Product Review Sentiment Analysis and Adapting to New Verticals

By Koushik Pati

WORD COUNT 4289

TIME SUBMITTED 19-JUL-2022 01:09PM

PAPER ID 88563828

Indian Statistical Institute

Master Thesis in Computer Science

Product Review Sentiment Analysis and Adapting to New Verticals

Master Candidate

Koushik Pati

Student ID CS2019

Supervisor

Prof. Utpal Garain

Indian Statistical Institute

Co-supervisor

Adhish Prasoon

Principal Data Scientist,Flipkart

Academic Year
2021/2022

Abstract

E-commerce company can measure product quality of a new product using sentiment extracted from reviews of a related old product.This can be done by *domain adaptation*,a sub field of machine learning .In domain adaptation there are two domain source domain and target domain. In this work we explore Unsupervised Domain Adaptation(UDA) where source domain has adequate labelled data but target domain has unlabelled data. In first part we consider sentiment analysis of product review within domain using convolutional neural network and language model BERT.Later we consider cross domain sentiment analysis that is domain adaptation of product review from source domain to target domain.We consider a state-of-the-art model Domain-Adversarial Neural Networks (DANN) for UDA with different architecture.We propose a BERT based Domain Adaptation(DA-BERT) that uses masked language model and fine tuning for UDA. DA-BERT achieves good accuracy for both source and target domain.We use three domain pairs of amazon review dataset.We conclude with some important observation and future work.

Contents

1 Introduction

In general in traditional machine learning we assume that training and test data follow same distribution in feature space.Therefore we expect good performance of a model on test data that trained on training data.In real world scenario data we have may not follow above assumption. This can happen due to different reason e.g., training and test data are collected from different source, training data changes over time. In this case, discrepancy in domain distribution leads to performance degradation.To overcome this domain gap ,domain adaptation is a sub area within machine learning came into picture . In domain adaptation training and test data are called source and target domain. Domain adaptation solves the domain disparity between source and target domain in such a way that model trained on source domain can be generalized to target domain .Domain adaptation is a sub field of another interesting concept in machine learning transfer learning.Both of these related technique aim to improve performance of a model with insufficient or inadequate labelled data in target domain from target domain with adequate labelled data.Transfer learning solves a class of problem where domain and/or task may change between two domain while in domain adaptation task remain same and domain may change.Domain adaptation can be supervised,semi-supervised and unsupervised.Unsupervised domain adaptation(UDA) refers to the situation where we have labelled source domain data but unlabelled target domain data.Unsupervised domain adaptation fits real world scenario better by learning only from unlabelled data which is available in both domain.

In domain adaptation, domain consists of three parts feature space X,label space Y, and corresponding probability distribution $P(x, y)$ that is $D = \{X, Y, P(x, y)\}$. Feature X is $k-$ dimensional subspace $X \in \mathbf{R}^k$,label space Y can be either binary class $\{0, 1\}$ or multi-class $\{1, 2, ..., C\}$ where C denote number of class and $p(x, y)$ is the probability distribution over feature-label pair. Probability distribution can be decomposed into marginal and conditional probability distribution as $p(x, y) = p(x)p(x|y)$ or $p(x, y) = p(y)p(y|x)$ .In traditional machine learning paradigm,if training instances $X = \{x_1, x_2, ...\}$ and labels $Y = \{y_1, y_2, ...\}$ are given, all machine learning model try to learn a function that generalizes well to unseen data.In case supervised set up we have $\{x_i, y_i\}_{i=1}^n$ and in unsupervised case we only have $\{x_i\}_{i=1}^n$ where n denote number of instances.In domain adaptation the goal is to learn a map from source D_s that generalize well to target D_t.

