Aspect extraction on user textual reviews using Multi-Channel Convolutional Neural Network

Aminu Dau Corresp., [1] , **Naomie Salim** [1]

[1] Computing, University of Technology Malaysia, Skudai, Johor, Malaysia

Corresponding Author: Aminu Dau
Email address: dauaminu@graduate.utm.my

Aspect extraction is one of the important subtasks of aspect-based sentiment analysis. Existing approaches to aspect extraction typically rely on using handcrafted features, linear and integrated networks architectures. Although these methods can achieve good performance, however, they are time-consuming and often very complicated. Thus, in this paper, we present a multichannel convolutional neural network for aspect extraction. The model consists of deep CNN with two input channels. One for word embedding to encode semantic information of the words and the other for POS (part of speech) tag embedding to facilitate the sequential tagging process. To get the vector representation of words, we initialized the word embedding channel and POS channel using pre-trained word2vec and one hot vector of POS tags respectively. Both the word embedding and POS embedding vectors were fed into the convolutional layer and concatenated to a one-dimensional vector which is finally pooled and processed using a softmax function for sequence labeling. We finally conducted a series of experiments using four different datasets, namely SemEval2014-L, SemEval2014-R, SemEval2015-R and SemEval2016-R datasets. The results indicated better performance with the integration of POS tag embedding layer and fine-tuned domain specific embeddings. Further comparison showed that our approach outperformed the state- of-the-art methods .

Aspect Extraction on User Textual Reviews Using Multi-Channel Convolutional Neural Network

Aminu Da'u [a] *, Naomie Salim [a]

[a] **Faculty of Computing University of Technology Malaysia, Malaysia.**

***Corresponding Author Tel: Tel: +2348069763032, +60187922528**

Email Addresses: dauaminu@graduate.utm.my (A. Da'u), Naomie@utm.my (N. Salim)

Abstract

Aspect extraction is one of the important subtasks of aspect-based sentiment analysis. Existing approaches to aspect extraction typically rely on using handcrafted features, linear and integrated networks architectures. Although these methods can achieve good performance, however, they are time-consuming and often very complicated. Thus, in this paper, we present a multichannel convolutional neural network for aspect extraction. The model consists of deep CNN with two input channels. One for word embedding to encode semantic information of the words and the other for POS (part of speech) tag embedding to facilitate the sequential tagging process. To get the vector representation of words, we initialized the word embedding channel and POS channel using pre-trained word2vec and one hot vector of POS tags respectively. Both the word embedding and POS embedding vectors were fed into the convolutional layer and concatenated to a one-dimensional vector which is finally pooled and processed using a softmax function for sequence labeling. We finally conducted a series of experiments using four different datasets, namely SemEval2014-L, SemEval2014-R, SemEval2015-R and SemEval2016-R datasets. The results indicated better performance with the integration of POS tag embedding layer and fine-tuned domain specific embeddings. Further comparison showed that our approach outperformed the state-of-the-art methods.

Keywords: Convolutional Neural Network, Aspect Based Sentiment Analysis, Aspect Extraction, Multi-channel CNN.

1. Introduction

Recently aspect-based sentiment analysis becomes a promising research dimension thereby attracting many attentions from the research community. One of the important subtasks of aspect-based sentiment analysis is the aspect of extraction. Aspect extraction is simply the act of extracting attributes of an entity about which opinions are expressed [1]. Aspect extraction can generally be performed using either unsupervised methods [2-6] or supervised methods [7-9]. Our work particularly focuses on the supervised deep learning approach. For many years, the state of the art methods of aspect extraction basically depends on the conditional random fields (CRF) [8], recurrent neural network (RNN) [10] or linguistic patterns and syntactic rules [5], [11]. Both of these approaches have their own shortcomings. For example, Conditional Random Field (CRF) is typically linear in nature. Thus, it requires a large number of datasets to effectively work. RNNs are generally ineffective in predicting word labels or phrases that are determined by the context due to their feedback nature. Syntactic rules and linguistic patterns need to be hand-crafted and their accuracy generally depends on the grammatical accuracy of the sentence.

To address the aforementioned issues among others, few approaches have been proposed to exploit deep Convolutional Neural Network architectures to improve the performance of the aspect extraction models [12-14]. These models do not usually require predefined features to be manually handpicked, instead, they can automatically learn sophisticated features from their dataset. Generally, words are usually represented in form of a vector and the extraction of the feature is left to the network. Consequently, words with similar semantics can be mapped using these models to nearby locations in their coordinate system.

Even though these approaches have shown better performance than their prior approaches, however, there are some important issues worth to be considered for further improvement: Firstly, most of the existing approaches typically used only general pre-trained word embeddings such as Google Word2vec or Glove embeddings as the main semantic feature for the aspect extraction. Although word embeddings have shown effectiveness in capturing both syntactic and semantic information of words. However, in some cases, due to the distributional hypothesis, word embeddings alone fail to efficiently capture the syntactic information of some aspects terms, for example, in the latent space, bad and good are typically mapped together as neighbors while analyzing these words is very critical in aspect classification. Moreover, due to the complexity of aspect extraction task, fine-grained embeddings are particularly important to achieve a better performance [16]. Therefore, we urge that using a domain-specific embedding is very crucial for the information extraction performance. Thus we exploit both the general and domain-specific embeddings to examine which embeddings are superior over the other.

Additionally, many aspect extraction approaches in the past, particularly rely on either lexicons, handcrafted features or integrated models. For example, most previous CNN models for aspect extraction are either stacked [17] or integrated with other models such as LSTM [18]. These consequently increase the complexity of the model parameters. Although these may improve the model performance, however according to [19], in real-world applications, a simple model is always preferred and more useful over the complicated model. This is particularly important when a model is used for a real-life situation such as chatbot in which a complex model will retard the inferential performance of the model. Thus, achieving a competitive performance while ensuring a simple architecture without manually crafting feature and much complexity is always a crucial direction to explore. This paper proposes to achieve such a goal.

