Breaking Language Barriers: A Study On Advancing Aspect-Based Sentiment Analysis for Low Resource Languages

Thesis submitted in partial fulfillment
of the requirements for the degree of

Master of Science in Computational Linguistics by Research

by

Arghya Bhattacharya
20161087
arghya.b@research.iiit.ac.in

International Institute of Information Technology
(Deemed to be University)
Hyderabad - 500 032, INDIA
July 2023

Copyright © Arghya Bhattacharya, 2023
All Rights Reserved

International Institute of Information Technology
Hyderabad, India

CERTIFICATE

It is certified that the work contained in this thesis, titled "Breaking Language Barriers: A Study On Advancing Aspect-Based Sentiment Analysis for Low Resource Languages" by Arghya Bhattacharya, has been carried out under my supervision and is not submitted elsewhere for a degree.

_____ _____

Date Adviser: Prof. Manish Shrivastava

To *Baba* and *Ma*

Acknowledgments

I would like to express my deepest gratitude to the following individuals for their invaluable support and contributions to my research journey:

- First and foremost, I am indebted to my parents for their unwavering love and care throughout my life. Their support has been instrumental in enabling me to pursue my academic goals and conduct research.
- I would also like to thank my siblings, Apala and Leo, for their unconditional love and constant encouragement. Their support and belief in me have been a source of strength and motivation throughout my academic journey.
- I am grateful to my Advisor, Prof Manish Shrivastava, for his guidance, expertise, and unwavering support. His mentorship has been invaluable in shaping my research interests and developing my skills as a researcher.
- I would also like to express my gratitude to Allen Jojo Antony, for being a senior, mentor, and the person who introduced me to the fascinating world of AI. His support and guidance have been crucial in shaping my research interests and helping me navigate the academic landscape.
- I would also like to thank Alok Debnath, for being the most supportive and collaborative reserch peer i've had. His contributions to this work cannot be put into words.
- I am also thankful to Arnav Sharma, for his invaluable mentorship over the last two years. His constant push to new limits and unwavering belief in my potential have been pivotal in my growth as a researcher.
- Finally, I would like to acknowledge the contributions of Bhavathi Reddy, who motivated me during the most challenging period of my research journey. Her own academic excellence and unwavering support, even in the face of adversity, have been a source of inspiration.

Thank you all for your invaluable support, encouragement, and guidance. Your contributions have been instrumental in making my research journey a meaningful and rewarding experience.

Abstract

In recent years, due to the advent of technology and the Internet, the amount of opinionated data targeting products has increased exponentially. With this increase, there has emerged a need to understand the opinions and their associated sentiment to enhance the feedback loop for organizations manufacturing the products and users looking for opinions about the usability of the same to base their future purchasing decisions on.

The major bottleneck in achieving the above is the constraint on resources available for certain languages. We explore ways to work through these constraints and attempt to achieve good results for the tasks of Sentiment Analysis and it's variants.

In this thesis, We first attempt to extract language invariant features for downstream tasks like sentiment analysis to be able to retain decent performance in a low resources setting. We find that the ability to do so is task-dependent and hypothesize patterns in tasks where the approach can and cannot work effectively.

We then take a closer look at the reasons for the poor performance of models in Aspect Term Extraction and Aspect Term Polarity Classification for Hindi, which is a variant of Sentiment Analysis based on Opinion Mining. After a detailed analysis of the same, we conclude that there is a gap in the state of the existing gold standard dataset for Hindi. We then go ahead and describe our methodology for developing a high-quality dataset parallel to the Gold English dataset for these tasks and establish that the new dataset adequately represents the task. To further improve the state of Aspect Term Extraction (ATE) and Aspect Term Polarity Classification (ATPC), we develop a novel architecture that achieves new state-of-the-art results for Hindi and near state-of-the-art results for English. We also show the fullness of our method in solving the task in a multilingual setting and achieving near-state-of-art results and hence establishing that for tasks where we cannot extract language invariant features, we can develop models that can learn features crucial for the task in a manner which can be leveraged to give high performance reliably.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

"Big data isn't about bits, it's about talent."

— Douglas Merrill

Aspect Based Sentiment Analysis is a task of Sentiment Analysis that deals with extracting sentiment from a given text with respect to a particular aspect. For example, in a review of a restaurant, the sentiment with respect to the "food" aspect would be different from the sentiment with respect to the "ambience" aspect. It is used to analyze the sentiment of a text with respect to a given aspect. Technically, it allows for a more fine-grained analysis of the sentiment of a text, which can be useful for many applications such as opinion mining and customer sentiment analysis.

But why is this important? Why solve ABSA for Indian Languages? Why study sentiment analysis? In this chapter, I try to explain the motivation for my study, some of the core pillars that motivate choice of the setting of this study and the relevance of the problem we solve and explain the organizational structure of the rest of this thesis.

1.1 Motivation

1.1.1 Aspect Based Sentiment Analysis as a challenging fine-grained task of Sentiment analysis.

Because of the globalization of the Internet, the amount of web-generated material is growing at an alarming rate. The enormous quantity of data has presented a number of new problems and possibilities to the scientific community. Customers or users nowadays depend significantly on other users' opinions on a product or service before deciding whether or not to purchase or use it themselves. In order to get an impartial judgment, one must first extract and read all of the evaluations, which is a time-consuming and difficult job. According to Pang and Lee (2008), sentiment analysis is the issue of automatically detecting the polarity of the sentiment/opinion conveyed by a user in a piece of text or review by a computer. Polarity is classified into four categories: positive, negative, neutral, and conflict polarity in

a review. Aspect-based sentiment analysis (ABSA) is a fine-grained study of sentiments at the aspect, feature, or attribute level that is performed on text. When we talk about an aspect, we're talking about a characteristic or a component of a product or service that has been discussed in a review. Generally speaking, the issue of aspect-based sentiment analysis may be viewed as a two-step procedure. Part of the process involves two steps: the first is aspect word extraction, which is concerned with finding different phrases that indicate aspects, and the second is sentiment classification, which is concerned with categorizing the feelings in relation to the aspect.

According to recent research, a rise in the amount of effort being done in fine-graining downstream NLP jobs has been seen. The utilization of aspect information is a popular technique for doing fine-grained analysis. When it comes to aspect terms, they are entities of interest that each identify a certain aspect of a specified subject or area of interest (Pontiki et al., 2014). For example, in the restaurant industry, service and seasoning are both important considerations. In the past, aspect extraction (AE) was often thought of as a subtask of fine-grained aspect-based sentiment analysis (ABSA), but recent advances in the literature have established it as a separate task that can be used in a variety of downstream tasks, including summary generation (Frermann and Klementiev, 2019) and topic-specific information retrieval, such as opinion mining (Asghar et al., 2019). There have been many aspect extraction datasets and models created for a wide range of languages (as a subtask of aspect-based sentiment analysis). Previously, ABSA was a shared task in SemEval 2014 (Pontiki et al., 2014), 2015 (Nakov et al., 2015), 2016 (Pontiki et al., 2016), and 2017 (Pontiki et al., 2017). In SemEval 2017, ABSA was included as a subtask of the overall task of sentiment analysis on Twitter (Rosenthal et al., 2017). All of these tasks have received a great deal of attention in a number of different languages such as Arabic, Chinese; Dutch; French; Russian; Spanish, and Turkish. For each monolingual dataset, there were one or two domains, with each language containing between 4,000 and 9,000 phrases total (depending on the language) (including the train and test split). There has been some effort in creating a dataset for aspect extraction in Hindi (Akhtar et al., 2016) and Telugu (Akhtar et al., 2016) for Indian languages (Regatte et al., 2020). In Hindi, there has only been a small amount of effort done to improve the status of AE and ABSA beyond the creation of a single dataset, which was developed by Akhtar et al. (2016). When compared to English AE as well as comparable sequence tagging tasks in Hindi, current sequence tagging models (both general and specialized to AE) have done very badly on this dataset, according to available evaluations. 4.3

1.1.2 The resource constrained setting of Indian Languages

Hindi has more than 500 million native speakers who are dispersed across the globe, and the number of web pages that provide information in Hindi is growing. These online sites are a valuable source of vital information for the public. Individuals, governments, and corporations may all benefit from this information by mining it for valuable insights.