Different probability distribution is called domain gap and different label space is called category gap in domain adaptation.In some cases related domains may have some common labels and also some private labels.These type of domain adaptation is called open set domain adaptation.Traditional domain adaptation falls into closed set domain adaptation where target and source domains have same classes but there exists domain gap. Based on domain gap domain adaptation can be divided into three classes such as prior shift,covariate shift and concept shift [6].Concept shift refers to the situation where $p_s(x) = p_t(x)$ but $p_s(y|x) \neq p_t(y|x)$.prior shift refers to the situation where prior distribution of labels are different ,$p_s(y) \neq p_t(y)$ but posterior probability remain same $p_s(y|x) = p_t(y|x)$.This is solved under supervised domain adaptation as we need labelled data in both domain. In covariate shift the marginal distributions $p_s(x) \neq p_t(x)$ but conditional distributions remain same $p_s(y|x) = p_t(y|x)$.Main reason for covariate shift is sample selection bias and missing data.Most domain adaptation technique try to solve this type of domain gap.Approach to domain adaptation can be categorized into model-centric,data-centric and set hybrid approaches[13].Data centric approach concentrate on data in hand and use pseudo labelling,data selection .Hybrid approach is combination of model centric and data centric approach. Model centric methods redesign model architecture , loss function or regularization.In this project we concentrate on model centric approach.

2 Background

2.1 Motivation

Domain adaptation has very nice application in data-driven industries like retail and e-commerce .Consider a practical scenario,suppose a e-commerce launches a new product A recently related to an existing old product B.For product B they have enough consumer review and ratings.For product A there are adequate consumer review but no ratings.Although in current e-commerce echo-system consumer must have to give rating with review as well.They are looking for better product quality measurement technique.Using unsupervised domain adaptation considering reviews of product B as source domain and reviews of product A as target domain we can predict ratings for for product A.That is we can measure product quality through UDA.

2.2 Problem Statement

Keeping in mind the above mentioned scenario we formulated our problem as *Product Review Sentiment Analysis and Adapting to New Verticals*.First part of the problem will deal with building a robust sentiment analysis model from the e-commerce review data for a particular shopping vertical with abundant labeled data.In the second phase of the project we will deal with domain adaptation techniques to adapt/translate these models to new shopping verticals which don't have enough labeled data.

3 Related Work

3.1 Model-centric approaches to UDA :

3.1.1 Pivot based

There are two types of pivot based method structural correspondence learning (SCL)[3] and spectral feature alignment(SFA) [16] .Ziser and Reichart propose a method autoencoder structural correspondence learning (AE-SCL) that combine the pivot-based methods with autoencoder neural networks .To learn latent representations to map non-pivots to pivots autoencoders are used and these encodings are then used to augment the training data.Recently a hybrid approach to UDA *Pivot-based domain adaptation for pre-trained deep contextualized embedding models(PERL)*[2] uses pivots with contextual embedding.

3.1.2 Autoencoder based

Early neural UDA based on autoencoders.Autoencoders are neural network that learn latent representation in an unsupervised fashion by learning with an input reconstruction loss.SDA [9] learns a robust and unified feature representation for all domains by stacking multiple layers, and artificially corrupts the inputs with a Gaussian noise that the decoder needs to reconstruct.SDA fails to generalize to high-dimensional data.Marginalized stacked denoising autoencoder (MSDA) which improves limitation of SDA by marginalizes the noise .

3.1.3 Domain adversaries

Most popular method is domain-adversarial neural networks(DANN) [7].This is inspired by generative adversarial network(GAN)[10] minimize the discrepancies between training and synthetic data .The aim of domain-adversarial neural network(DANN) is to estimate an right predictor for the task while maximizing the confusion of an auxiliary domain classifier so that it differentiate source feature from target feature.Also to learn domain-invariant feature it uses a loss function via a gradient reversal layer that make sure source and target domain feature distribution are similar.DANNs applied in many NLP task such as sentiment analysis [7],language identification[12],natural language inference[14],POS tagging,parsing,trigger identification.If domain classifier accurately discriminate source and target distribution then DANNs suffer from vanishing gradient problem [15].Wasserstein distance [1] based approach have more stable training than GRL [4]. DANNs becomes broadly used UDA approach in NLP.

