Journal of Artificial Intelligence and Capsule Networks (2020)
Vol.02/ No. 03
Pages: 163-169
http://irojournals.com/aicn/
DOI: https://doi.org/10.36548/jaicn.2020.3.003

TransCapsule Model for Sentiment Classification

Dr. Akey Sungheetha,
Data Science SIG member,
Computer Science and Engineering, School of Electrical Engineering and Computing,
Adama Science and Technology University, Adama, Nazret, Ethiopia.

Dr. Rajesh Sharma R,
Image Processing SIG member,
Computer Science and Engineering, School of Electrical Engineering and Computing,
Adama Science and Technology University, Adama, Nazret, Ethiopia.

Abstract: Aspect-level sentiment classification is the aspect of determining the text in a given document and classifying it according to the sentiment polarity with respect to the objective. However, since annotation cost is very high, it might serve a big obstacle for this purpose. However, from a consumer point of view, this is highly effective in reading document-level labelled data such as reviews which are present online using neural network. The online reviews are packed with sentiment encoded text which can be analyzed using this proposed methodology. In this paper a Transfer Capsule Network model is used which has the ability to transfer the knowledge gained at document-level to the aspect-level to classify according to the sentiment detected in the text. As the first step, the sentence is broken down in semantic representations using aspect routing to form semantic capsule data of both document-level and aspect-level. This routing approach is extended to group the semantic capsules for transfer learning framework. The effectiveness of the proposed methodology are experimented and demonstrated to determine how superior they are to the other methodologies proposed.

Keywords: Trans-Capsule; aspect-level semantic; sentiment classification; Neural Network;

1.Introduction

In sentiment analysis, aspect-level sentiment classification (ASC) is a small part of the sentiment analysis online that can be used to analyze the polarity of the statement by reading the document [1]. ASC is said to determine the sentiment polarity when an aspect and a sentence is given. Traditionally, machine algorithms were commonly used, with specific features that can help to develop sentiment classifiers for the algorithm [2]. These methodologies will require massive linguistic resources or laborious feature engineering. Over the years many models have been proposed that have incorporated neural networking with deep learning technique to perform ASC [3]. All these models will require a pre-defined number of data to train the classifier in order to get the right output. The biggest drawback [4] is that the annotation of the target turns out to be very expensive in ASC. Lack of labelled data is another huge drawback in using this algorithm [5]. Moreover, datasets that is publicly available for this purpose will hold only a small count of training samples. However, document-level labelled data can be found in websites like Amazon and Yelp where reviews can be accessed easily [6-7]. In general, these reviews will be coupled with a rating factor which is given by the customer, representing the polarity sentiment. This data in the document-level will hold important sentiment information that can be used to analyse the data in aspect-level as they have similar semantic and linguistic patterns [8]. However, the use of document-level data is done by only one study in order to perform the ASC tasks.

Similarly, a framework holding PRET+MULT [9]was proposed in 2018 which attempted the use of multi-task and pre-training learning approaches. But their model will share only the LSTM layers and shallow embedding

ISSN: 2582-2012 (online)
Submitted: 20.05.2020
Accepted: 10.07.2020
Published: 17.07.2020

between the DSC and ASC tasks [10]. The drawback with this approach is that this algorithm was not able to handle more complicated patterns such as irony and euphemism which will need the use of high-level semantic knowledge to understand the complete sentence [11]. For example, while reading the sentence, "The product could have been a bit more worth the money", PRET+MULT will not be able to predict the negativity of the sentence as it is not possible to identify the negative polarity directly. Hence to overcome this issue, we have proposed the use of Transfer Capsule Network model which has the capacity to change the semantic knowledge of the sentence from its preliminary DSC [12] form to that of ASC. A dynamic routing approach is proposed to cluster and store the features such that sentence-level features can be generated by DSC and ASC. We have also extended this approach to make it adapt to the transfer learning framework. Observation of the output showed that the proposed methodology could excel other similar methodologies.