The introduction of deep learning methods has made it possible to do high-quality textual analysis. One important aspect in the success of these neoteric methods that has been ignored is their reliance on

huge annotated datasets collected from various data sources that are linked to or derived from newspapers, tweets, photographs, and product evaluations, among other sources.

Because most languages have a scarcity of annotated data, developing deep learning-based solutions for them may be a difficult job to do. As a result, there is an urgent need to devote particular focus to creating solutions that are capable of operating effectively in low-resource environments.

1.2 Research Questions

Driven by the main motivations described in section 1.1, the work presented in this thesis attempts to answer the following research questions :

1. Given a low resource setting, is it possible to solve sentiment analysis like problems using deep learning techniques?
2. Is the current state of resources for Aspect Based Sentiment Analysis for Indian Languages adequate to assist research improvements in the same?
3. What kind of information is required to be captured by deep Learning models in order to effectively learn features to solve the problem of Aspect Based Sentiment Analysis?

1.3 Main Contributions

To summarize, the main contributions of this thesis are presented below :

1. Exploration followed by a solution for Sentiment Analysis in Low resource settings, the Language Invariant Sentiment Analyzer (LISA) architecture out-performs the previous state-of-the-art approaches on all the languages in the Multilingual Amazon Text Classification dataset. The approach also achieved significant performance gains over prior research on the low-resource Telugu Sentiraama corpus.(Gangula and Mamidi, 2018)
2. An In-depth qualitative and quantitative analysis of the existing Hindi dataset for Aspect Based Sentiment Analysis
3. A new resource for aspect extraction in Hindi by manually translating the SemEval 2014 corpus into Hindi.
4. Detailed guidelines and challenges associated with the creation of this corpus, as well as explaining the quality of the translations and annotations.
5. Benchmarking the new dataset using state-of-the-art neural sequence labeling models for aspect extraction in Hindi in monolingual and multilingual settings.
6. Context-Localization-and-Pooling (CLaP) architecture, a BERT-based, end-to-end model for aspect term extraction and polarity classification (ATEPC) that out-performs the existing state-of-the-art aspect term extraction and polarity classification. We perform model ablations to show the importance of accounting for aspect-level and sentence-level information.

1.4 Thesis Organization

The work presented in this thesis is organized as follows :

- **Chapter 1** gives a brief introduction on the problem of low resources settings and aspect based sentiment analysis. We address the primary motivation for our work
- **Chapter 2** provides a brief overview of the deep learning architectures we leverage for our experiments and provides a primer on the work that has already been done in the field of Aspect Based Sentiment Analysis.
- **Chapter 3** describes our efforts to solve a simpler problem of Sentiment Analysis in a Low Resource Setting in the absence of Parallel corpora, in order to understand the task complexity in relationship to the type of data available, i.e Parallel Corpora vs A very small labelled dataset.
- **Chapter 4** describes our efforts to analyse the existing state of Aspect Extraction and Polarity Classification and develop a new parallel corpus to facilitate research in the same. We establish our dataset as the gold standard for Hindi Aspect Based Sentiment Analysis.
- **Chapter 5** presents CLaP, a novel architecture to efficiently learn features that are required to solve the problem of Aspect Extraction and Aspect Polarity Classification.
- **Chapter 6** addresses the advantages and shortcomings of the proposed approach and state our concluding remarks and discussions on future work in this direction.

Chapter 2

Background and Related Works

"If I have seen further than others, it is by standing upon the shoulders of giants."

— Sir Isaac Newton

In this chapter, I provide a comprehensive overview of the various components that are required for a deeper understanding of the solutions proposed in this thesis. Throughout this chapter, we will provide a high level understanding of the tasks that have been studied in detail in this thesis and of the neural net designs that are used in the popular approaches to solving the tasks. We will start by looking into the tools required tasks of performing sentiment analysis in a low resource setting, specifically Cross Lingual transfer learning, after that we take a deep dive into looking at the some of the theory required to understand the research we've conducted on Aspect Based sentiment analysis as a task and the nuances of it. Following that, we will look into how the modern transformer based architecture can be leveraged for better learning of language features as a primer to understanding the methods proposed in chapters 4 and 5, Finally we look at some related work that has been conducted previously for ABSA.

2.1 Cross Lingual Transfer Learning in NLP

Transfer learning has become one of the most popular approaches to tackle the problem of learning a language task in a resource-constrained setting. A more mathematically rigorous definition was provided by Pan et al, 2009. in their publication titled "A Survey on Transfer Learning"

1. A domain D which is defined as a set consisting of a feature space $\mathcal{X}$ and a marginal probability distribution $P(X)$ where,
    - $D = \{\mathcal{X}, P(X)\}$
    - $X = \{x_1, \ldots, x_n\}$
    - where $x_i \in \mathcal{X}$ is a sample data point.

2. A task T which is defined as a set consisting of a class label space $\mathcal{Y}$ and a conditional probability distribution $P(Y|X)$ where,

   - $Y = \{y_1, \ldots, y_n\}$
   - $y_i \in \mathcal{Y}$ where y_i is a class label.

3. Given a source domain D_S, a source task T_S, a target domain D_T and target task T_T where $D_S \neq D_T$ or $T_S \neq T_T$, transfer learning is defined as a function that learns the conditional probability distribution $P(Y_T|X_T)$ in the target domain D_T by leveraging the information that was learnt from solving T_S on D_S.

Cross-lingual transfer learning (CLTL) is a technique that involves transferring resources, labels, or trained models from a language with many resources (the source language) to a language with fewer resources (the target language). CLTL is a type of transductive transfer learning, which means that the domains being transferred between are languages. The goal of CLTL is to use concepts that are shared across languages to improve natural language processing (NLP) for low-resource languages. CLTL can be used in two different settings: zero-shot learning or one-shot learning.

One-shot learning, also known as few-shot learning, is a technique that is used when there is a limited amount of training data available in the low-resource target language. While the amount of labeled data in the low-resource language may not be sufficient to train a supervised model from scratch, it may be enough to fine-tune a model that was trained on a resource-rich language. One-shot learning is useful in real-world scenarios where there is not enough labeled data for every class, or where new classes are frequently added. One-shot learning was first introduced by Fei-Fei et al, 2006. in their paper "One Shot Learning of Object Categories," in which they used Bayesian approaches for object classification. Zero-shot learning, also known as zero-data learning, is a more extreme version of one-shot learning in which there is no labeled data available in the low-resource language.

2.2 Convolutional Neural Networks

Convolutional neural networks differ from other neural networks in that they perform better with picture, voice, and audio signal inputs than they do with other signal inputs. They are divided into three kinds of strata, which are as follows:

- Convolution Layer
- Pooling Layer
- Fully-connected layer

The convolutional layer is the initial layer of a convolutional network, and it is responsible for learning the parameters of the network. However, although convolutional layers may be followed by further convolutional layers or pooling layers, the fully-connected layer is always the last layer to be applied. The complexity of the CNN rises with each layer that is added.

2.2.0.1 Convolution

The convolutional layer is the fundamental building component of a CNN, and it is also where the vast bulk of the computation takes place. Only a few components are required: input data, a filter, and a feature map, amongst other things.