4

Sentiment Analysis

4.1 Within Domain

4.1.1 CNN Model

Although originally invented for computer vision task Convolutional Neural Network(CNN) model have been shown to be effective in Natural Language Processing(NLP) task and achieve good results.We started this experiment with a CNN based model.let $w_i \in \mathbf{R}^d$ is a d dimensional vector represents a word. We can consider a sentence which is padded accordingly of length n as concatenation of vectors as follows :

$$w_{1:n} = w_1 + w_2 + \ldots + w_n$$

where + represents concatenation.A new feature is produced by convolution operation . In convolution operation a filter $f \in \mathbf{R}^{l*d}$ is applied to a window of l words.Then a feature f_j is generated from a window of words $w_{j:j+l-1}$ by $f_j = g(f.w_{j:j+l-1} + b)$ where b is bias and g is a non linear function.A feature map $f' = [f_1, f_2, \ldots, f_n]$ is produced by applying the filter over all possible windows of words in the sentence $\{w_{1:l}, w_{2:l+1}, \ldots, w_{n-l+1:n}\}$ where $f' \in \mathbf{R}^{n-l+1}$.Then applying max-pooling over the feature map we take $\hat{f} = max\{f'\}$ as feature corresponding to this filter.This would capture most important feature. By the above process we get one feature from one filter.This model uses multiple filter to obtain multiple feature .These features are then pass into a fully-connected sigmoid layer whose output is probability distribution over labels.

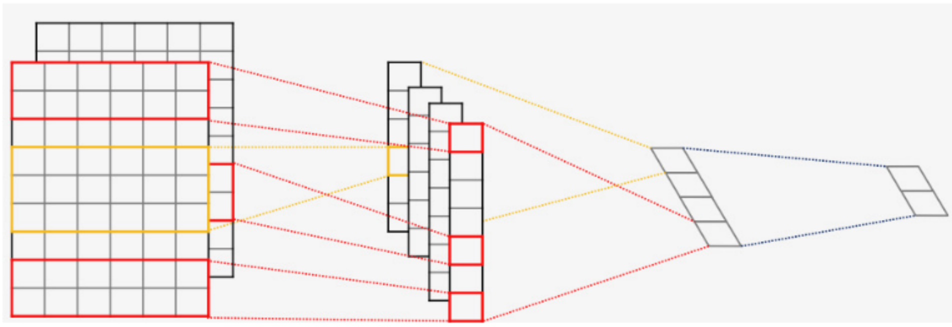

Figure 4.1: CNN MODEL

Experimental setup and Results

Model Architecture : Embedding layer with dimension 300,Convolutional layer with multiple filter width and feature maps(mentioned below) with max-pooling,Fully connected layer with dropout and sigmoid output.(figure 4.1)
Datasets : In this experiment we used Amazon review dataset for the product electronics,home and kitchen appliances ,CDs and Vinyl.All dataset are balanced. labels are binary: 0" if the overall rating is up to 3 , and 1" if the overall rating is 4 or 5 .Electronics contains 666242 reviews,home and kitchen has 192956 reviews ,Cds and vinyl has 389180 reviews.
Hyperparameters : binary cross-entropy as loss function,filter windows(l) of 3,4,5 with 100 feature maps each , learning rate 0.001,dropout probability 0.5, rectified linear unit as activation ,Adam as optimizer , batch size 50.
Word embedding : Here we used word2vec word embedding obtained by continuous bag-of-words architecture. The vectors have dimensionality 300 .Also the vectors are kept fixed during training.Also we take maximum sentence length as 300(99 percentile of review length).Padded by 0 if review length is less than 300 and discarded if review length is greater than 300.

