Paper Critique - 2

Intro of Data Mining - Fall, 2021

Student

Aissata Diallo (Undergraduate) & Sheng Kai Liao (Graduate)

Date:10/21/2021

Paper:

Neural Topic Models for Hierarchical Topic Detection and Visualization

Critique: (Eddy)

In this paper, Pham et. al, deliver a visual hierarchical neural topic model which not only is able to analyze and identify the topic hierarchies but also provide a visualisation functionality for showcasing the relationship between documents and topics in 2-D dimensional space. Such a model could be useful for the NLP domain considering the importance of data visualization decreasing the difficulties of data interpretation for humans. Since it is a novel model in the such domain, authors modify several models as baseline with different alternative methodologies in order to give a pair comparison between them and their own model. For instance, LDA is a NLP model which is able to learn the topics but not their structure. At the same time, nCRP (the nested Chinese Restaurant Process) and TSNTM are hierarchical topic models but do not provide visualization of topics and documents. For most recent topic models, PLSV and its variants, which include topic hierarchical detection and visualization yet due to the models being flat topics models, they couldn't train the data and visualize hierarchically.

In general, this paper HTV model which could detect and visualize the hierarchical topic.

To achieve that, HTV will create an infinite tree in which nodes will be represented as different topics for this case. Meanwhile, the most general topic will be at the root nodes and the more specific topics will be treated as left nodes. To create an infinity tree for recording unpredictable amounts of topics, authors employ doubly-recurrent neural networks (DRNN) to address this issue. At the same time, in order to visualize the hierarchical topic, the model employs a graph layout function Kamada-Kawai algorithm (KK) as a visualization tool. Finally, the paper experiments their HTV model and the baselines with four different dataset, BBC which only have 2225 documents within 5 classes (topics), Reuters which has 7674 articles within 8 categorises, 20 Newsgroup has 18251 posts within 20 categorises and Web of Science includes 46985 paper's abstract and keywords from 7 research domains.

In order to measure the quality of the tree-structure and visualization, this paper offers several estimation methods in different aspects, such as document specialization in visualization space to focus on the correlation of general documents and root nodes in 2-d space, classification in visualization space for testing the quality of scatter plot visualization via K-NN accuracy, and hierarchical affinity to measure how the child nodes are similar with their parent node by comparing the average of cosine similarity of its children node and the reset of non-related children nodes. According to the experiments, HTV achieved competitive results with the best baseline's performance in each measurement I mentioned above. Meanwhile, comparing the computational cost, HTV has better training efficiency from nCRP and has competitive results compared to PLSV. Also, by observing the scatter plot directly, HTV has clearier document cluster compared to the baselines which could help people understand the result easily.

In conclusion, this paper provides a novel visual hierarchical neural topic model, HTV for detecting and visualizing the topic hierarchy over documents. By experimenting with other

state-of-the-art topic hierarchy models, HTV not only obtains competitive results in many aspects but also provides clearer vision of clustering in visualization. In my words, I think HTV could have a good chance to be a novel research tool to help the researchers easier observe the correlation between documents and topics in order to resolve their problems or support their assumptions.

Paper:

NA-Aware Machine Reading Comprehension for Document-Level Relation Extraction

Critique: (Eddy)

According to Zhang et. al, the traditional NLP feature extraction methods, such as sentence-level relation extraction, provides limited information from the data for the NLP model. To break through the limitation, several contributions have been offered to move sentence-level relation extraction to document-level relation extraction with significant success. However, those contributions basically treat the document as a long sentence and generate the representations which are target-agnostic. Based on that, the authors think that the representations may contain noise which may also confuse the model. Hence, inspired by the nowadays NPL problems, MRC style task (Machine Reading Comprehension), the authors focus on addressing document-level relation extraction representations issue with their own NA-aware MRC (NARC) model.

At the beginning, NARC follows the MRC task to formulate the head entities as query, but afterward it adds tail entities as the candidate answers, then employs the relation classification for each candidate tail entities. To achieve that, the input firstly be formulated as a specific form and fed it into the query context encoder which can be considered as a pre-train NLP model such as BERT. Furthermore, in MRC style, it often generates the queries which have no correct answer (65.5 % in DocRED dataset) which means the MRC can find the great relation between the document and its entities based on the dataset. To solve the issue, the author purpose a NO-ANSWER attribute for each query for defining those no correct answers. Yet, since incorrect answers usually take a big proportion of results, the NO-ANSWER oftenly causes the imbalance of data distribution. In order to minimize the affection of imbalanced, the author introduce a novel answer vector assembler model to integrate representation in each layers of encoder as final output for each entities, and decompose the NO-ANSWER candidate into related component and not related component, then compose both as a query-specific NA vector.

For the experiment, they compared their own NARC mode to 5 BERT without model such as CNN, BiLSTM, ContextAware, AGGCN and Eog, and 5 BERT within model, BERT-RE, BERT-HIN, BERT-Coref, BERT-GLRE and BERT-LSR. All models were implemented with DocRED dataset. For the estimation, the authors choose micro F1 and micro Ign F1 as evaluation metrics. Ign F1 reflects the additional relation fact within the training set and testing set.