Furthermore, it is worth mentioning that, despite many successes in using POS tag for the NLP task such as sentiment analysis and aspect extraction, many deep learning methods ignore to fully utilize POS tags for the aspect extraction. They generally exploit the feature to some extent without paying more attention to the integration methods while feature embedding integration methods have been shown to be very crucial in the deep CNN models performance [20].

Thus this paper proposes a deep multichannel convolutional neural network leveraging two different embedding layers, namely, word embedding layer and POS tag embedding layer for an effective aspect extraction. For the word embeddings, we particularly use google word2vec [21] trained on the 100 billion words of google news corpus as the general embedding and domain-specific embedding in which we trained CBOW architecture on the Amazon and Yelp reviews datasets for the laptop and restaurant domain respectively. For the POS embeddings similar to [22] we use a Stanford tagger with 45 tags and apply a one hot vector encoding to generate a 45-dimensional vector.

To achieve a simple architecture while ensuring a competitive performance, we propose a purely CNN model for sequential labeling. Unlike the LSTM models whose main drawbacks is that is sequentially inclined in which the backpropagation and forward pass must sequentially go through the whole process which make it slow in the training process. CNN model which is nonlinear network architecture can fit data more easily with relatively few parameters and has been successfully used in many works involving NLP (natural language processing). The major contribution of our work can be summarized as follows:

1. We introduce a simple multichannel convolutional neural network model leveraging two different input channels for aspect extraction, namely, word embeddings and POS Tag embeddings channel to encode the contextual information and enhance sequential tagging of words respectively.
2. We showed the importance of using domain-specific embeddings over the general purpose pre-trained embeddings in aspect extraction.
3. We showed that our approach leveraging POS tag embeddings channel outperformed the baseline methods with significant gains across all the given datasets.

The remainder of the paper is arranged as follows. Section 2, related work, section 3 the proposed model, section 4 experimental study, section 5 results and discussion and section 6 conclusion and future direction.

2. Related Work

Aspect extraction as the subtask of aspect-based sentiment analysis was prevalently studied by many researchers. One of the earliest studies on aspect extraction was conducted by [23]. They proposed a rule-based method for the explicit aspect categorization. This method was later improved by many approaches among which include Titov & Mcdonald [2] and Popescu and Etzioni [11] who used point-wise mutual information between the product class and noun phrase for product feature extraction.

Generally, aspect extraction can be performed using unsupervised methods such as rule-based, which typically employ either syntactic rules or handcrafted features about some relations [5, 6], topic modeling which uses probabilistic models based on LDA and its variants [3,4,24] and mining of the frequent nouns and nouns phrase [11,25]. Recently, a semi-supervised approach [26,27]have been proposed to further improve the aspect extraction model performance.

The supervised methods typically treat aspect extraction task as a sequence labeling problem. Traditionally, the supervised methods mainly involve hidden Markov model [9] and Conditional random field [7,8]. With the recent success of deep learning in different areas such as image classification and pattern recognition, representation learning for the aspect extraction has become a common trend among the researchers, for instance with RBM [28], a heterogeneous structure is used into a hidden layer to jointly address the problem of sentiment-aspect extraction. [10] used the recurrent neural network and demonstrated that RNN method is superior over the CRF based models. This method was later improved by [29], they applied more sophisticated variants of the Recurrent Neural Network using fine-tuned word vectors and used additional linguistic features for better improvement. To tag each word with non-aspect or aspect label, a multi-layer deep convolutional neural network was proposed by Poria [12]. The authors additionally used syntactic and linguistic patterns to improve the accuracy of the model.

To further improve the effectiveness of aspect extraction, neural attention-based model has been applied to learn the representation of the informative words [30,31]. Tree-based methods have been shown effective for improving the performance of the aspect extraction model. For instance, [16] introduced a dependency path approach in which both the dependency and linear contextual information are considered for the word representation. They further integrated the CRF with word embeddings to further improve the aspect extraction. A similar method was proposed by [32] to exploit the dependency tree along with CRF for the co-extraction of aspect and opinion terms. [33,34] also exploited deep learning for co-extraction of the aspect and opinion terms. To further exploit the sentence information, a tree-based convolutional neural network was introduced by [17]. They applied tree based convolution over a sentence dependency parse tree. Recently a bidirectional dependency tree was proposed by [35] they proposed an end-to-end method to integrate BiLSTM, CRF and word embeddings for aspect term extraction.

Our approach is closely related to the work of [36] who use dual attention for the function satisfiability and product compatibility. Our approach is also relevant to the work of [13] in which a double embeddings method have been used to model aspect extraction using two different in-domain word embeddings. One major drawback of this methods is the failure to utilized the POS tag for the sequential tagging and that POS tag feature has been proved effective for the sequential tagging tasks [12]. Thus, in our approach, POS tags feature are additionally utilized to improve the model performance. Unlike previous methods, we particularly leverage two different channels as the input to the convolutional network architecture. We used both general and domain-specific embedding in one channel specifically to capture the syntactic and semantic information of the word and POS tag embedding in another channel to specifically improve the sequential labeling of the aspect. To the best of our knowledge, this is the first work to leverage a multi-channel CNN deep architecture with word embeddings and POS tag embeddings channels differently for the aspect extraction.

3. Our Model

Figure 1 illustrates the proposed model architecture. The model is based on the CNN structure proposed by KIM [37] which has been modified and successfully used by many researchers [12]. Specifically, the proposed model is made up of CNN architecture with two input layers leveraging word vector and POS vector. It consists of 3 convolutional layers followed by a max pooling strategy, RLU (rectified linear unit) optimizer, a fully connected layer, and a softmax classifier to predict the multi-class labels of aspects with labeling space Y = {B, I, O} with "I", "O" and "B" representing Inside, Outside or Beginning of the aspect term respectively. The model is fine-tuned on different datasets and the input vectors are modified based on the different variants of the model (as described in section 3E). Details of the model are described as follows.