Accuracy :

Data	Accuracy
electronics	0.85
cds and vinyl	0.84
home and kitchen	0.86

4.1.2 BERT Model

Bidirectional Encoder Representations from Transformers(BERT) [5] is a attention[17] based architecture has presented SOTA results in variety of NLP tasks,Natural Language Inference including Question Answering and many other.BERT extracts contextualised embedding from the corpus.BERT became popular and powerful language model because of two following reason :
1)It is pre-trained on unlabeled data from BooksCorpus and Wikipedia contains 800M words and 2,500M words respectively.
2)BERT learns information from a sequence of words from both left and right direction.
$BERT_{BASE}$ has 12 layers of transformer block,hidden size 768 , number of self attention heads is 12 and number of parameter 110M.BERT takes specific input representation.Maximum number of token in a sentence can be 512.The first token of every sentence is [CLS] token.Also there is another token [SEP] which separates two sentence.The input representation is formed by toke,segment and position embedding.Bert is pre-trained with two task masked language model and next sentence prediction.
We can use BERT in a downstream task by fine tuning pre-trained BERT.Classification task(sentiment analysis) are done by adding a classification layer on top of the Transformer output . Embedding of [CLS] token are then fed into this layer.Here we used pre-trained bert base model which has pooler layer which outputs embedding vector for [CLS] token. Then fine tuned pre-trained bert base model with labelled review dataset.

Experimental setup and Results

Architecture : BertModel ,dropout layer with dropout probability 0.5,fully connected layer with relu activation.
Datasets : same three review dataset used above.amazon review dataset for the products electronics,home and kitchen , and cds and vinyl.
Accuracy :

Data	Accuracy
electronics	0.88
cds and vinyl	0.88
home and kitchen	0.87

4.2 Cross Domain

In this section we analysed sentiment of product review from target domain using source domain labelled data.We described two approaches in detail one by one.

4.2.1 Model Trained with Source Only

Before going to domain adaptation we did an experiment over two model mentioned above.Earlier train and test are done within same domain.Now we train the model on source domain data and tested on target domain data. Results we got are given below :

Accuracy on target domain :

Source	Target	CNN	BERT
electronics	home and kitchen	0.81	0.86
electronics	cds and vinyl	0.70	0.87
cds and vinyl	home and kitchen	0.71	0.85

4.2.2 Domain Adaptation Using CNN(DACNN)

Let X denote input space and Y is binary label space.Also $d \in \{0, 1\}$ denote domain label , $d = 0$ for source domain, $d = 1$ for target domain.At training time for an input $x \in X$ we know $y \in Y$ corresponding to $d = 0$ but we do not know $y \in Y$ for $d = 1$.It is a deep Neural architecture that for given input x it predicts class label y and also domain label d.It consists three parts: feature extractor(F) ,label classifier(C),domain classifier(D).Feature extractor contains many CNN layer and parameters of F are denoted by θ_f.Feature extractor maps each input x to a feature vector $f \in \mathbf{R}^m$ that is $f = F(x, \theta_f)$.Then this feature vector is transformed by C to a label y and we denote parameter of this mapping by θ_c .Also f is transformed to domain label d by D and parameters of this mapping is denoted by θ_d.

At training time our objective is to minimize loss of C on source data and parameters of both F and C are optimized in such a way that source domain empirical loss is minimized.By this we get discriminative features f. Combining

F and C we get overall good prediction performance on the source domain.