2. TransCap Model

"For a given leaning task T_A, target aspect-level corpus C_A, learning task T_D and document-level corpus C_D, the proposed TransCap is used to improve the function of target prediction $f_A(.)$ by transferring knowledge to T_A from T_D." In 2011, capsule network was initially proposed to be used in computer vision for image classification. When compared with Convolutional neural network, the proposed TransCapsule will incorporate vector-output capsules in the place of scalar-output feature detection as it can preserve extra information like thickness and position. There are two capsule layers in the vanilla CapsNet namely class layer and primary layer. The class layer is used to generate probability of classification based on individual capsules that belong to one class while the primary layer holds image feature maps of low-level. In recent years, CapsNet has been used in many NLP tasks such as relation extraction and text classification. With the use of dynamic routing, CapsNet will be able to transfer data between the two layers. Moreover, every class in CapsNet hold parameters which are distinctive in nature. In a typical transfer learning scenario, CapsNet will be able to satisfy our need by including many tasks and polarities. The proposed methodology accomplishes ASC tasks by exploiting CapsNet.

3. Proposed Methodology

3.1 Architecture

Fig.1. describes the architecture of the proposed TransCap comprising of 4 layers

- **ClassCap Layer:** The Class capsules are generated at this layer, denoting T_D and T_A as the sentiment polarities.
- **SemanCap Layer:** This layer separates the various feature capsules and further categorizes them into semantic capsules in the sentence-level with respect to aspect.
- **FeatCap layer:** This layer will transform the extract from the word vectors features (N-gram) into feature capsules.
- **Input Layer:** This layer will represent the words as real-valued vectors in low dimension.

It is worth noting that the tasks T_D and T_A are shared by the all the layers except ClassCap layer. This is because, both the tasks are related and are used to realise the sentiment features and polarity, thereby complementing each other. Thus the features produced by these layers will help improve the architecture together.

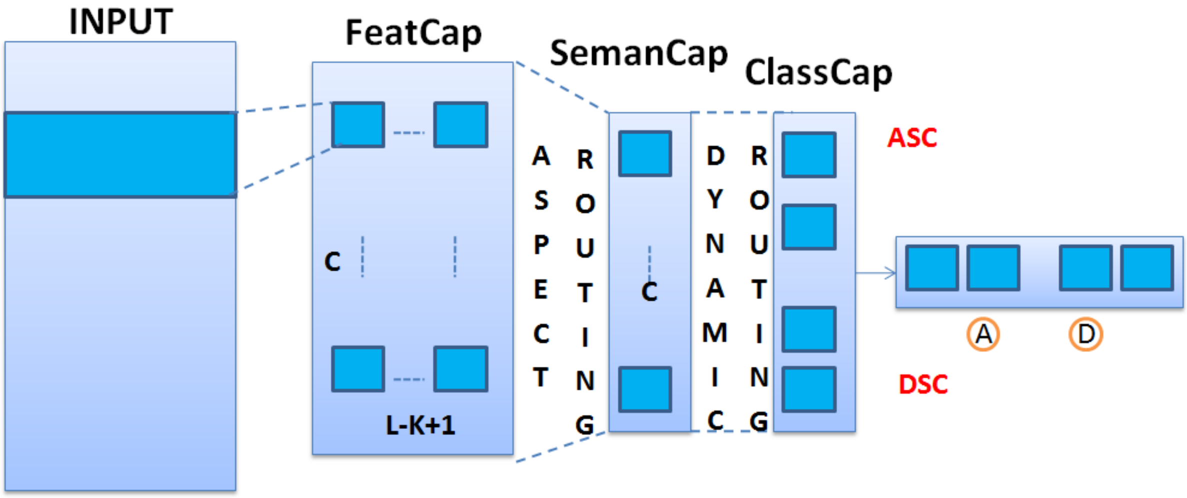

Fig.1. TransCap Architecture

3.2 Class Capsule Layer

A typical capsule network will involve a classification task that can represent their lengths and classes as classification probabilities with the help of class capsules. But, these are considered to be different problems and will require sentiment polarities to be separated into classes. In order to accomplish this, we have developed within the TransCap 2 classes amounting to 6 capsules. This will further ensure that T_D and T_A are trained in a similar framework.