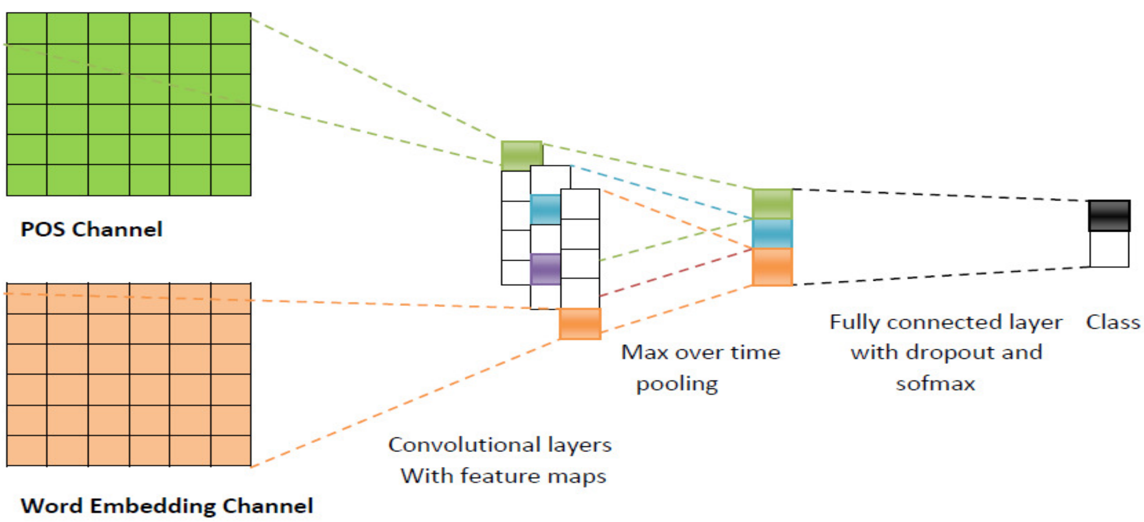

Figure1: Multi-channel CNN

3.1 Input channels

The model typically comprises two sets of vectors, each of which is an input channel to the network. Each channel is acted upon by filters using three regions of different sizes. And the backpropagation of the gradient is only through one of the channels. Similar to [37] each word in the sentence is mapped into a low dimensional vector by lookup layer transformation.

For the word embedding channel, the main idea is to capture the semantic information of the words. For that, we use both general and domain-specific embeddings. For the general embeddings, we particularly apply a pre-trained embedding trained on 100 billion words of google corpus [21]. While for the domain-specific embeddings, we specifically train CBOW[21] model on the Amazon reviews and Yelp reviews for the laptop and restaurant domain respectively. In this case, each word was encoded as 300-dimensional vectors. We use word padding to make sure that all sentences are of the same length. To capture the contextual features of the words, i-th words are mapped to a k-dimensional embedding. The semantic feature of a sentence of length n is given as concatenating all its words embedding which is given as: $|X|_1^n = \{ x_1 ......, x_n \}$, $X \in R^K$.

For the POS Tag embeddings, the main idea is to improve the aspect extraction process based on POS tagging. Specifically, we employ one hot vector in which each tag is transformed into a K dimensional vector. Similar to [22] we use a Stanford POS Tagger with 45 tags. These are encoded as 45-dimensional vector and represented as a matrix. This can be represented as: $|S|_1^n = \{ s_1 ......, s_n \}$, $\in R^{45}$.

3.2 Convolutional Layer

After all the textual information is encoded into vectors and zero padding is applied to make all the embedding channels of the same length, the convolution operations are then applied to generate local features. Thus, the main purpose of the convolutional layer is to extract local features from the embedding layer. Here we use two different filter sizes for POS feature ***P*** and Semantic Feature ***Z*** accordingly. Typically, a convolution is a dot product involving filters with weights $W \in R^{hk}$ and a vector of h-gram in a sentence [37]. Let $w_p \in R^{hk}$ and $w_z \in R^{hx6}$ be filter applied to *h*-gram for the matrix ***P*** *and* matrix **Z** respectively. Then the features generated can be given as:

$$C_i = f(w \cdot x_{i+h} + b) \quad (1)$$

Were f is a nonlinear function (such as hyperbolic tangent or ReLU), *b* stands for a bias term. This is applied to each window, $[x_{1:h},\ x_{2:h+1}, ...x_{n-h:n,}]$. With the $w_p \in R^{n-k+1}$ and $w_z \in R^{n-k+1}$, for the matrix *P and* matrix Z respectively. The features generated for *p* is given by:

$$c_p = [c_1^p, c_1^p....c^p{}_{n-h+1}] \quad (2)$$

And to generate the feature map for matrix Z, we have:

$$c_z = [c_1^z, c_1^z....c^z{}_{n-h+1}] \quad (3)$$

However, it is worth to mention that, different semantic and POS features can be extracted using several filters.

3.3 Max Pooling Layer

Pooling operation is basically aimed at reducing the feature resolution maps by applying a pooling function to several units in a local region of a size based on a parameter known as pooling size. The pooling operation generally serves as generalizations over the features captured from the convolutional operation. Thus, the basic idea behind utilizing max poling layer is to extract the most salient features from the convolutional layer. Typically, pooling layer takes the maximum element in each generated feature map. This can be given as:

$\check{C}_p = max[c_1^p, c_1^p....c^p{}_{n-h+1}]$ and $\check{C}_z = max[c_1^z, c_1^z....c^p{}_{n-h+1}]$ for ***P*** and ***Z*** respectively.

When the max pooling is applied, the final maximum feature is generated by concatenating the semantic and POS features using a filter. This can be given as $C = \check{C}_p \oplus \check{C}_s$. Where $\oplus$ is the concatenation operator. As we use several features for the POS and semantic features, we have the final feature as:

$$C = \check{C}_p^1 \oplus .....\oplus \check{C}_p^n \oplus \check{C}_s^m \oplus .....\oplus \check{C}_s^m \quad (4)$$

Where ***n*** and ***m*** are the filters for semantic and POS features respectively.