Also we want domain invariant features f.In order to get domain-invariant features we look for parameters θ_f maximize the domain classifier to make two feature distribution as similar as possible,while simultaneously seeking the parameter θ_d that minimizes loss of D.Also we want to minimize the loss of C.To achieve domain invariant feature we inserted a gradient reversal layer(GRL) between feature extractor .This GRL has no parameter except the meta-parameter λ. During forward propagation GRL work as a identity function.During the back-propagation though, GRL takes the gradient from the successive layer, multiplies it by $-\lambda$ and passes it to the preceding layer.Mathematically gradient reversal layer can be defined as pseudo-function $R_\lambda(x)$ by two equation (one for forward propagation and one for back-propagation) :

$$R_\lambda(x) = x; \frac{dR_\lambda}{dx} = -\lambda * I \ .$$

Finally the objective pseudo-function E of $(\theta_f, \theta_c, \theta_d)$ which is optimized by stochastic gradient descent :

$$E(\theta_f, \theta_c, \theta_d) = \sum_{i=1,2,...,N d_i=0} L_c(C(F(x_i;\theta_f);\theta_c), y_i) + \sum_{i=1,2,...N} L_d(D(R_\lambda(F(x_i;\theta_f));\theta_d), y_i$$

where $L_c(.,.)$ and $L_d(.,.)$ are loss function of label classifier and domain classifier respectively,N is the number of training examples.

Experimental setup and results

Model Architecture : Embedding layer with dimension 300,Feature extractor F is Convolutional layer with multiple filter width and feature maps(mentioned in below) with max-pooling,both label classifier C and domain classifier are Fully connected linear layer(it has two layer,first layer with relu activation and second layer with sigmoid activation)

Datasets : Here also we used the same dataset as above that is amazon review dataset of product electronics, home and kitchen,cds and vinyl.

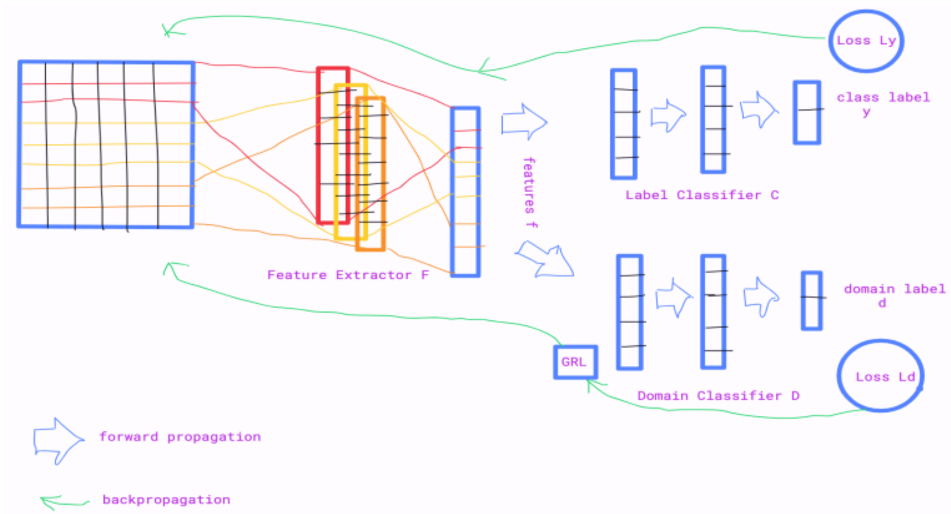

Figure 4.2: DACNN

Word embedding : In this experiment we used two different type of word embedding . One is word2vec word embedding obtained by continuous bag-of-words architecture.Another is obtained from Bert.In case of bert embedding we tested with last hidden layer embedding and also sum of last four hidden layer embedding (results are almost same in all cases).

Hyperparameters : Filter windows(l) of 3,4,5 with 100 feature maps each , dropout probability 0.5, rectified linear unit and sigmoid as activation ,binary cross-entropy as loss function ,Adam as optimizer , learning rate 0.001, batch size 50.

Accuracy on target domain :

Source	Target	DACNN(word2vec)	DACNN(bert)
electronics	home and kitchen	0.84	0.86
electronics	cds and vinyl	0.76	0.76
cds and vinyl	home and kitchen	0.78	0.79

5 BERT based Domain Adaptation

Pre-trained language model Bidirectional Encoder Representations from Transformers (BERT) have shown effective result in many language task.In case of unsupervised domain adaptation we can not simply fine tune bert because we have unlabelled data in target domain.Fine tuning only with source data lead to domain gap between training and testing data ,which degrade BERT performance.Bert has almost no understanding of opinion of text.Cross-domain brings challenge for BERT to distinguish source and target distribution.

