes the information loss during the preprocessing and maximize the information gain during the pos tagging, just by looking at the design of this Tree Kernel design, we should be able to tell the result should be promising.

The result indeed confirms our hypothesis, for the unigram model doing binary classification tasks, tree kernels outperform the unigram and the senti-feature by 2.58% and 2.66% respectively. Although one important point was Kernel+Senti-feature mixed model not as good as Unigram+Senti-feature mixed model. In the paper, author did not mention the reason, I personally suspect this is because the unigram only records the probability of a term showing up but tree kernel records a lot of other information such as POS tagging and order of the terms. There might be some conflicting information between tree kernel model and unigram model, and that conflict might drag the performance down a little bit. For the 3-way classification task, the balanced dataset is 1709 instances for each, and therefore the chance of baseline is 33.33%. For this task, tree kernel-based model outperform the unigram and Senti-feature model by 4.02% and 4.29% respectively. And for the mix model comparison, in this case, Kernel+Senti-featue still outperform the Unigram+Senti-feature model, but not by a large percentage, I think this is because both of them have reached the high performance and if anyone wants to be better than the other one in a significant way, parameters have to be carefully tuned and also senti-features have to be modified to adapt the kernel or unigram model.

The contribution of this paper is: 1) introduced POS-specific prior polarity feature 2) introduced the use of tree kernel to obviate the need for tedious feature engineering. The conclusion of this paper is the sentiment analysis of Twitter data is not that different from sentiment analysis for other genres such as newswire.

After reading this literature, I also would like to talk about the application of sentiment analysis in real life. Some real-life events such as presidential election are very much related to sentiment of population, if we can crawl enough information and use precise algorithm, we might be able to predict the result.[5] Natural language, however, are quite complicated, in a sense that, some sentences might have completely different meanings in different context, and also some features like sarcasm might not be easy to detect. Recently, however, the rise of deep learning makes computer smarter and more similar in term of capabilities, and that's why the following two papers are all related to deep learning.

2 Sentiment Analysis by Capsules

This paper was published in 2018. The purpose of this paper was to address some defects that neural network models have in the task of sentiment analysis, including but not limited to: 1) the quality of sentiment analysis heavily depend on the quality of instance representation, i.e., vector. 2) require prior linguistic knowledge, 3) domain specific dataset. The use of capsule solves the above problems. The result seems promising, it ranked top 3 in year 2018, still ranked top 4 recently [6]. It is also the first paper trying to perform sentiment analysis by capsules. The paper starts by introducing capsule. A capsule is a group of neurons which has rich significance. Each capsule contains an attribute, a state and three modules i.e., representation module, probability module and reconstruction module. The attribute of a capsule reflects the dedicated sentiment category i.e., Positive capsule and Negative capsule are built for a problem with two sentiment categories. The state of a capsule, i.e., active or inactive is determined by the probability modules of all capsules in the model. Therefore, a capsule state is active if the output of its probability module is the largest among all capsules. Representation module uses the attention mechanism to build capsule representation. Probability module uses the capsule representation to predict the capsule's state probability. Reconstruction module is used to rebuild the representation of the input instance, which is computed through RNN. The main contributions of this paper are: 1) RNN-Capsule is the first attempt to use a capsule to solve the problem of sentiment analysis, it marks the foundation for the third paper of this directed reading, 2) RNN-Capsule does not require any linguistic knowledge to achieve state-of-the-art performance. The model is able to attend opinion words that reflect domain knowledge of the dataset. 3) conduct experiments on benchmark datasets, show that capsule model is competitive and robust.

In the RNN-based capsule model, the number of capsule N is the same as the number of sentiment categories to be modeled. All capsules take the same instance representation as their input, which is computed by an RNN network. Given an instance, RNN encodes the instance and outputs the hidden vectors.