2.2.0.2 Pooling Layer

Input dimensionality is reduced via the use of pooling layers, also known as downsampling, which reduces the number of parameters in the input. The pooling operation works in a similar way to the convolutional layer in that it sweeps a filter over the whole input, with the exception that this filter does not contain any weights. Instead, the kernel applies an aggregation function on the data contained within the receptive field, resulting in the values within the output array being populated. Pooling may be divided into two categories:

- Max Pooling
- Average Pooling

2.2.0.3 Fully Connected Layer

The name of the fully linked layer is a good description of what it is. As previously stated, the pixel values of the input picture are not directly linked to the output layer in partly connected layers because the input image is not directly connected to the output layer. When compared to this, each node in the output layer links directly to a node in the preceding layer when compared to the fully-connected layer.

Using the characteristics collected from the preceding layers and their various filters, this layer conducts the job of categorization on the data. Contrary to convolutional and pooling layers, which often use ReLu functions to categorise inputs correctly, FC layers typically employ a softmax activation function to do so, resulting in a probability ranging from 0 to 1.

2.3 Transformer based Language Modelling

By paying attention to certain portions of the input and not others, the model may generate predictions by looking at the whole input (rather than just the most recent segment) and selectively attending to some of it. This decision is made based on a set of weights that are learnt throughout the training phase of the process. For instance, the fox saw a bunny. It was very hungry, so it attempted to grab it, but it was able to avoid it just in time. The attention mechanism may be utilised to determine which word each ”it” in the input sequence refers to by analysing the input sequence.

The attention mechanism is used by transformers, which are models with an encoder-decoder structure and a decoder structure. Encoding is performed by the encoder component, which use the attention

mechanism to selectively attend to various portions of the input data. Encodings are then sent to the decoder component, where they are decoded.

2.3.1 BERT

Google AI introduced an encoder-based language model which is trained in both directions.The inputs of the BERT model are encoded in a specific way consisting of three parts as illustrated:

Figure 2.1 The input embeddings in BERT are made up of three separate embeddings

- **WordPiece tokenization embeddings**: Tokenization, as the name implies, is a technique for breaking up large words into smaller parts and converting them into vectors. For example, the word playing will be divided into two parts: play and "## ing", and then each of these two parts will be transformed into a 768-dimensional vector using the algorithm.
- **Segment embeddings**: Most often, this is used when the input consists of a pair of sentences, in which case it may be used to distinguish which tokens belong to the first sentence and which tokens belong to the second sentence. 0 represents all of the tokens that correspond to the first sentence, whereas 1 represents the remainder of the tokens. Following that, each 0/1 value is transferred to a 768-dimensional vector.
- **Position embeddings**: BERT supports a 512-context window, which corresponds to the values 0–511. Each of these locations is represented by a 768-dimensional vector, which is updated continuously during the training process. It should be noted that a specific [CLS] token is added to the beginning of all sequences at distinguish them. It is common to use this token for classification problems since it may be regarded as a representation of the whole input sequence. A unique separator token, [SEP], is also added at the end of each sentence to distinguish it from the others.

The three embeddings described above are combined together (elementwise), giving the model with a 768-dimensional input vector that is used to pretrain the model before being used to train the model. We will now examine the two pretraining paradigms

2.3.1.1 Masked Language Model

This task is required in order to accommodate the model's bidirectionality. It is not possible to train both left-to-right and right-to-left conditional language models using the conventional conditional language models, in which the target word is predicted from the previous or next word. This is due to the fact that with bidirectional conditioning, the term perceives itself indirectly. In order to produce the bidirectional character of the BERT language model, 15 percent of the training tokens are masked using an unique [MASK] token created specifically for this purpose. The model has been trained to anticipate the appearance of these masked tokens.

2.3.1.2 Next Sentence Prediction

A binary classification task is utilised as the second pre-training assignment in order to ensure that BERT performs well on different downstream tasks that rely on comprehending the connection between phrases. This phase takes into consideration the labelled sentence pairs (A and B). In half of the pairings, IsNext indicates that sentence B follows sentence A, while NotNext indicates that the phrases do not belong to each other in the remainder of the pairs.

2.3.2 M-BERT

A multilingual embedding model is a useful tool that encodes text from different languages into a shared embedding space, which can be used for various tasks like text classification and clustering. Existing methods for creating such embeddings, like LASER or mUSE, rely on parallel data and translation of phrases from one language to another to ensure consistency across sentence embeddings. However, these methods may not perform as well on high-resource languages as bilingual models that use translation ranking tasks with translation pairs as training data. It may also be challenging to expand multilingual models to cover more languages while maintaining acceptable performance due to limited model capacity and poor-quality training data for low-resource languages.

Multilingual BERT is a model trained on the Wikipedia dataset with 104 languages, similar to BERT, and capable of producing representations that can be applied across languages. This is conceivable because of a variety of language factors, including as

- Language-independent named entity recognition — names and nouns are the same across languages.
- Word piece overlaps - the same words occur in several languages at the same time.
- Structural similarity, The sequence in which the subject, verb, and object appear in sentences

In particular, the underrepresentation of low resource languages — languages that may not have enough parallel datasets for the model to capture contextual knowledge on par with English – is a significant disadvantage of BERT.

2.4 Aspect Based Sentiment Analysis

Aspect-Based Sentiment Analysis (ABSA) is a natural language processing (NLP) task that involves identifying the sentiment polarity of opinions expressed towards specific aspects of a product, service, or entity mentioned in the text. ABSA is an important task in NLP, as it helps in understanding the customer's opinion about specific aspects of a product or service, which can be used by companies to improve their products and services.

Several studies have been conducted on ABSA in both English and Hindi languages.

2.4.1 ABSA in English

In ABSA for English, the most commonly used approach involves identifying the aspects or entities mentioned in the text, followed by sentiment classification for each aspect. This can be achieved through a variety of methods, including rule-based techniques, machine learning algorithms, and deep learning models.

There have been numerous academic studies conducted in the field of ABSA, particularly in the English language. Here are a few notable examples:

- "Aspect-Based Sentiment Analysis of Movie Reviews on Discussion Boards" (2009) by Janyce Wiebe, Theresa Wilson, and Claire Cardie. This study focused on analyzing sentiment in movie reviews posted on discussion boards. The authors developed a system to identify aspects mentioned in the reviews and then assign sentiment scores to each aspect. The study found that aspect-based analysis was more accurate than document-level analysis.
- "Deep Learning for Aspect-Level Sentiment Classification: Survey, Vision, and Challenges" (2018) by Xiaodong Liu, Yeging Li, and Wei Lu. This study surveyed the state-of-the-art techniques for aspect-based sentiment analysis using deep learning methods. The authors discussed the various challenges associated with aspect-level sentiment classification, such as the need for large amounts of labeled data and the difficulty of handling complex sentence structures.
- "Adaptive Recursive Neural Network for Target-dependent Twitter Sentiment Classification" (2016) by Duyu Tang, Bing Qin, and Ting Liu. This study proposed a recursive neural network model for aspect-based sentiment analysis of tweets. The model uses a tree structure to represent the dependency relationship between words and their corresponding aspects. The authors showed that their model outperformed several state-of-the-art approaches on a Twitter sentiment analysis dataset.
- "Aspect-Based Sentiment Analysis with Gated Convolutional Networks" (2018) by Baolin Peng, Xiaodong Li, and Kam-Fai Wong. This study proposed a gated convolutional neural network (GCN) model for aspect-based sentiment analysis. The GCN model uses convolutional layers to capture local contextual information and gating mechanisms to control the flow of information. The authors showed that their model achieved state-of-the-art performance on several benchmark datasets.