To overcome these problems we proposed a BERT based Domain Adaptation(DA-BERT)(figure 5.1) which has two parts : **Re-training** and **Fine tuning**.

Re-training : Lack of labelled target domain data limits the direct fine tuning of BERT.Therefore we use BERT Re training .Intuition behind this model is that we are injecting some target domain knowledge in advance.In BERT Re-training we take BertForMaskedLM model(weights are initialized by pre-trained $BERT_{BASE}$) for unsupervised target domain Masked Language Model(MLM) .To make BERT domain aware we use MLM.In MLM it predicts randomly masked words in a sentence to learn deep bi-directional representation from the target domain text data.Though we have no label data but we have adequate unlabelled data in target domain.We replace 15% of the token in a sentence by the token $[MASK]$.Then last layer hidden vectors of all 300 token are fed into feed-forward NN which outputs a vector(logits) of length of vocabulary.Finally we get the prediction token id for masked token by softmax and argmax.Then MLM

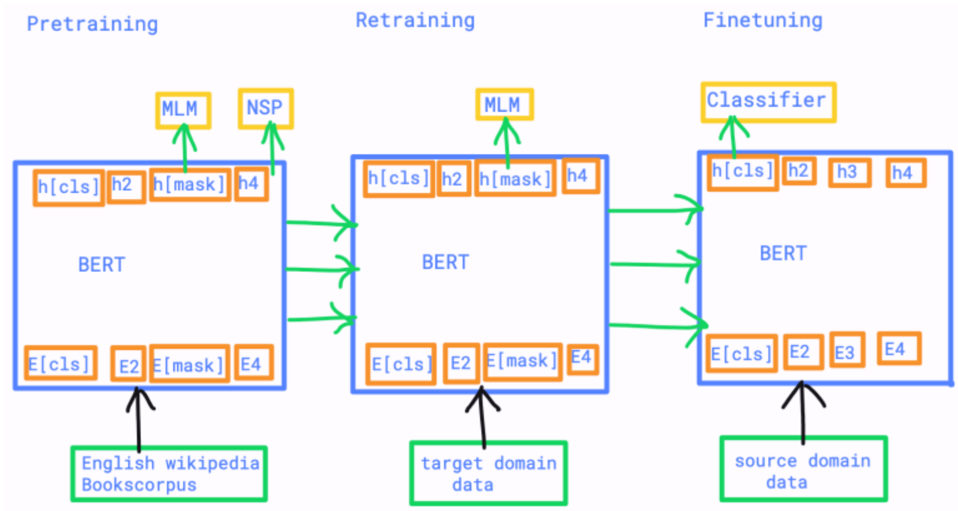

Figure 5.1: DA-BERT

loss is calculated.

Fine tuning : After Re-training we now fine tune BERT with source domain labelled data.A sentiment classifier is designed operating on the embedding vector of length 768 corresponding to $h_{[CLS]}$(first token of each sentence),that is classification embedding $[CLS]$.Embedding corresponding to this token from last layer is used as aggregate representation of the input sequence for classification task.The sentiment classifier is a fully-connected feed forward layer with sigmoid activation.
This experiment is done by amazon review dataset for product electronics,home and kitchen,cds and vinyl.We evaluate performance of this model for both source and target domain.