They use two popular datasets, Movie Review [7] and Stanford Sentiment Treebank [8], together with one proprietary patient opinion dataset Hospital Feedback for performance experiment. Because two benchmark datasets are widely used, we could directly compare the result with other models such as Recursive Auto Encoder (RAE) [9], Recursive Tensor Neural Network (RNTN)[10] and etc. On the Movie Review dataset, the capsule model achieves the best accuracy of 83.8, outperforming all other models and level with LR-Bi-LSTM and NCSL. However, LR-Bi-LSTM and NCSL require linguistic knowledge like sentiment lexicon and intensity regularizer which requires a lot of human effort. The capsule model does not use any linguistic knowledge and still performs the best. As for the SST dataset, the capsule model is the second-best model after CNN-Tensor. However, one thing worth-noting is CNN-Tensor requires much more computational power because of the tensor product operation. And also, it outperforms the LR-Bi-LSTM model or similar models that requires dedicated linguistic knowledge.

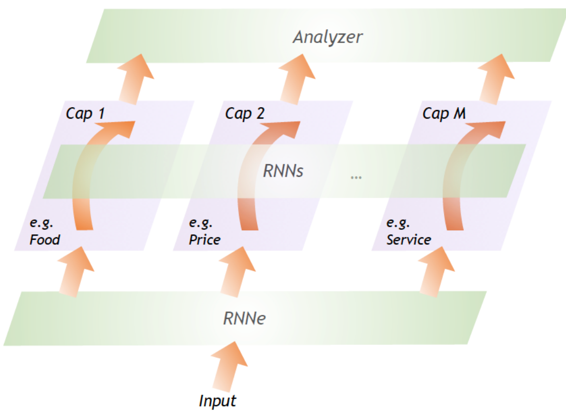
As for the evaluation on Hospital Feedback, this is because although the performance on two benchmarks looks promising, it is better to run an evaluation on a relatively new dataset. The comparison models are all from traditional machine learning models, including but not limited to Naïve Bayes, Support Vector Machines (SVM) using unigram and bigram representations, and also SVMs with dense vector representations obtained through Word2vec and Dco2vec and KSTM based baselines. The best among traditional machine learning models is the Linear SVM with unigram and bigram model, with an accuracy of 88.9. The capsule model, however, is the top performer among all models, improves the performance level to 91.6.

The authors conclude RNN-Capsule model as a sample capsule structure that uses each capsule to focus on one sentiment category. Each of the capsules outputs its active probability and the reconstruction representation. And the whole learning process is to maximize the active probability and the distance between reconstruction representations with the instance representations, and also to minimize its reconstruction representation and the other capsules' active probability.

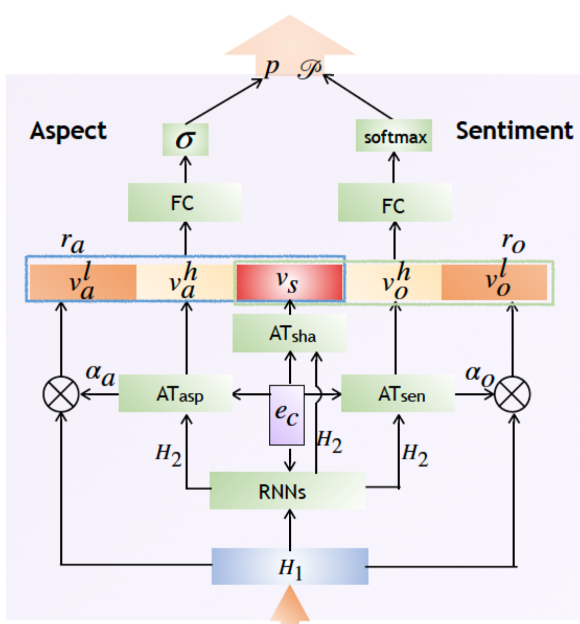
Personally, I think this is a very good paper, in a sense that it shows the state-of-art sentiment classification accuracy is actually achievable without any carefully designed instance representation or prior linguistic knowledge. It also proposed a method that was never used before. We can imagine the work built on top of it would make it even better, which comes to the last paper