These studies are just a few examples of the academic work that has been done in the field of aspect-based sentiment analysis in English. As the field continues to develop, we can expect to see even more innovative techniques and approaches for analyzing sentiment towards specific aspects of entities.

The popular approaches use deep learning models to perform aspect extraction and sentiment analysis simultaneously. These models typically utilize attention mechanisms and recurrent neural networks (RNNs) to capture the dependencies between words and aspects in the text. For example, the attention-based model proposed by Wang et al. (2016) uses a recursive neural network (RNN) with attention to perform ABSA. The model dynamically selects the most relevant words and aspects based on their importance to the sentiment of the text.

Another popular approach for ABSA in English is to use pre-trained language models, such as BERT (Bidirectional Encoder Representations from Transformers). These models have shown significant improvements in various NLP tasks, including ABSA. The pre-trained language models are fine-tuned on ABSA-specific datasets, which results in highly accurate models for sentiment analysis.

The most common architecture used for attention-based ABSA models is the multilayered bidirectional Long Short-Term Memory (BiLSTM) network with an attention mechanism applied on top of it. This architecture is used to encode the input text and produce a fixed-length representation of the text that captures the most relevant information for the task.

The attention mechanism allows the model to assign different weights to different parts of the input text, based on the importance of the information for the task. The attention weights are computed based on the similarity between the hidden state of the BiLSTM network and a context vector representing the aspect being discussed in the text. The attention weights are then used to weight the hidden states of the BiLSTM network and produce a weighted sum of these hidden states, which is used as the final representation of the input text.

Once the input text has been encoded and the attention weights have been computed, the sentiment of the aspect being discussed can be predicted using a classifier that takes as input the final representation of the input text. Recent research has also explored the use of pre-trained language models such as BERT and RoBERTa for ABSA, which have shown to achieve even better results than traditional attention-based models by leveraging the large amounts of pre-training data and the ability to model the context of the input text more effectively. These models typically fine-tune the pre-trained language model on a smaller ABSA task-specific dataset to adapt.

Overall, ABSA for English involves a variety of techniques and models, with deep learning and pre-trained language models showing promising results. However, for low-resource languages like Hindi, there is still a need for research and development to achieve similar levels of accuracy and interpretability in ABSA.

2.4.2 ABSA in Hindi

Research on ABSA for Hindi is limited, but some studies have proposed approaches that utilize both rule-based and machine learning-based methods. For example, Gupta et al (2020) proposed a

hybrid approach that combines rule-based and machine learning-based methods for Hindi ABS. The rule-based method involves identifying aspect terms using a list of pre-defined keywords, while the machine learning-based method uses a Support Vector Machine (SVM) classifier to classify the sentiment of the review.

Similarly, another study by Kolekar et al. (2018) proposed a hybrid approach that combines rule-based and machine learning-based techniques for ABS in Hindi. The study used a rule-based system for aspect extraction, followed by a Naive Bayes classifier for sentiment classification. The study showed promising results, achieving an accuracy of 75

However, these studies only provide a limited understanding of ABSA in Hindi. There is still a need for research that focuses on developing high-performing models for ABSA in Hindi using deep learning methods, similar to those used in English ABSA. Additionally, research on the dataset for ABSA in Hindi is necessary to gain a better understanding of the linguistic features of the text and to develop models that can perform well on the data.

2.4.3 Filling the Gaps

In this thesis, we aim to bridge the gap in ABSA research for low-resource languages, particularly for Hindi. Our research focuses on exploring the Hindi Review Corpus dataset to gain a deeper understanding of the linguistic features of the text. We propose a novel deep learning model for ABSA in Hindi that utilizes a combination of attention mechanisms and convolutional neural networks. The proposed model achieves high accuracy on the Hindi Review Corpus dataset, and we analyze the results to provide insights into the performance of the model. The contribution of our research is not only the development of a high-performing model for ABSA in Hindi but also the interpretability of the results and a better understanding of the Hindi ABSA Corpus dataset, followed by development of a new one. ABS

Chapter 3

Exploring Sentiment Analysis in Low Resource Settings

"A word is characterized by the company it keeps."

— John Rupert Firth

In this chapter, we attempt to solve sentiment analysis in a low-resource setting by employing transfer learning techniques.

3.1 Introduction

Sentiment analysis refers to a series of methods, techniques, and tools aimed at extracting the intended sentiment from a written opinion. Traditional sentiment analysis techniques have relied on using supervised term weighting methods including terms' distribution of classes, word-level polarity scoring and using SVMs [21] and Naive Bayes classifiers for pattern extraction using hand-crafted features. The advent of deep learning techniques for sentiment analysis has now enabled the extraction of high quality sentiment data from written texts. One majorly overlooked factor in the performance of these neoteric approaches is their dependency on large annotated datasets compiled from multiple data sources related to or sourced from newspapers, tweets, photos and product reviews. [66, 37, 69, 33].

Given global nature of the current information sharing infrastructure, most data generated belongs to one of the three languages : English, Mandarin or Spanish. This abundance of raw data aids and motivates the creation of annotated resources in these languages. Conversely, the paucity of annotated data in most languages makes it a challenging task to develop deep learning based solutions for them. Hence there is a pressing need to pay special attention to developing solutions capable of sentiment analysis in a low resource setting.

Some of the initial methods that attempt to tackle this problem of data scarcity using transfer learning (training a neural model on one language and applying the trained model on another language via weight sharing) do not perform well due to the limited overlap between the vocabularies of the different languages and difference in their syntactic structure [17].

Cross-lingual sentiment classification (CLSC) methods try to alleviate this problem by leveraging labeled data from one language to improve the performance on another language [11]. However, these methods typically rely on auxiliary cross-lingual resources such as a parallel corpora [77, 75], bilingual lexicons [49] or the use of machine translation systems [35, 72, 36]. Unfortunately, the curation of such cross-lingual resources is both a time and a labour intensive task. Hence, there is a need for architectures that can perform well in the absence of such cross-lingual resources.

In this chapter, we address this problem by presenting a neural *Language Invariant Sentiment Analyzer* (LISA) architecture that is capable of training on multiple monolingual sentiment labelled datasets to learn language agnostic sentiment features that can be transferred to perform sentiment analysis in low-resource languages **without leveraging any form of cross-lingual supervision**.

Approach : We formulate this problem as a *multi-lingual transfer learning* (MLTL) language adaptation task where we attempt to learn language agnostic sentiment features via adversarial training on labelled documents $(s_1, s_2...s_n)$ from multiple (source) languages to improve the performance on documents $(t_1, t_2...t_m)$ from a low resource (target) language. The key components of our approach include learning monolingual word embeddings from $s_1, s_2...s_n, t_1, t_2...t_m$ and projecting them to a shared multilingual semantic space. We employ an LSTM network to learn latent features (z) from this multilingual space which is then used by a sentiment classifier $(\mathcal{S}_\mathcal{C})$ to predict the sentiment polarity of a document $d \in s_1...s_n, t_1...t_m$. Concurrently, a language classifier $(\mathcal{C}_\mathcal{L})$ is trained to predict the language of document d based on z. During the adversarial training we try to minimize the binary cross-entropy loss of $\mathcal{C}_\mathcal{S}$, while at the same time we maximize the cross-entropy loss of $\mathcal{C}_\mathcal{L}$. This results in a setting where the LSTM learns to produce latent features z that predicts the sentiment of document d correctly independent of the language of document d. We hypothesize that in this setting, the latent features (z) trained would contain sentiment features that are language agnostic.