3.3 Semantic Capsule Layer

When an aspect term appears in a document or sentence, two copora C_D and C_A will occur. These tasks play an important role in T_A but are irrelevant as far as T_D is concerned. T_A will use this data to find the relation between aspects and contexts. When a sentence holds sentiments of opposite polarities, they must be identified and separated as different contexts based on the difference in their aspects. For example, consider the sentence “That was a great show but the lighting was horrible!” In this sentence the word “horrible” should be complementing the aspect “lighting” and also contrasting the aspect “show”.

ISSN: 2582-2012 (online)
Submitted: 20.05.2020
Accepted: 10.07.2020
Published: 17.07.2020

Hence we have proposed a routing approach which acts based on the weight of the context work on calculation with respect to T_A. We propose to use fusion convolution operation in Y along with the kernel F_a∈ $R^{d_h X\ K}$. Here a represents the routing weight of aspect such that:

$$a_i = signoid\ (Y_{i:i+K} * F_a + T_a e_a + b_a)$$

Here b_a is the bias, T_a is a transfer matrix such that $T_a \in R^{l\ X\ d_w}$ which is used to coorelate the values of b_a and e_a. Aspect information is also fused with the routing weight a_i∈ [0,1]. This will determine if information can be sent to the subsequent layer. When the value for a_i is found to be 0, feature capsule will be completely prevented from transmitting to the next layer. When T_D is confronted with absence of aspect in the document, it needs to identify the difference between C_D and C_A which proves to be a challenge. To help in this scenario, we have designed j_i as a piecewise function that can be determined from the values of X and r_i such that:

$$g_i = \begin{Bmatrix} a_i, & X\epsilon C_A \\ 1{,}0 & X\epsilon C_D \end{Bmatrix}$$

On sliding in X, we can determine the value of g for the complete S(D). This process can be repeated C number of times, representing the C channels of feature capsule, to obtain the weights of aspect routing G∈ $R^{C\ X\ l\ X\ (L-K+1)}$. Routing of the capsules is finally accomplished by using the weights:

$$Q=R \odot G$$

where ⊙ represents multiplication (element wise) and Q∈ $R^{C\ X\ d\ X\ (L-K+1)}$.

3.4 Feature Capsule Layer

The n-gram features are extracted from their sentences in this layer. Within a pre-defined window, these features will hold local and raw semantic meaning. In order to extract the ith feature, we can use multiple convolution such that

$$r_i = X_{i:i+k} * F + b$$

where r_i is the feature vector for ith n-gram, d_p is a feature capsule's dimension, K is the size of n-gram and (d_hx K) represents convolutional kernel size. When F $\in R^{d_h X\ (d_h XK)}$ is implemented in X, we can obtain r such that r $\in R^{d_p X (d_h XK)}$. It represents the encapsulating features that are obtained from the sentence. This process is repeated many times in order to obtain the different channels of feature capsules that are used to represent C. In this equation F represents only one category of semantic meaning. Based on the observations made, the final output is represented as:

$$P \in R^{CXd_pX\ (L-K+1)}$$

ISSN: 2582-2012 (online)
Submitted: 20.05.2020
Accepted: 10.07.2020
Published: 17.07.2020

3.5 Input Layer

The input layer is made up of two layers. A look-up table comprising of pre-trained words is represented as:

$$E_w \in R^{d_w X|V|}$$

where |V| is the size of the vocabulary, and d_w is the word vectors' dimension. Thus S (D) the word sequence can be mapped to word vectors listed in the lookup table such that $\{w_1,...,w_a,...,w_L\}$.