3 Aspect-level Sentiment Analysis using AS-Capsules

This paper was published in 2019 on World Wide Web Conference. The background of this paper was based on the fact that most existing solutions to aspect-level sentiment analysis use two-stage approach: the first task is to detect the specific aspect category of a document, the second task is to do a categorization on the polarity of opinion expressions. Two separate tasks are dealt accordingly. There are two problems with the existing solutions: 1) error accumulations: error made in the first subtask will cause the final result wrong even if the approach to deal with the second subtask is correct. 2) the aspect detection task and the aspect-level sentiment classification task are highly correlated. The contribution of this paper is that authors proposed a new model: aspect-level sentiment capsules (AS-Capsules) model that could utilize the correlation between aspect and its corresponding sentiments. It enables us to perform the two tasks jointly: aspect detection and aspect-level sentiment classification together.



The architecture of the proposed model is depicted as above. The number of capsules M is the same as predefined aspect categories. And as in RNN-Capsule [11], we use an encoder RNN named RNNe to encode the input text. What this encoder do is that for a piece of text W, RNNe encode the given text and outputs a matrix H1 which is the hidden representation of the given instance. Formally: H1 = RNNe(W). All the capsules share the same RNN named RNNs. Also, they take the same hidden matrix H1 as their input. However, because different capsule has different aspect, they will value different parts of the input H1. Notably, RNNs allows capsules to communicate with each other, which is capable of preventing capsules from attending conflict parts.



Each capsule has the output of a probability representing how likely this aspect is and a sentiment distribution module. The structure of every capsule is as follows: A capsule consists of an attribute, a state, a capsule embedding and four modules. The attribute of a capsule reflects its dedicated aspect category, which is preassigned. The state of a capsule is determined by aspect probability. The capsule's state is active if the current aspect appears in input text. Then it maximizes the aspect probability of the current active capsule. Capsule embedding ec is a vector representation of the current capsule learned during training as in [12]. Aspect representation module learns the aspect representation ra including three parts [val, vah, vs]. Given the hidden representation H1 as input, we are to compute the high-level representation H2 through RNNs.

The used dataset for testing is SemEval 2014 Task 4 dataset [13] which mainly contains all sorts of reviews from customers who bought laptops or had a meal in restaurants, but only restaurant reviews are annotated. Therefore authors use restaurant reviews as the experiment data. The restaurant reviews consist of about three thousands lines of English sentences from [14]. They randomly cut out one eighth as validation dataset, and use the rest for training purpose, and additional unavailable restaurant reviews in the dataset of [13] are used for testing. The set of aspects is {food, service, price, ambience, anecdote}. There are four sentiment categories: positive neutral, negative and conflict. Each sentence is assigned one or more aspects together with a polarity label for each aspect.

The result is quite good. There are in total three subtasks: aspect detection, sentiment classification on given aspects and aspect-level sentiment analysis. For the first subtask, they compare AS-Capsule with state-of-the-art baselines designed for aspect detection. More specifically, compare detailed F1 of all categories with effective joint deep learning baselines including Bi-LSTM, AE-LSTM, AT-LSTM and RNN-Capsule. KNN is the baseline provided by SemEval official, but in terms of the performance, AS-Capsules are the second best, after Hybrid-WRL [15], which is a word representation learning method using hybrid features including shared-features and aspect-specific features. For the second subtask, which is to predict the sentiment expressed in the text on the given aspect. The authors compared AS-Capsules with four baselines: Bi-LSTM, AE-LSTM, AT-LSTM and Attention based LSTM with Aspect Embedding. AS-Capsule achieves the best accuracy of 85.0. It outperforms all baselines with respect to F1 result on the positive and negative sentiment categories. However, for the neutral sentiment which is also the most difficult sentiment, there is not sufficient data, it performs not good enough, after ATAE-LSTM and AE-LSTM. For the third subtask, which is to detect <aspect, sentiment> pairs from the input text, the authors compare AS-Capsules with Bi-LSTM, AE-LSTM and AT-LSTM. AS-Capsule leads the competition in all possible ways and achieves an accuracy of 68.1

I think the second and the third literature are very inspiring, in a sense that it applies the concept of capsule into the field of sentiment analysis, and adapt different methodology for different tasks, e.g. use an RNN for the task of pure sentiment analysis, and later add new components to support more specific aspect-level sentiment analysis. I think deep learning has a huge potential in the field of sentiment analysis, and we will be able to see more good papers on this topic that utilize the power of deep learning.