In summary, the main contributions of this chapter are :

- We introduce a language independent neural architecture for sentiment analysis without the use of language specific features or cross-lingual supervision.
- We provide extensive evaluations of the LISA architecture in two settings :
  (i) **Low-resource Setting** : Where labeled data in the target language is available in limited amounts.
  (ii) **No-resource Setting** : Where the is no labeled data available in the target language.
- Our experiments on the Multilingual Amazon Review Text Classification dataset and the Sentiraama dataset show that the proposed LISA architecture achieves better performance compared to prior work in the low-resource setting.

3.2 Dataset Description

We conduct our experiments on two publicly available sentiment classification datasets :

The Multilingual Amazon Review Text Classification dataset [36] consists of sentiment labelled data in multiple languages. The vast amount of prior work on this dataset helps us to directly compare our results with the pre-existing state-of-the-art CLSC methods.

The Sentiraama Corpus [24] is a real-world low resource sentiment corpus in Telugu (an agglutinating Indian language). We use this dataset to test the robustness of our system and evaluate our results in a truly low resource setting.

In the following subsections we describe both the corpora in detail.

3.2.1 Multilingual Amazon Review Text Classification dataset

The Multilingual Amazon Review Dataset contains sentiment labeled product reviews in four languages (English, German, French and Japanese) across three domains (Books, Dvd and Music). The German, French and Japanese reviews were crawled from Amazon and the corpus was enhanced with English reviews from [13]. Each review contains a domain label, a review summary, a review text, and a rating from the set $1, 2, 4, 5$ where $1, 2$ denotes negative sentiment and $4, 5$ denotes positive sentiment. The reviews in each domain for each language are split into three disjoint balanced sets, namely, Train set, Test set and Unlabeled set. The dataset statistics are presented in Table 3.1.

		Train	Test	Unlabelled
English	Books	2000	2000	50000
	DVD	2000	2000	30000
	Music	2000	2000	25220
German	Books	2000	2000	165470
	DVD	2000	2000	91516
	Music	2000	2000	60392
French	Books	2000	2000	32870
	DVD	2000	2000	9358
	Music	2000	2000	15940
Japanese	Books	2000	2000	169780
	DVD	2000	2000	68326
	Music	2000	2000	55892

Table 3.1 Multilingual Amazon Review Text Classification dataset statistics

3.2.2 Sentiraama Dataset

The Sentiraama dataset consists of sentiment labelled documents in four domains : Books, Movies, Products and Song Lyrics. Each document is given a positive or a negative label. The corpus statistics are presented in Table 3.2.

	Books	Movies	Products	Lyrics
Positive	100	136	100	230
Negative	100	131	100	109
Total	200	267	200	339

Table 3.2 Sentiraama corpus statistics

To avoid cross-domain discrepancies we restrict our experiments to the Books and Movies domain as it has similar counterparts in the Multilingual Amazon Review Dataset, i.e, Books and Dvd respectively. We divide the Books and Movie domains of the Sentiraama dataset to create a Train set and a Test set using an 80-20 train-test split. The statistics of the subset of the corpus that are used in our experiments are listed in Table 3.3.

	Books		Movies	
	+ve	-ve	+ve	-ve
Train	80	80	108	105
Test	20	20	28	26

Table 3.3 Subset of the Sentiraama corpus used in our experiments

3.3 LISA Architecture

The input to the LISA model is a review r_i that is made up of a sequence of words $w_1, w_2, \ldots w_k$. Each review r_i is associated with a language label $l_i \in L$ where $L = \{l_1, l_2, \ldots l_p\}$ is the set of all language labels used in training. Additionally, each review r_i is also associated with a sentiment label $t_i \in \{positive, negative\}$ which denotes the sentiment polarity of the review. We project each word w_i to the multilingual semantic space (from section **??**) to obtain a sequence of n-dimensional word embeddings $e_1, e_2, \ldots e_k$ where $e_i \in \mathbb{R}^n$.

The following subsections describe in detail the individual components of the LISA architecture. Figure 3.1 shows the overall architecture of the proposed model.

3.3.1 Multilingual Sequence Encoder ($\mathcal{H}$)

The Multilingual Sequence Encoder ($\mathcal{H}$) processes the sequence of word embeddings ($e_1, e_2, \ldots e_k$) and transforms it into an m-dimensional (hidden) vector $\mathcal{H}(r_i)$. To this end, the embeddings for all the words in review r_i are passed sequentially through a Long Short-Term Memory (LSTM) network . LSTMs are a variant of RNNs that learns features that model the long-term dependencies between the

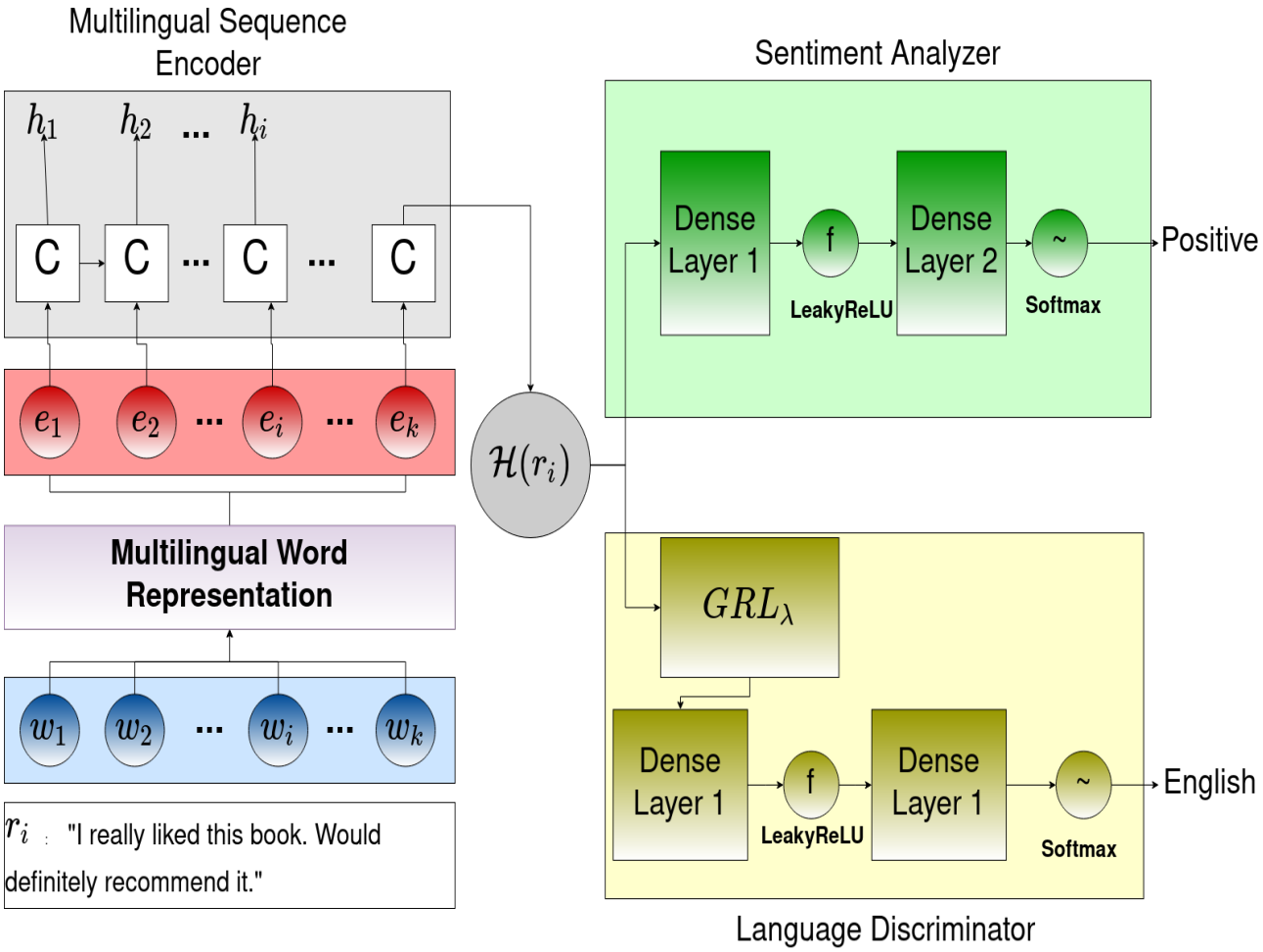

Figure 3.1 The architecture of the proposed model

words. The LSTM network, at each time step outputs a hidden state h_i for every input word embedding e_i, such that :

$$h_i = \text{LSTM}(e_i, h_{i-1}) \in \mathbb{R}^m$$

The final hidden state $\mathcal{H}(r_i) = h_k$ is then passed through a Language Discriminator ($\mathcal{C}_\mathcal{L}$) and a Sentiment Analyzer ($\mathcal{C}_\mathcal{S}$).