Results :

Source	Target	Accuracy on Source	Accuracy on Target
electronics	home and kitchen	0.90	0.91
electronics	cds and vinyl	0.89	0.89
cds and vinyl	home and kitchen	0.87	0.90

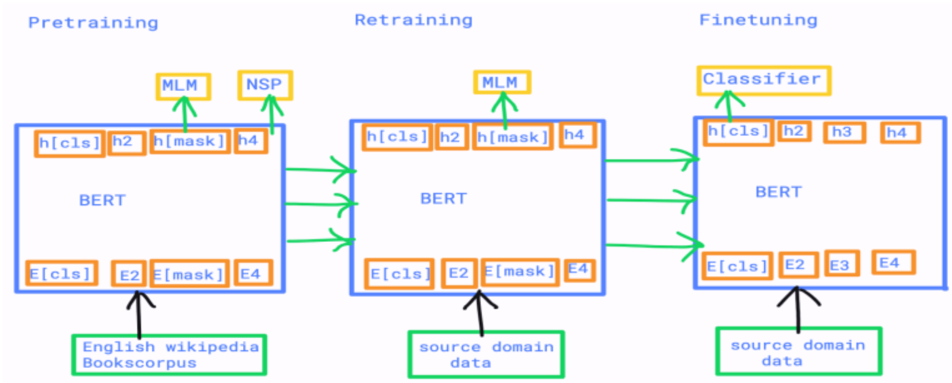

Figure 5.2: variant of DA-BERT

Also we have done one more experiment with a variant of DA-BERT(figure 5.2).In two stages of BERT-DA we used source data only. Results we got as follows :

Source	Target	Accuracy on Source	Accuracy on Target
electronics	home and kitchen	0.897	0.902
electronics	cds and vinyl	0.897	0.875
cds and vinyl	home and kitchen	0.87	0.89

From above two tables we see that DA-BERT(Re-train with target domain data) performs better than DA-BERT(Re-train with source domain data). So MLM with target data improve performance of BERT both source and target domain.

Finally we compare performance of our proposed model DA-BERT with two state-of-art methods Domain-Adversarial Neural Networks (DANN)[7] and Pivot-based domain adaptation for pre-trained deep contextualized embedding models(PERL)[2].

Accuracy on target domain :

Source	Target	DANN	PERL	DA-BERT
electronics	home and kitchen	0.854	0.906	0.91
electronics	cds and vinyl	0.738	0.85	0.89
cds and vinyl	home and kitchen	0.783	0.899	0.90

From the result we can see that DA-BERT outperforms other two state-of-the-art model .

6 Conclusions and Future Works

6.1 Error Analysis

- In this project we consider labels are binary: "0" if the overall rating is up to 3 , and "1" if the overall rating is 4 or 5 . Instead if we drop the reviews corresponding to rating 3 then performance of all model is better than previous.This is happening because reviews of rating 3 contains both positive and negative sentiments.For example a customer likes many feature of a electronics gadget but dislike some feature (and due to this customer gives rating 3).
- In data preprocessing part if we remove the negative stop words like don't,haven't,.. then the model takes negative reviews as positive . For example consider this review "i am not happy with the product ". Then removing 'not' it becomes "i am happy with the product".

6.2 Conclusion

In this project we explore sentiment analysis of product review and unsupervised domain adaptation through model centric approach.BERT is giving best result for sentiment analysis within domain.BERT based Domain Adaptation is giving best result for domain adaptation. In future we will look for more experiment with different approach to improve this result and also how these model perform in imbalanced dataset (because review dataset are always imbalanced).

References

[1] Martin Arjovsky, Soumith Chintala, and Leon Bottou. "Wasserstein Generative Adversarial Networks". In: 2018.

[2] Eyal Ben-David and Roi Reichart Carmel Rabinovitz. "PERL: Pivot-based Domain Adaptation for Pre-trained Deep Contextualized Embedding Models". In: 2020.

[3] John Blitzer, Ryan McDonald, and Fernando Pereira. "Domain adaptation with structural correspondence learning." In: In Proceedings of the 2006 Conference on Empirical Methods in Natural Language Processing. 2006, pp. 120–128.