3.4 Output layer

Here we finally apply the softmax classifier to generate the probability distribution over given aspects. The main idea of the softmax function is to carry out a classification process over the high-level features generated from the convolution operation and pooling layers. In this case, the softmax is used to find the probability distribution for all the output labels. Here, we specifically treat the aspect extraction as a sequence labeling process. Particularly we apply IOB scheme to indicate our aspect annotations as a tag sequence. Each word in the text is assigned with one of the 3 tags: I, O or B indicating beginning, Inside or Outside of an aspect term respectively.

4. Model Variations

In order to obtain robust and more reliable results, we conduct a series of experiments with several variants of the model.

- MCNN-Random: To assess the impacts of word embeddings, here the word embedding channel is randomly initialized while the input channel containing the POS Tag embeddings is ignored, meaning that only the randomized word embeddings channel is considered for training.
- MCNN+W2V: Here the word embedding layer is initialized with a pre-trained word2vec and optimized during training. Particularly, we used a general purpose word embeddings trained on the Google corpus [21].
- MCNN+W2V2: This is similar to the CNN+W2V variant.in this case, instead of using the general pre-trained word embedding, we use a domain-specific word2vec trained on the either Amazon or Yelp review datasets. This is specifically aimed to assess the impacts of the domain-specific word embeddings compared to the general word embeddings for the model performance.
- MCNN+W2V+POS: In this case, all the two input channels are considered for the training and optimization process. Specifically, we use the general word embeddings in one channel and POS Tag embeddings in the other channel. However, the model parameters are fine-tuned during optimization
- MCNN+W2V2+POS: This is similar to MCNN-W2V+POS variant, however, in this case, instead of applying a general pre-trained word2vec, a domain specific word embedding is used. All the parameters are fine-tuned.

5. Experimental Study

In this section, we first present a description of our datasets, we then provide a detailed experimental procedure for testing and evaluating the performance of our proposed approach. We also compare the performance of our approach against state-of-the-art models. We particularly apply the strategy to determine the best parameters for the optimum performance of our model. We use Recall, Precision and F1 score as the evaluation metrics to evaluate the model.

5.1 Dataset

We utilized four different benchmark datasets. The datasets typically comprise training and test snippets from two different domains, namely restaurant, and laptop domain. The datasets were collected manually and made available by the organizers for the SemEval competitions. The first two datasets are from SemEval2014 [38] which comprises reviews from laptop and restaurant domains respectively, while the third and fourth data sets are from semeval2015 [39] and SemEval2016 [40] respectively which contain reviews from restaurant domain. The statistics for all the datasets are shown in Table 1.

In order to initialize the word vectors, we particularly exploit two different word embeddings: General embeddings in which we use pre-trained Google word2vec trained on 100 billion words of google news corpus [21] using CBOW architecture, and domain-specific embeddings, trained on the restaurant review from the Yelp challenge and electronics reviews of the Amazon datasets for restaurant and the laptop domain respectively. The yelp and Amazon reviews datasets contain 2.2 million and 142.8 million reviews respectively. We set dimensionality of the word vectors to be 300 based on the empirical sensitivity studies. We use Gensim which has the implementation of CBOW to train all the datasets. Words that appear less than 5 times in the review are replaced with <other> token. To represent the unseen word. This token is used so as to provide a vector for each word. For the POS tag Embeddings, similar to [22] we used Stanford Tagger with of 45 tags in addition to the padding tag. We use one-hot encoding to encode these tags as a 45-dimensional vector.

Table 1: SemEval challenge datasets with the number of sentences and the aspect terms, L and R represent laptop and restaurant domain respectively.

Datasets	Train		Test	
	Sentence	Aspect	Sentence	Aspect
SemEval2014-L	3041	2358	800	654
SemEval2014-R	3045	3693	800	1134
SemEval2015-R	1315	1192	685	678
SemEval2016-R	2000	1743	676	622

5.2 Preprocessing

We carry out preprocessing with the aim of obtaining a clean and structured textual review. Specifically, we convert all the reviews into lower case comprised of only English texts and split the text into separate sentences. We apply noise removal strategy which includes removal of words with special characters, stop words, alphanumeric characters and words that have a length less than or equal to 1. The text is tokenized in which each word is categorized into tokens and extract its lemma and stems. We use the basic BIO coding scheme. Here "B" represents the beginning of the aspect term, "I" represent the inside of the aspect term and "O" indicate the outside of the aspect term.

5.3 Experimental Setup

We use 5-folds cross-validation strategy to choose the hyperparameters. Specifically, we choose three filter size of (3, 4, 5), each of which has two filters with 100 feature maps. We used a max pooling layer after each convolutional layer. As we wanted to tag each word, we use 1 as the stride for each convolutional layer. To tackle the issue of the parameter overfitting, we utilized drop out regularization on the penultimate layer with $L2$ constraints of 3. The training is conducted using stochastic gradient descendent over shuffled mini batches of size 64, maximum sentence length of 100 tokens and a dropout rate of 0.5. We apply ReLU for all the datasets and used 128 to be the size of the hidden rate. These values were chosen based on the careful grid search on the validation subset.