REFERENCES

[1] Agarwal, A., Xie, B., Vovsha, I., Rambow, O., & Passonneau, R. J. (2011, June). Sentiment analysis of twitter data. In Proceedings of the workshop on language in social media (LSM 2011) (pp. 30-38).

[2] Wang, Y., Sun, A., Han, J., Liu, Y., & Zhu, X. (2018, April). Sentiment analysis by capsules. In Proceedings of the 2018 world wide web conference (pp. 1165-1174).

[3] Wang, Y., Sun, A., Huang, M., & Zhu, X. (2019, May). Aspect-level sentiment analysis using as-capsules. In The World Wide Web Conference (pp. 2033-2044).

[4] Google Scholar. (n.d.). Google Scholar. Retrieved November 9, 2021, from https://scholar.google.com.sg/scholar?cluster=16109791220904979787&hl=en&as_sdt=0,5&as_vis=1&inst=10972715779114120479

[5] Ethan Xia, Han Yue, and Hongfu Liu. 2021. Tweet Sentiment Analysis of the 2020 U.S. Presidential Election. In <i>Companion Proceedings of the Web Conference 2021</i> (<i>WWW '21</i>). Association for Computing Machinery, New York, NY, USA, 367–371. DOI:https://doi.org/10.1145/3442442.3452322

[6] Papers with Code - MR Benchmark (Sentiment Analysis). (n.d.). MR Benchmark. Retrieved November 9, 2021, from https://paperswithcode.com/sota/sentiment-analysis-on-mr

[7] Bo Pang and Lillian Lee. 2005. Seeing Stars: Exploiting Class Relationships for Sentiment Categorization with Respect to Rating Scales. In Proc. ACL. 115–124.

[8] Richard Socher, John Bauer, Christopher D. Manning, and Andrew Y. Ng. 2013. Parsing with Compositional Vector Grammars. In Proc. ACL, Volume 1: Long Papers. 455–465.

[9] Richard Socher, Jeffrey Pennington, Eric H. Huang, Andrew Y. Ng, and Christopher D. Manning. 2011. Semi-Supervised Recursive Autoencoders for Predicting Sentiment Distributions. In Proc. EMNLP. 151–161.

[10] Richard Socher, John Bauer, Christopher D. Manning, and Andrew Y. Ng. 2013. Parsing with Compositional Vector Grammars. In Proc. ACL, Volume 1: Long Papers. 455–465.

[11] Yequan Wang, Aixin Sun, Jialong Han, Ying Liu, and Xiaoyan Zhu. 2018. Sentiment Analysis by Capsules. In Proc. WWW. 1165–1174.

[12] Yequan Wang, Minlie Huang, Li Zhao, and Xiaoyan Zhu. 2016. Attention-based LSTM for Aspect-level Sentiment Classification. In Proc. EMNLP. 606–615.

[13] Maria Pontiki, Dimitris Galanis, John Pavlopoulos, Harris Papageorgiou, Ion Androutsopoulos, and Suresh Manandhar. 2014. SemEval-2014 Task 4: Aspect Based Sentiment Analysis. In Proc.Workshop on Semantic Evaluation, SemEval@COLING. 27–35.

[14] Gayatree Ganu, Noemie Elhadad, and Amélie Marian. 2009. Beyond the Stars: Improving Rating Predictions using Review Text Content. In Proc. WebDB.

[15] Xinjie Zhou, Xiaojun Wan, and Jianguo Xiao. 2015. Representation Learning for Aspect Category Detection in Online Reviews. In Proc. AAAI. 417–424.