[4] Tao Lei Darsh J Shah et al. "Adversarial Domain Adaptation for Duplicate Question Detection". In: 2018.

[5] Jacob Devlin et al. "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding". In: 2019.

[6] Abolfazl Farahani et al. "A BRIEF REVIEW OF DOMAIN ADAPTATION". In: 2020.

[7] Yaroslav Ganin and Hana Ajakan. "Domain-Adversarial Training of Neural Networks". In: 2016.

[8] Yaroslav Ganin and Victor Lempitsky. "Unsupervised Domain Adaptation by Backpropagation". In: 2014.

[9] Xavier Glorot, Antoine Bordes, and Yoshua Bengio. "Domain adaptation for large-scale sentiment classification: A deep learning approach". In: 2011.

[10] Ian Goodfellow. "Generative Adversarial Nets". In: 2014.

[11] Yoon Kim. "Convolutional Neural Networks for Sentence Classification". In: 2014.

[12] Yitong Li, Timothy Baldwin, and Trevor Cohn. "Whats in a Domain? Learning Domain-Robust Text Representations using Adversarial Training." In: In Proceedings of NAACL-HLT 2018. 2018, pp. 474–479.

[13] Alan Ramponi and Barbara Plank. "Neural Unsupervised Domain Adaptation in NLPA Survey". In: 2020.

[14] Gil Rocha and Henrique Lopes Cardoso. "A comparative analysis of unsupervised language adaptation methods". In: In Proceedings of the 2nd Workshop on Deep Learning Approaches for Low-Resource NLP (DeepLo 2019. 2019, pp. 11–21.

[15] Jian Shen et al. "Wasserstein Distance Guided Representation Learning for Domain Adaptation". In: 2018.

[16] Xiaochuan Ni Sinno Jialin Pan et al. "Cross-Domain Sentiment Classification via Spectral Feature Alignment". In: In Proceedings of the 19th international conference on World wide web. ACM. 2010, pp. 751–760.

[17] Ashish Vaswani, Noam Shazeer, and ukasz Kaiser. "Attention Is All You Need". In: 2017.

Acknowledgments

I am extremely grateful to my supervisors **prof. Utpal Garain** and **Adhish Prasoon** for their support and guidance in completing this project .Their valuable suggestion inspired me to do lot of experiments .

Also i would like to thank **Flipkart** for providing me required experimental set up .

Finally, i want to thank **Indian Statistical Institute** for allowing industrial collaboration with **Flipkart**.

Product Review Sentiment Analysis and Adapting to New Verticals

ORIGINALITY REPORT

4%
SIMILARITY INDEX

PRIMARY SOURCES

1	arxiv.org Internet	65 words — 2%
2	Michael Shell, John Doe. "Pose-invariant Feature Learning for Vehicle Re-identification", IEEE Internet of Things Journal, 2020 Crossref	28 words — 1%
3	export.arxiv.org Internet	24 words — 1%
4	Yao, Cuili, Yue Qu, Bo Jin, Li Guo, Chao Li, Wenjuan Cui, and Lin Feng. "A Convolutional Neural Network Model for Online Medical Guidance", IEEE Access, 2016. Crossref	17 words — < 1%
5	Debasmit Das, C.S. George Lee. "Unsupervised Domain Adaptation Using Regularized Hyper-Graph Matching", 2018 25th IEEE International Conference on Image Processing (ICIP), 2018 Crossref	15 words — < 1%
6	"Knowledge Science, Engineering and Management", Springer Science and Business Media LLC, 2018 Crossref	14 words — < 1%

7	boa.unimib.it Internet	14 words — < 1%
8	par.nsf.gov Internet	14 words — < 1%

EXCLUDE QUOTES	ON	EXCLUDE SOURCES	< 14 WORDS
EXCLUDE BIBLIOGRAPHY	ON	EXCLUDE MATCHES	< 14 WORDS