5.4 Baselines

To assess the proposed approach, we first make a comparison between the variants of our model (as described in section 3E) to identify the best performing variant and then make a further comparison against the state-of-the-art models. To this end, we used the following state of the art models as our baselines:

- DLIREC [41]: DLIREC is the wining system in the SemEval2014 (subtask 1) which employ a variety of lexical and semantic features derived from NLP source to improve the performance
- IHSR & D [42]: This is another top winning systems in the semeval2014 which typically exploit CRF and used additional features including lexical and statistical features.
- NLANGP [43] : NLANGP is the top system for restaurant semeval2016 challenge.
- ELIXA [44]: This is the top winning system in semeval2015, restaurant domain which used an average of perceptron with BIO tagging system for the aspect extraction task.
- WDEmb [16]: This model typically used dependency words integrated into CRF with path embedding for aspect term.
- BiLSTM-CNN-CRF [45]: This is an integrated deep learning based model with the CRF layer. It is the state-of-the-art aspect extraction approach from the Named Entity Recognition Community.
- RNCRF [32]: This model jointly uses CRF and a dependency-based recursive neural network for co-extracting aspects and opinion terms. The method also exploits additional handcrafted features.
- CMLA [46]: This is a multi-layer coupled-attention model for opinion and aspect terms co-extraction.
- MIN [34]: This is a multi-task learning approach that exploits lexicons and dependency rules to jointly perform co-extraction of aspect terms and opinion terms. It uses two different LSTMs and another LSTM for polarity classification of sentences.
- DTBCSNN [17]: This is a dependency tree based convolutional stacked neural network which used inference layer for the final output.
- DE-CNN [13]: This is a CNN based model exploiting double embeddings for aspect extraction.

- BiDTreeCRF [35]: This is a tree-based deep learning based approach which uses bidirectional LSTM and the CRF layer for improving aspect extraction.

Table 2: Results of comparison in terms of F1 scores with the state-of-the-art methods

Model	SemEval2014-L	SemEval2014-R	SemEval2015-R	SemEval2016-R
HIS_RD	74.55	79.62	-	-
NLANGP	-	-	67.12	72.34
DLIREC	73.78	84.01	-	-
ELIXA	-	-	72.05	-
WDEmb	75.16	84.97	69.73	-
RNCRF+F	78.42	84.93	-	-
CMLA	77.8	-	-	-
MIN	77.58	85.29	70.73	73.44
BidTreeCRF	80.57	84.83	70.83	74.49
DTBCSNN	75.66	83.97		
DE-CNN	**81.59**	-	-	74.37
MCNN+WV+POS	79.84	84.69	**72.84**	72.62
MCNN+WV2+POS	80.63	**86.89**	72.65	**75.71**

Table 3: Comparison results among the different variations of our model in terms of recall, precision, and F1 score performance.

Variant	SemEval2014-L			SemEval2014-R			SemEval2015-R			SemEval2016-R		
	R	P	F	R	P	F	R	P	F	R	P	F
MCNN+Rand	68.50	73.41	70.87	80.76	83.45	82.08	60.20	70.50	64.94	65.61	70.25	67.85
MCNN+WV	74.30	82.65	78.25	83.50	85.20	84.34	62.60	73.01	67.41	68.71	74.32	71.40
MCNN+WV2	75.85	86.61	80.87	85.71	86.14	85.92	65.54	75.87	70.33	70.56	74.54	72.50
MCNN+WV+POS	74.85	85.54	79.84	83.32	86.10	84.69	71.32	74.43	72.84	69.12	76.50	72.62
MCNN+WV2+POS	77.65	86.65	81.90	86.24	87.01	86.62	70.08	75.41	72.65	72.17	79.61	75.71

6. Results and Discussion

Table 2 shows the results of our approach in comparison to the state-of-the-art models. Here, the results of the best two settings of our approach are recorded for each dataset. It can be shown that the best performing variants of our approach significantly outperform the state of art approaches. The statistical t-test shows the improvement is significant at the confidence level of 95%.

Compared to the best-performing systems in the SemEval competition, our model performs better than HIS_RD and DLIREC with gains of 6.08 %, 7.27% and 6.85 %, 2.88 % F1 score on the semEval2014-L and SemEval2014-R datasets respectively. Similarly, our approach also achieves significant gains against ELIXA and NLANGP by 0.79%, 5.72% and 3.37% F1 score on the SemEval2005-R and SemEval2016-R respectively. Even compared to the WDemb approach which exploits word dependency with additional embedding, still, our model achieved significant gains against the WDemb approach across all the datasets. One can also notice from the Table 3 that, our model outperforms MIN which is a multitasking approach, with a gain of 3.05%, 1.6%, 2.11%, and 2.27% F1 score on the SemEval2014-L, SemEval2014-R, SemEval2015-R and SemEval2016-R respectively. Our model also outperforms CMLA which is a multilayer approach by 2.83% F1 score on the semeval2014-L datasets.

In spite of exploiting additional handcrafted features by RNCR+F and DTBCSNN still, our approach achieves 2.21 %, 1.96% and 4,97%, 2.92% F1 score gains over the two approaches on the semeval2014-L and semeval-2014-R respectively. Moreover, our model outperforms the recent tree-base bidirectional method, BidTreeCRF by 0.06%, 2.06 %, 2.01% and 1.22% F1 score on the semeval2014-L, semeval2014-R, semeval2015-R and semeval2016-R respectively. Compared to the double embedding CNN approach, DE-CCN which is the state of the art double embedding approach our model suffered a low performance on the semeval2014-L, however, it manages to achieve a gain of 1.34% F1 score on the semeval2016-R datasets which apparently shows the superior performance of our model over the DE-CNN model.

It can be observed from Table 3, that different variants of the model have different performance across the four different datasets. MCNN-WV2-POS performs better than all the other variants across all the datasets while the MCNN-random records relatively lowest performance except on the semeval2015-R where the MCNN-WV2-POS records the best results. This is likely due to the relatively smaller size of the semeval2015-R datasets. Similarly, one can notice from the Table 3, that in all the variants, the best results were recorded on the restaurant domain while relatively lower results are recorded on the laptop domain in all the datasets. This is likely due to the lower number of the aspects term contained in the restaurant domain than in the laptop review domain.