3.3.2 Language Discriminator ($\mathcal{C}_\mathcal{L}$)

The goal of the Language Discriminator ($\mathcal{C}_\mathcal{L}$) is to predict the language label l_i based on $\mathcal{H}(r_i)$. In other words, $\mathcal{C}_\mathcal{L}$ tries to predict the language from which the sequence of words $w_1, w_2, \ldots w_k$ come from. The $\mathcal{C}_\mathcal{L}$ comprises of a Gradient Reversal Layer (GRL_λ), followed by two Dense Layers and an output Softmax Layer that applies the softmax function over all the languages used in training. During

backpropagation, GRL_λ multiplies the gradients by a factor of $-\lambda$ and during the forward pass it acts as the identity function. λ is hyperparameter in the network.

3.3.3 Sentiment Analyzer ($\mathcal{C}_\mathcal{S}$)

The Sentiment Analyzer ($\mathcal{C}_\mathcal{S}$), as the name suggests, tries to predict the sentiment label t_i of the input review r_i based on $\mathcal{H}(r_i)$. The $\mathcal{C}_\mathcal{S}$ is made up of two Dense Layers followed by an output Softmax Layer that applies the the softmax function over the two sentiment polarities (positive and negative).

	German			French			Japanese		
	Books	DVD	Music	Books	DVD	Music	Books	DVD	Music
CL-MT	79.68	77.92	77.22	80.76	78.83	75.78	70.22	71.30	72.02
BiDRL	84.14	84.05	84.67	84.39	83.60	82.52	73.15	76.78	78.77
UMM	81.65	81.27	81.32	80.27	80.27	79.41	71.23	72.55	75.38
Bi-PV	79.51	78.60	82.45	84.25	79.60	80.09	71.75	75.40	75.45
CR-RL	79.89	77.14	77.27	78.25	74.83	78.71	71.11	73.12	74.38
CL-SCL	79.50	76.92	77.79	78.49	78.80	77.92	73.09	71.07	75.11
MAN-MoE	82.40	78.80	77.15	81.10	84.25	80.90	62.78	69.10	72.60
LISA-LR	**85.45**	**84.90**	**86.55**	**86.25**	**85.35**	**85.60**	**79.20**	**83.30**	**80.892**
LISA-NR	55.60	55.50	58.90	68.95	70.65	64.30	62.20	56.50	59.80
LISA-NoLD	81.20	77.70	80.75	82.80	80.10	80.50	79.05	83.15	**82.542**

Table 3.4 Results on the Multilingual Amazon Review Text Classification dataset. The numbers denote binary classification accuracies

3.4 Adversarial Training

Inspired by recent works [25], we train the LISA model using adversarial training on a set of labeled reviews $R = \{r_1, r_2, \ldots r_n\}$. The aim of the LISA model is to predict the sentiment label t_i for a given review r_i independent of the language label l_i.

We formulate the learning objective in a way that minimizes the sentiment classification loss from $\mathcal{C}_\mathcal{S}$ and maximizes the language classification loss from $\mathcal{C}_\mathcal{L}$. As a result, the LISA model tries to jointly optimize the below functions:

$$\arg\min_{\mathcal{H},\mathcal{C}_\mathcal{S}} f(\mathcal{C}_\mathcal{S}(\mathcal{H}(r_i)), t_i) - f(\mathcal{C}_\mathcal{L}(\mathcal{H}(r_i)), l_i) \tag{3.1}$$

$$\arg\max_{\mathcal{C}_\mathcal{L}} f(\mathcal{C}_\mathcal{L}(\mathcal{H}(r_i)), l_i) \tag{3.2}$$

Where f denotes the loss function used. This results in a setting where the $\mathcal{C}_\mathcal{L}$ tries to predict l_i based on a given $\mathcal{H}(r_i)$ and the encoder $\mathcal{H}$ tries to 'fool' the $\mathcal{C}_\mathcal{L}$ by learning to create $\mathcal{H}(r_i)$ that is minimally influenced by the language label l_i while at the same time, is maximally influenced by the $\mathcal{C}_\mathcal{S}$ to predict he sentiment label t_i correctly.

The M-LiST model [26] presents a similar setting for the task of open domain event detection that was trained using a Gradient Reversal Layer GRL_λ between $\mathcal{H}$ and C_L. By using GRL_λ, the optimization functions (equations 3.1 and 3.2) can be simplified as :

$$\arg \min_{\mathcal{H},\mathcal{C}_\mathcal{S},\mathcal{C}_\mathcal{L}} f(\mathcal{C}_\mathcal{S}(\mathcal{H}(r_i)), t_i) + f(\mathcal{C}_\mathcal{L}(GRL_\lambda(\mathcal{H}(r_i))), l_i) \tag{3.3}$$

3.5 Experiments and Results

In this section we present an extensive set of experiments conducted on the Multilingual Amazon Review Text Classification dataset and the Telugu Sentiraama sentiment classification corpus. We evaluate our approach in the two settings described below :

3.5.1 Low-resource setting

We evaluate the performance of the LISA architecture in the low-resource setting (termed **LISA-LR**) by training it on the Train sets from multiple source languages and the limited Train set in the target language and then testing on the Test set of the target language.

3.5.2 No-resource setting

In the no-resource setting, we assume that the training data is not available for the target language. We train the LISA model (termed **LISA-NR**) on the Train sets of the source languages and evaluate the model on the target language Test set.

3.5.3 LISA - No Language Discriminator

To show the effectiveness of the Language Discriminator ($\mathcal{C}_\mathcal{L}$), we conduct ablation experiments in the low-resource setting where we remove $\mathcal{C}_\mathcal{L}$ from the LISA architecture. In this variant of the LISA model (termed **LISA-NoLD**), the Sentiment Analyzer only depends on the MUSE embeddings to learn $\mathcal{H}(r_i)$ to learn sentiment features. Our experiments show that **LISA-LR** performs significantly better in most cases than **LISA-NoLD**.

For the Multilingual Amazon Review Text Classification dataset in the low-resource setting, we train **LISA-LR** on the Train sets of all the four languages. We then test it on the Test set of the target

language. In the no-resource setting, we train **LISA-NR** on the Train sets of three languages and test it on the Test set of the fourth language. We do this for each domain in the corpus independently. We compare our results against prior state-of-the-art methods that uses Machine Translation Systems (**CL-MT** and **BiDRL**), methods that leverage cross-lingual supervision (**UMM**, **Bi-PV**, **CR-RL** and **CL-SCL**) and the cross-lingually unsupervised **MAN-MoE** method of chen2018zero. The results are presented in Table 3.4.