Regarding the experiment result, NARC obtains best performance over the most evaluation metrics. In addition, this paper also analyses the performance in three aspects, various distances of entities, various amount of entities mentions and evidence sentence, comparing their model to two well performanced BERT variants, BERT-MRC and BERT-RE. In all the testing, the NARC gains best performance in all the cases. However,the authors found some interesting observations. For distance of entities, as expected, there models consistently dropping while the distance of entities increasing due to the dissimilarity between the entities are enlarging. For the amount of entities mentions, interestingly three models have best performance when the entities mentions 3 times which the authors think too much time of entities mention may also confuse the

models. For the number of evidence sentences, three models reach the best when three evidence sentences are provided since basically the models consider three entities (head entity, tail entity, and a relay entity). Finally, for computation cost, NARC cost similar time with the state-of-the-art MRC style model.

For conclusion, this paper offered a novel NA-aware MRC (NARC) model as a refined model of MRC which successfully improved the document-level relation extraction performance with slightly computation cost increasing. I think the NARC model could be a stepstone for maximizing the NLP relation extraction task in order to obtain greater and more valuable information.

Paper:

Taking Over the Stock Market: Adversarial Perturbations Against Algorithmic Traders

Critique: (Aissata)

Machine learning is used in algorithmic trading in order to predict the markets behaviour and implement an investment strategy. Even though machine learning has been deemed to be vulnerable to adversarial examples known as manipulation of inputs; there is limited information on the impact of adversarial learning on the trading domain. In this paper Nehemya et.al demonstrate the use of adversarial learning such as target universal algorithmic perturbation (TUAP) by an attacker to manipulate the data streams in algorithmic trading. An attacker is able to accomplish the manipulation of inputs through the creation of universal adversarial perturbation that is agnostic to the targeted model and based on time of use, which remains impossible to perceive when added to the input stream. Also through adversarial examples, data samples are maliciously modified to create confusion and misclassification by the targeted model. Adversarial perturbations are a threat to machine learning and to the functions of algorithmic trading because it threatens the reliability of machine learning models and could potentially compromise "sensitive applications." Furthermore, the researchers determined that through the exploitation of adversarial examples an attacker can gain access to algorithmic trading (AT) bot's alpha model and as a result alter the system's actions.

Although adversarial perturbations in algorithmic trading are rare since the data is unpredictable, extremely dynamic and usually under surveillance by law enforcement agencies, the ability of attackers to control multiple algorithmic systems is a risk to the entire stock market.

In this research, Nehemya et.al investigated the adversarial perturbation in algorithmic trading domains through the use of realistic scenarios where an attacker manipulates the High Frequency Trading (HFT) data stream to gain access to bot's in real time. They show an algorithm that utilizes common market data to create a targeted universal adversarial perturbation (TUAP), which can deceive the alpha model. Their method was evaluated using real world stock data and three different prediction models in both white-box and black-box settings. They presented various mitigation methods and discussed their limitations, based on the algorithmic trading domain. The researchers show that perturbation can deceive the trading algorithms in long term unseen data points, in both white-box and black-box settings, when added to the input stream.

In the observation, they set up real intraday market data from the S&P 500 and they made sure that each stock contained the open high and low close at one minute intervals. They defined the manipulation input sample as a stream of 25 continuous one-minute records, where each record consists of details about the stock such as the opening price, high price, low price, closing price, and the volume the stock was trading in at the end of the minute. They divided the dataset into three sets. A set for training the alpha models, a set for crafting TUAPs, and six test sets to evaluate the attack. The data between the dates 11/9/2017-1/1/2018 was used to train the alpha model. To craft the TUAPs, 40 samples were uniformly sampled for each of three trading days which resulted in 120 samples in total. The data generated between the five trading days per week 5/1/2018-

15/2/2018 were used to create six test sets from T1 to T6. For each week they built a corresponding test set by uniformly sampling 100 samples that represent an increase of the stock price and an additional 100 that represent a decrease of the stock price. Next, they performed processing on the raw data, before feeding the input into the model. While only using the closing price of each minute, they formed five groups of five consecutive minutes, and for each group, they extracted the following features: the trend's indicator, standard deviation of the price, and average price. The trend indicator is set to be the linear coefficient among the closing prices of five consecutive minutes. The features extracted from the sliding window of 25 minutes of the raw data are used to build one input sample for the alpha model.

For the experiment they examined five stocks: Google (GOOG), Amazon (AMZN), BlackRock (BLK), IBM, and Apple (AAPL). For each stock, they used algorithm 1 to craft three TUAPs, one for each alpha model, and randomly sampled three perturbations that are the same size as the TUAPs for instance 0.02% of the stock's price. Then they evaluated the attack performance on the six test sets T1 −T6. Their expectation was that the performance of the TUAPs will gradually degrade as they moved away in time from the time period of the training set. Therefore, the TUAP will achieve the highest TFR and UFR for T1, and the lowest for T6. The results showed that TUAPs can create dramatic effects in all alpha model classifications, thus allowing the attacker the ability to use the TUAP to control Alpha model's prediction.

In conclusion, Nehemya et.al demonstrated how adding a TUAP in real time to the market data stream allows an attacker to control the alpha model's predictions, thus gaining access and control to all the algorithmic trading systems. The researchers hope to use their findings as a warning to the finance community regarding the threats in this area and promote further research on the risks associated with using automated learning models in the trading domain. This was a very interesting way of showing how adversarial examples impact not only machine learning but the stock market as well. Even though TUAP is not a common problem in Stock Trading right now, I do believe that it is wise for us to know that such threats do exist so that we can prepare to mitigate and remove such threats in the future.