As can be seen from Table 3 and figure 2, all the variants of our model with the exception of MCNN-random demonstrate relatively competitive results with significant improvement across all the domains. This specifically indicates the weakness of the randomly initialized word embeddings for the aspect extraction. This is because MCNN-random is randomly initialized while the other variants are particularly initialized with general pre-trained word embeddings and domain-specific word embeddings. This translates the importance of pre-trained word embeddings over the randomly initialized word embeddings. It is also shown from the results that initializing word vector with the domain-specific word embedding for both laptop and restaurant domains perform better than the general word embeddings (Google embeddings) initialization. This support the intuition that domain-specific embeddings typically contain opinion specific information related to a particular domain (Laptop or restaurant) which helps to perform better than the general word embeddings which are merely trained on the google news corpus typically composed of textual reviews about the commonly discussed matters on the news.

One can observe from figure 3 that in both laptop and restaurant domain the model suffers from low recall, meaning that it missed some vital aspect terms. However, using POS tag which is an important linguistic feature help to overcome some drawbacks thereby improving the performance of the model. This specifically indicates the importance of using POS tags features in addition to pre-trained word embeddings in aspect term extraction.

We further conduct an experiment to assess the sensitivity of the model towards word embeddings dimensions. We specifically use different word embedding dimensions from 50 to 375 with the intervals of 25. i.e. {50, 75, 100, 125, 150, 175, 200, 225, 250, 275, 300, 325, 350, 375}. The laptop domain uses embeddings trained on the Amazon reviews and restaurant domain use the embeddings trained on the Yelp reviews datasets. Figure 4 shows the experimental results on the MCNN-WV2 variant. The results indicate the highest performance at around 300 dimensions and relatively remains stable above 150. This particularly implies insensitivity of the model towards the dimension of word embeddings provided it is within the appropriate range such as 100 to 375.

However, our model experienced two sources of error which include inconsistent labeling of the frequent words and the emergence of the unobserved aspects that require the extraction of the combination word such as “and “or “with”. For instance, if X and Y are two different aspect term and when X and Y appear, Y should also be extracted but not. The other error is the one that comes from the inconsistent labeling for instance.

It is clear that two key factors are basically the reasons behind the outperformance of our models over the state-of-the-art approaches. First, the POS embedding input layer which uses POS tags to help in detecting the aspect terms and the domain-specific pre-trained word embeddings which was trained on the target domain corpus of the review datasets. The advantage of our approach is that it is relatively uncomplicated and automatic that does not require any feature engineering. This saves time, cost and improve the high performance of the model.

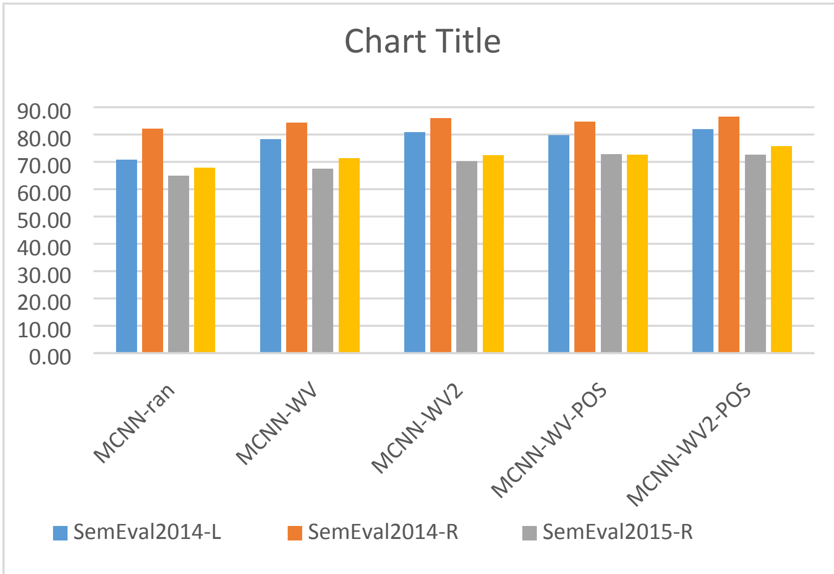

Figure 2: performance of the different model variants in term of **F1** score accuracy

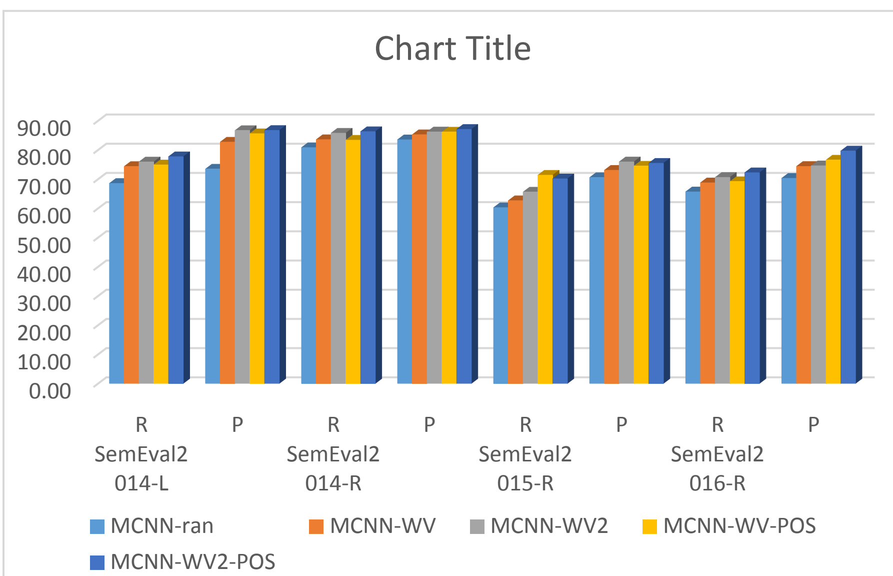

Figure 3: performance of the different model variants in term of recall and precision