For the Sentiraama Corpus in the low-resource setting, we train **LISA-LR** by leveraging the Train sets of all the languages in the Multilingual Amazon dataset along with the Sentiraama Train Set. We then test the system on the Sentiraama Test set. In the no-resource setting, **LISA-NR** only utilizes the Train set of all the languages in the Multilingual Amazon dataset and test the system on the Sentiraama Test set. We do this for the Books and Movies domain separately. We evaluate the results of **LISA-LR**, **LISA-NR** and **LISA-NoLD** against the Bernoulli Naive Bayes and SVM baselines that use TF-IDF features which were set by [53]. The experimental results are given in Table 3.5

	Books	Movies
SVM	55	51.851
Naive Bayes	65	75.9
LISA-LR	**72.5**	**85.185**
LISA-NR	57.5	57.407
LISA-NoLD	67.5	68.51

Table 3.5 Results on the Sentiraama Dataset. The numbers denote binary classification accuracies. Note that the Naive Bayes and SVM accuracies presented in the table differ from the ones presented by . We attribute this to the difference in the train/test splits and the the lack preprocessing guidelines which makes it hard to adequately replicate their results

3.6 Conclusion

The results on the Multilingual Amazon Review Text Classification dataset proves our hypothesis that our model learns language invariant features that can be generalized across languages. The empirical results in Table 3.4 show that our model outperforms pre-existing state-of-the-art methods on this dataset. While our experiments on the Sentiraama dataset proves that our model can be applied in a real-world setting to enhance sentiment retrieval in a truly low resource language. The ablation experiments (LISA-NoLD vs LISA-LR) show that between language pairs that have similar syntactic structure (example : English, French and German), LISA-LR performs much better than LISA-NoLD. This shows the the performance gains over prior work are not just due to the use of MUSE embeddings. Rather, they are attributed to the adversarial training of the Language Discriminator and the Sentiment classifier that extracts language agnostic sentiment features from the MUSE semantic space. But for Japanese (which

is dissimilar with respect to other languages in the corpus), the results show that LISA-LR does not have a significant boots over LISA-NoLD. This is because our language adversarial training will retain only features that are invariant across all four languages, which is restrictive such that the information learnt will be too sparse to be useful. Finally, the poor performance of LISA-NR shows that our approach cannot be used for Zero-Shot learning but will achieve state-of-the-art performance in the presence of limited amounts of data.

Through our experiments in this work we conclude that high quality resource is very crucial when learning a task, especially in a low resource scenario, Transfer learning with no resource in the target language fails even for simple tasks like sentiment-analysis. Keeping these learnings in mind in the next chapter we explore the task of Aspect Based Sentiment analysis and dive deeper into the available resource for the same in Hindi.

Chapter 4

Exploring Aspect Extraction in Hindi

"With public sentiment, nothing can fail. Without it, nothing can succeed."

— Abraham Lincoln

Keeping in mind, the learnings in the previous chapter, We take a deeper look into the existing state of ABSA in Hindi, and propose improvements in the dataset, that will set us up for success to solve the problem in a low-resource setting.

4.1 Introduction

The ubiquity of product reviews and opinion data on the internet has led to the need for an automated solution in sentiment analysis and opinion mining [9]. A fine-grained analysis of reviews shows that users often provide different opinions about various *aspects* of a given product or service, which provides valuable insights if extracted and analyzed in context [18, 58]. The example in Figure 4.1 shows a review that highlights three different aspects of a laptop with different polarities.

Generally speaking, the issue of aspect-based sentiment analysis may be viewed as a two-step procedure. Part of the process involves two steps: the first is aspect word extraction, which is concerned with finding different phrases that indicate aspects, and the second is sentiment classification, which is concerned with categorizing the feelings concerning the aspect. As a result, a review phrase may include more than one aspect word and the emotion connected with each of the terms. A study with this level of fine-graininess offers more insight into the emotions conveyed in the written evaluations. Therefore, an increasing tendency in recent years has been towards more fine-grained sentiment analysis, i.e., aspect-based sentiment analysis, rather than at a more general level of sentiment analysis (ABSA).

4.1.1 Aspect Based Sentiment Analysis in Hindi: State of existing research

Recent literature has seen an increase in the amount of work being done in fine-graining downstream NLP tasks. One common method of fine-grained analysis is the use of aspect information. An aspect

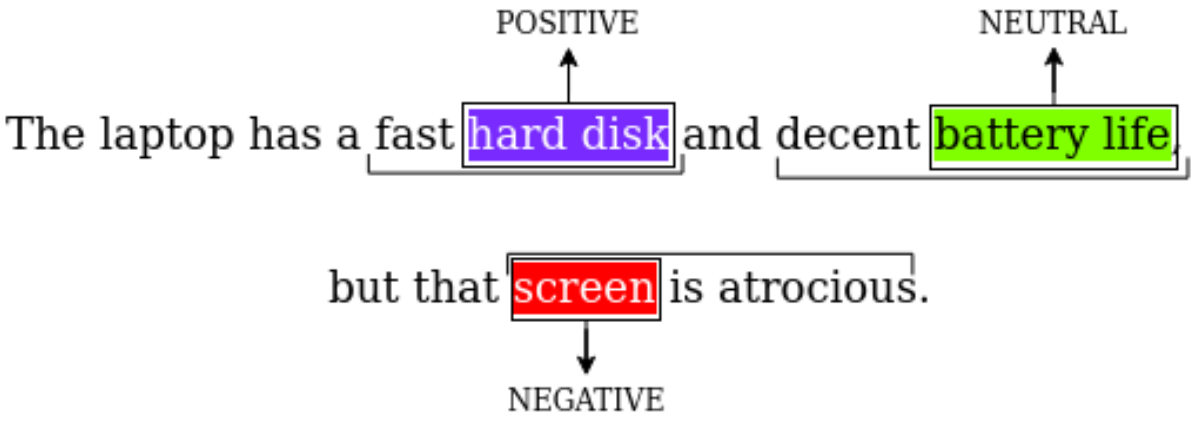

Figure 4.1 An example of aspect term extraction and aspect polarity classification. We highlight the aspects in this laptop review. The contexts for each aspect are marked using a bracket.

term is an entity of interest which identifies a unique aspect of a predefined topic or domain [58]. For example, in the *restaurant* domain, *service* and *seasoning* are aspects. While aspect extraction (AE) has been often seen as a subtask of fine grained aspect-based sentiment analysis (ABSA), recent advances in literature have established it as an independent task which can be used in other downstream tasks as well, such as summarization [23] and topic-specific information retrieval such as opinion mining [8].

Aspect extraction (as a subtask of aspect-based sentiment analysis) datasets and models have been developed for multiple languages. ABSA has been a shared task in SemEval 2014 [58], 2015 [51], 2016 [56], and as a part of the overall task of sentiment analysis on Twitter in SemEval 2017 [64]. These tasks have garnered a lot of attention in various languages including Arabic, Chinese, Dutch, French, Russian, Spanish and Turkish. Each monolingual dataset consisted of one or two domains with each language having anywhere between 4,000 to 9,000 sentences overall (including the train and test split). For Indian languages, there has been some work in developing a dataset for aspect extraction in Hindi [2] and Telugu [62].

Limited work has been done on improving the state of AE and ABSA in Hindi beyond the development of a singular dataset, namely [2]. Existing evaluations show that existing sequence tagging models (both general and specific to AE) have performed very poorly on this dataset when their performance is compared to English AE as well as in similar sequence tagging tasks in Hindi such as named entity recognition (NER) and event detection.