In addition to this, we have also used an alternative additional position lookup table. Based on the calculation of the absolute distance between v_a (aspect word) and v_i (context word) will be used to determine the extra position in the sequence S. If there is no information on the aspect, then the position sequence will be '0' for T_D. Let the lookup table for position embedding be represented by $E_l \in R^{d_w X|V|}$, then it can be mapped to a series of position vectors such that $\{p_1,...,p_a,...,p_L\} \in R^{d_w X|V|}$. Thus the sentence can be completely changed into $Y=\{y_1,...,y_L\} \in R^{d_h X\ L}$.

4. Results and Discussion

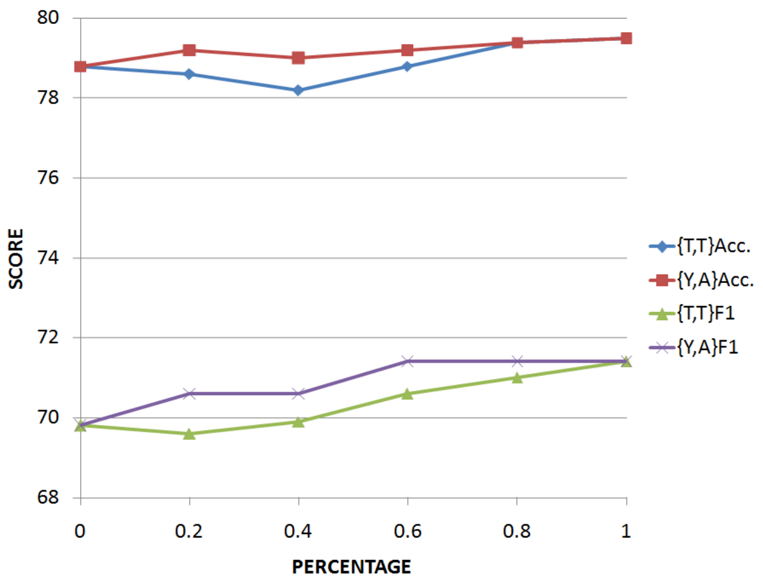

Fig.2. Influence of C_D

ISSN: 2582-2012 (online)
Submitted: 20.05.2020
Accepted: 10.07.2020
Published: 17.07.2020

Changes in T_A are observed by varying the size of the Corpus, C_D. This represents how DSC tasks will influence the ASC task. As shown in Fig.2, an increase in knowledge of document, a subsequent increase will be reflected in the curve. Our model's effectiveness can be observed from this curve as knowledge is transferred from the document. In the beginning when only 15% or 35% of C_D are given, there is a slight fall in the performance. On getting enough C_D, the values of T_D and T_A become stable.