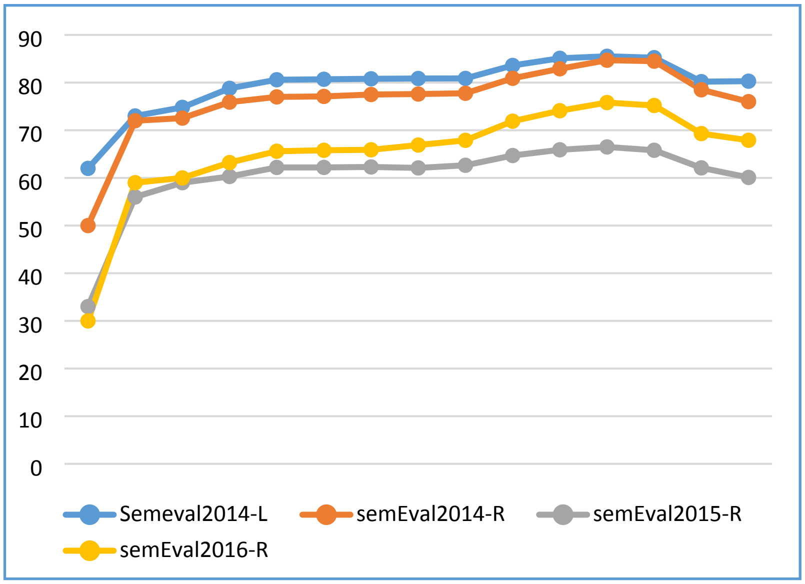

Figure 4: F1 score of the MCNN-WV2-POS on different word embeddings dimension.

7. Conclusion and Future Direction

In this research, we proposed an aspect extraction approach using a Deep multichannel convolutional neural network leveraging two different channels namely, word embeddings and POS tag embeddings. We presented a series of experiments and the results on various baseline models showed that our proposed approach outperformed the state-of-the-art methods. Our results support the well-known evidence that pre-trained word vectors is critically essential for better the deep learning-based aspect extraction and that the use of POS tag embeddings substantially improve the accuracy of aspect extraction performance. We also demonstrated the importance of using a domain specific word embedding for a CNN model on the corresponding domain review datasets. As a feature direction of the research, we think that, applying ensemble deep learning model for improving aspect extraction is worth to explore and that integrating lexicon in the word embedding layer in the multichannel CNN is also another direction to explore.

References

[1] B. Liu, *Sentiment Analysis and Opinion Mining*, no. May. Morgan & Claypool Publishers., 2012.

[2] I. Titov and R. McDonald, “A joint model of text and aspect ratings for sentiment summarization,” in *Proceedings of ACL08 HLT*, 2008, vol. 51, no. June, pp. 308–316.

[3] S. M. Street and B. Liu, “Extracting and Ranking Product Features in Opinion Documents,” *Program*, vol. 17, no. August, pp. 1462–1470, 2009.

[4] H. Wang and M. Ester, “A Sentiment-aligned Topic Model for Product Aspect Rating Prediction,” in *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2014, pp. 1192–1202.

[5] Q. Guang and L. Bing, “Bu Jia jun, et al. Opinion Word Expansion and Target Extraction through Double Propagation,” *Comput. Linguist.*, no. July 2010, 2009.

[6] B. Wang and H. Wang, “Bootstrapping Both Product Features and Opinion Words from Chinese Customer Reviews with Cross-Inducing.,” *Ijcnlp*, no. 60675035, pp. 289–295, 2008.

[7] F. Li *et al.*, “Structure-aware Review Mining and Summarization,” in *Proceedings of the 23rd International Conference on Computational Linguistics*, 2010, no. August, pp. 653–661.

[8] J. Lafferty and A. Mccallum, “Conditional Random Fields : Probabilistic Models for Segmenting and Labeling Sequence Data,” in *Proceedings of the 18th International Conference on Machine Learning 2001 (ICML 2001)*, 2001, vol. 2001, no. June, pp. 282–289.

[9] W. Jin and H. H. Ho, “A novel lexicalized HMM-based learning framework for web opinion mining,” in *Proceedings of the 26th Annual International Conference on Machine Learning - ICML ’09*, 2009, pp. 1–8.

[10] O. Irsoy and C. Cardie, “Opinion mining with deep recurrent neural networks,” *Proc. 2014 Conf. Empir. Methods Nat. Lang. Process.*, pp. 720–728, 2014.

[11] A. Popescu and O. Etzioni, “Extracting Product Features and Opinion from Reviews,” *Hum. Lang. Technol. Empir. Methods Nat. Lang. Process. Vancouver, Br. Columbia*, pp. 339–346, 2005.

[12] S. Poria, E. Cambria, and A. Gelbukh, “Aspect extraction for opinion mining with a deep convolutional neural network,” *Knowledge-Based Syst.*, vol. 108, pp. 42–49, Sep. 2016.

[13] H. Xu, B. Liu, L. Shu, and P. S. Yu, “Double Embeddings and CNN-based Sequence Labeling for Aspect Extraction Hu,” 2018.

[14] B. Wang and M. Liu, “Deep Learning for Aspect-Based Sentiment Analysis,” *CS224N Proj.*, pp. 1–9, 2015.

[15] D. Yang, C. Huang, and M. Wang, “A social recommender system by combining social network and sentiment similarity: A case study of healthcare,” *J. Inf. Sci.*, vol. 43, no. 5, pp. 635–648, 2017.

[16] Y. Yin, F. Wei, L. Dong, K. Xu, M. Zhang, and M. Zhou, “Unsupervised word and dependency path embeddings for aspect term extraction,” in *IJCAI International Joint Conference on Artificial Intelligence*, 2016, vol. 2016–Janua, pp. 2979–2985.

[17] H. Ye, Z. Yan, and Z. Luo, “Dependency-Tree Based Convolutional Neural Networks for Aspect Term Extraction,” no. April 2017, 2017.

[18] F. Dong, Y. Zhang, and J. Yang, “Attention-based Recurrent Convolutional Neural Network for Automatic Essay Scoring,” *Proc. 21st Conf. Comput. Nat. Lang. Learn. (CoNLL 2017)*, pp. 153–162, 2017.

[19] A. Blumer, A. Ehrenfeucht, D. Haussler, and M. K. Warmuth, “Occam’s Razor,” *Inf. Process. Lett.*, vol. 24, no. 6, pp. 377–380, 1987.