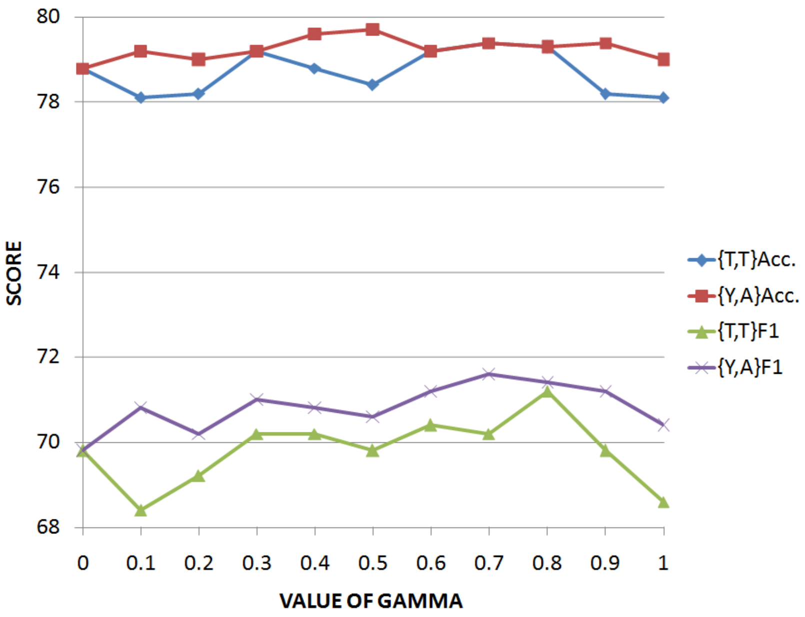

Fig.3. Effects of γ

Similarly the DSC task will also vary based on the effect of γ. In the Fig.3, there is a turning point at 0.7 such that for values of γ greater than TP the curve has a download or flat and for values less than TP, the curve turns upward. Multi-task learning mechanism can be used to better understand this phenomenon. In the upward direction, T_A performance can be improved greatly with the help of sentiment knowledge. At the point where T_D exceeds that of T_A, the value of T_D will begin to dominate the entire model and T_A will begin to deteriorate in performance while losing mastership over the progress.

5. Conclusion

In this proposed work, we present a new Capsule model known as TransCap to classify aspect-level sentiment. This work addresses the issues of data that is labelled in the aspect-level with the help of document-level data that is present in abundance. The knowledge is transferred to the aspect level task from document-level task using simple routing algorithms like dynamic routing and aspect routing. Comparing the output obtained with other baselines show that the proposed approach outperforms the others by a considerably large difference. The proposed methodology can be further enhanced by implemented new technological inventions in the prospect of reading the text and identifying sentiments such as sarcasm.

References

[1] Yukun Ma, Haiyun Peng, and Erik Cambria. 2018. Targeted aspect-based sentiment analysis via embedding commonsense knowledge into an attentive LSTM. In AAAI Conference on Artificial Intelligence (AAAI 2018).

[2] Sinno Jialin Pan and Qiang Yang. 2010. A survey on transfer learning. IEEE Transactions on Knowledge and Data Engineering.

[3] Diederik P. Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. Computer Science.

[4] Zhu, Z., Peng, G., Chen, Y., & Gao, H. (2019). A convolutional neural network based on a capsule network with strong generalization for bearing fault diagnosis. *Neurocomputing*, *323*, 62-75.

[5] Xiang, C., Zhang, L., Tang, Y., Zou, W., & Xu, C. (2018). MS-CapsNet: A novel multi-scale capsule network. *IEEE Signal Processing Letters*, *25*(12), 1850-1854.

[6] Koresh, M. H. J. D., & Deva, J. (2019). Computer vision based traffic sign sensing for smart transport. *Journal of Innovative Image Processing (JIIP)*, *1*(01), 11-19.

[7] Inokuma, Y., Yoshioka, S., & Fujita, M. (2010). A molecular capsule network: guest encapsulation and control of Diels–Alder reactivity. *Angewandte Chemie International Edition*, *49*(47), 8912-8914.

[8] Vijayakumar, T. (2019). Comparative study of capsule neural network in various applications. *Journal of Artificial Intelligence*, *1*(01), 19-27.

[9] Kim, Y., Wang, P., Zhu, Y., & Mihaylova, L. (2018, October). A capsule network for traffic speed prediction in complex road networks. In *2018 Sensor Data Fusion: Trends, Solutions, Applications (SDF)* (pp. 1-6). IEEE.

[10] Vijayakumar, T., & Vinothkanna, M. R. (2020). Capsule Network on Font Style Classification. Journal of Artificial Intelligence, 2(02), 64-76.

[11] Iesmantas, T., & Alzbutas, R. (2018, June). Convolutional capsule network for classification of breast cancer histology images. In International Conference Image Analysis and Recognition (pp. 853-860). Springer, Cham.

[12] Inokuma, Y., Ning, G. H., & Fujita, M. (2012). Reagent-Installed Capsule Network: Selective Thiocarbamoylation of Aromatic Amines in Crystals with Preinstalled CH3NCS. *Angewandte Chemie International Edition*, *51*(10), 2379-2381.