[20] M. Rouvier and B. Favre, “SENSEI-LIF at SemEval-2016 Task 4 : Polarity embedding fusion for robust sentiment analysis,” in *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016)*, 2016, pp. 207–213.

[21] T. Mikolov, W. Yih, and G. Zweig, “Linguistic regularities in continuous space word representations,” *Proc. NAACL-HLT*, no. June, pp. 746–751, 2013.

[22] S. Jebbara and P. Cimiano, “Aspect-Based Relational Sentiment Analysis Using a Stacked Neural Network Architecture,” *Artif. Intell. 29 August-2 …*, 2016.

[23] M. Hu and B. Liu, “Mining and summarizing customer reviews,” *Proc. 2004 ACM SIGKDD Int. Conf. Knowl. Discov. data Min. - KDD ’04*, p. 168, 2004.

[24] S. Poria, E. Cambria, L.-W. Ku, C. Gui, and A. Gelbukh, “A Rule-Based Approach to Aspect Extraction from Product Reviews,” in *second workshop on natural language processing for social media (SocialNLP)*, 2014, pp. 28–37.

[25] Z. Chen, A. Mukherjee, and B. Liu, “Aspect extraction with automated prior knowledge learning,” *52nd Annu. Meet. Assoc. Comput. Linguist. ACL 2014 - Proc. Conf.*, vol. 1, pp. 347–358, 2014.

[26] S. Zhou, Q. Chen, and X. Wang, “Active deep learning method for semi-supervised sentiment classification,” *Neurocomputing*, vol. 120, pp. 536–546, 2013.

[27] N. Commons, “A neural network model for semi-supervised review aspect identification A Neural Network Model for Semi-supervised,” pp. 668–680, 2017.

[28] L. Wang, K. Liu, Z. Cao, J. Zhao, and G. de Melo, “Sentiment-Aspect Extraction based on Restricted Boltzmann Machines,” in *Proceedings of the 53rd Annual Meeting of the Association for*

Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers), 2015, pp. 616–625.

[29] L. Pengfei, J. Shafiq, and M. Helen, "Fine- grained opinion mining with recurrent neural net- works and word embeddings," in . *In Proceedings ofthe 2015 Conference on Empirical Methods in Natural Language Processing,* 2015, pp. 1433–1443.

[30] P. Chen, Z. Sun, L. Bing, and W. Yang, "Recurrent Attention Network on Memory for Aspect Sentiment Analysis," in *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, 2017, pp. 452–461.

[31] R. He, W. S. Lee, H. T. Ng, and D. Dahlmeier, "An Unsupervised Neural Attention Model for Aspect Extraction," in *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 2017, pp. 388–397.

[32] W. Wang, S. J. Pan, D. Dahlmeier, and X. Xiao, "Recursive Neural Conditional Random Fields for Aspect-based Sentiment Analysis," *Proc. 2016 Conf. Empir. Methods Nat. Lang. Process.*, pp. 616–626, 2016.

[33] L. Xu, J. Lin, L. Wang, C. Yin, J. Wang, and J. W. Lamei Xu, Jin Lin, Lina Wang, Chinyong, "Deep Convolutional Neural Network based Approach for Aspect-based Sentiment Analysis," *Adv. Sci. Technol. Lett.*, vol. 143, no. Ast, pp. 199–204, 2017.

[34] X. Li, W. Lam, E. Management, and H. Kong, "Deep Multi-Task Learning for Aspect Term Extraction," pp. 2886–2892, 2017.

[35] H. Luo, T. Li, B. Liu, B. Wang, and H. Unger, "Improving Aspect Term Extraction with Bidirectional Dependency Tree Representation," 2018.

[36] H. Xu, S. Xie, L. Shu, and P. S. Yu, "Dual Attention Network for Product Compatibility and Function Satisfiability Analysis," 2017.

[37] Y. Kim, "Convolutional Neural Networks for Sentence Classification," in *proceedings of the 2014 Emperical Methods in Natural Language Processing (EMNLP)*, 2014, pp. 23–31.

[38] M. Pontiki and J. Pavlopoulos, "SemEval-2014 Task 4: Aspect Based Sentiment Analysis," in *Proceedings of the 8th International Workshop on Semantic Evaluation*, 2014, no. SemEval, pp. 27–35.

[39] M. Pontiki, D. Galanis, and H. Papageorgiou, "SemEval-2015 Task 12 : Aspect Based Sentiment Analysis," 2015, p. Semeval.

[40] M. Pontiki *et al.*, "SemEval-2016 Task 5 : Aspect Based Sentiment Analysis," in *SemEval*, 2016, pp. 19–30.

[41] Z. Toh and W. Wang, “DLIREC: Aspect Term Extraction and Term Polarity Classification System,” in *Proceedings of the 8th International Workshop on Semantic Evaluation (SemEval 2014)*, 2014, no. SemEval, pp. 235–240.

[42] M. Chernyshevich, “IHS R & D Belarus : Cross-domain Extraction of Product Features using Conditional Random Fields,” in *SemEval*, 2014, no. SemEval, pp. 309–313.

[43] Z. Toh and J. Su, “NLANGP at SemEval-2016 Task 5: Improving Aspect Based Sentiment Analysis using Neural Network Features,” *Proc. SemEval-2016*, vol. 2015, no. Subtask 1, pp. 282–288, 2016.

[44] R. Agerri, “EliXa : A modular and flexible ABSA platform,” in *SemEval*, 2015, pp. 748–752.

[45] N. Reimers and I. Gurevych, “Reporting Score Distributions Makes a Difference: Performance Study of LSTM-networks for Sequence Tagging,” pp. 338–348, 2017.

[46] W. Wang, S. J. Pan, and D. Dahlmeier, “Multi-task memory networks for category-specific aspect and opinion terms co-extraction,” *Aaai*, pp. 3316–3322, 2017.