4.1.2 Proposed Approach : Overview

In this Chapter, we thoroughly analyze the existing dataset for AE in Hindi and explain the reason for the poor model performance. We then propose the creation of a parallel corpus, by manually translating the SemEval-2014 ABSA corpus [58]. We provide detailed guidelines and challenges faced during the creation of this resource. We show that our dataset performs much better than the existing dataset for

Hindi using baseline as well as state-of-the-art neural models for AE. Finally, we leverage the SemEval-2014 corpus to perform zero-shot and fine-tuned aspect extraction in Hindi using multilingual BERT with baseline and SoTA neural models in the dataset we have created.

Therefore, the main contributions of this work are:

- providing an in-depth qualitative and quantitative analysis of the existing Hindi AE dataset,
- creating a new resource for aspect extraction in Hindi by translating the SemEval 2014 corpus into Hindi [1],
- providing detailed guidelines and challenges associated with the creation of this corpus, as well as explaining the quality of the translations and annotations, and
- evaluating the new dataset using state-of-the-art neural sequence labeling models for aspect extraction in Hindi in monolingual and multi-lingual settings using transfer learning.

We establish that our corpus is a more robust and representative corpus for aspect extraction in Hindi, and its parallel nature can be exploited for a large number of downstream tasks including review translation, cross-lingual opinion mining, and aspect-based sentiment analysis.

4.2 Dataset Development

As discussed in Section 4.1, [2] is the only corpus for aspect term extraction and aspect-based sentiment analysis in Hindi. In this section, we discuss the inadequacy of this corpus by analyzing the data qualitatively, statistically and through experiments using established aspect extraction models. We also detail the process of creating a parallel aspect extraction dataset from the English gold standard dataset [58]. This resource can be treated as an individual Hindi aspect extraction dataset or can be considered a parallel resource for the aspect extraction task.

The annotation format used for the existing dataset and the dataset being created is the Begin-Inside-Outside or `BIO` sequence labeling format [61]. This format annotates each word with a corresponding label where a word labeled `B` denotes the first word of an aspect, `I` denotes any word within the aspect span and `O` denotes the words outside the aspect span.

4.2.1 Analyzing Existing Datasets

In this section, we aim to prove based on a qualitative and statistical analysis of the Hindi ABSA dataset for AE, and compare it to the SemEval-2014 English ABSA dataset. Some of the metrics for comparison include the number of sentences, number of aspects, ratios of `B`s, `I`s, and `O`s and the number of marked sentences (sentences with one or more aspects). We explain how these comparisons explain the quality of the dataset for this task as well. We also show a quantitative performance analysis of these

[1] `https://drive.google.com/file/d/1wrCHi3VbQjosvhmpfS577TXZL-O6Ubo/view?usp=sharing`

datasets on baseline model as well as the state-of-the-art models in sequence tagging and aspect term extraction in Section 4.4.1.

The [2] dataset consists of $5,147$ sentences, and a total of $4,509$ aspect terms. The combined SemEval-2016 dataset shows a similar trend, with $6,092$ sentences and $6,072$ aspect terms. However, on a closer analysis of the dataset, detailed in Table 4.1, we see three prominent distinguishing factors:

1. While the percentage of marked sentences (sentences with one or more aspects) is higher in the Hindi dataset than in the English one, there is a noticeable difference between the average number of aspects per sentence (both for marked sentences and overall).
2. The percentage of `I`s in the Hindi dataset (3.26%, $3,135$ out of $96,140$ words) are higher than the English dataset (2.96%, $2,564$ in $86,552$ words), while the number of `B`s in the Hindi dataset are lower. This implies that in multi-word aspects are far more common in Hindi than they are in English. Further, the percentage of `O`s is higher in the Hindi dataset as well, so there are not as many words which are aspect terms either.
3. The data in Hindi corpus is from *12* different domains, with some domains having less than 50 sentences. So, not only is there a large variety in topics and aspects per topic, there is also a high disparity in the number of samples per topic. In contrast, the English dataset is derived from only two different domains, with over 2000 sentences per domain.

This disparity in the number of aspects per topic as well as the noticeable difference in the number of multi-word aspect terms implies that corpus developed by [2] is sparse with very few examples of the syntactic features, aspects and their categories.

Further qualitative analysis of the data reveals discrepancies in the data creation methodology, particularly surrounding technical terms which do not have commonly used translations. Terms such as 'computer', 'megapixel', 'quad core' and 'processor' have been transliterated in some examples and have been left Romanized in others. Given the low number of examples per category, this inconsistency contributes to the data sparsity. Figure 4.3 shows a few examples of such inconsistencies.

Finally, we see examples of incorrect annotation which also contributes to the dataset quality in terms of performance in machine learning models. These incorrect annotations include incorrect spacing between words in the original review text, incomplete aspect annotation where the last character of the last word of the aspect was not a part of the aspect span, and subword level aspects due to stemming, lemmatization and dehyphenation.

There are two available task performance measures for the term of aspect extraction in the Hindi dataset:

- [2] analyzed aspect term extraction using the `BIO` annotation using conditional random fields (CRFs) for sequence labelling. They report an average F1 score of just 41.07%. The CRFs used were heavily feature engineered to use features such as semantic orientation, local context tagging and bigram specific features.
- [3] performed joint modeling and end-to-end aspect extraction on both the Hindi as well as the [58] English dataset. They reported a maximum F1 score of 83.36% for the English dataset using

<sentence id="lap_271"> स्वाइप अल्टीमेट में 8 मेगापिक्सल कैमरा पीछे की तरफ और 2 **एमपी** कैमरा आगे की तरफ दिया गया है। </sentence>	<sentence id="mob_771"> 20**MP** रियर स्नैपर इष्टतम प्रकाश की स्थितियों के तहत महान छवियों को लेता है और कम प्रकाश के तहत भी अच्छी तरह से प्रदर्शन करता है। </sentence>
<sentence id="lap_77"> मुझे लगा था कि ये सिर्फ मेरे ही **कम्प्यूटर** में है परंतु नहीं ये समस्या बहुत से, हजारों-लाखों एचपी/कॉम्पैक नोटबुक/लैपटॉप में है। </sentence>	<sentence id="lap_249"> इस Chip **Computer** में ऑलविनर एसओसी का इस्तेमाल किया गया है जो सस्ता होने के साथ-साथ पावरफुल भी है। </sentence>
<sentence id="mob_162"> एस60 में 5 इंच की एचडी आईपीएस स्क्रीन दी गई है साथ में 64 बिट 1.2 गिग **क्वॉड कोर** क्वॉलकॉम स्नैपड्रैगन 410 **प्रोसेसर**, 2 **जीबी** रैम और माली 450 एमपी जीपीयू। </sentence>	<sentence id="mob_667"> **Quad Core Processor**, 1 **GB** RAM, परफॉरमेंस अच्छी है। </sentence>

Figure 4.2 Some examples of inconsistent samples in the Hindi dataset. The words in bold face are the same in both examples, transliterated into Devnagari on the left and left Romanized on the right in different training samples.

an end-to-end architecture, while the maximum F1 score for Hindi using the same architecture was 52.03%. Other experiments also show this vast disparity.

These discrepancies show that even heavily feature-engineered statistical models as well as neural models do not perform well on the existing Hindi dataset and the neural models seem to perform a lot better on the SemEval 2014 dataset. An aspect term extraction task comparison for various models can be found in table 4.3 for a number of models described in section 4.4.1.

Table 4.3 shows that the discrepancies noted by [3] continue to hold across multiple neural models. The difference between the F1 scores between the two datasets is nearly 40% for all three models, with the maximum F1 score in the Hindi dataset being a mere 38.21% for the DeCNN model. We conclude through this thorough analysis that the [2] dataset is inadequate as a benchmark dataset for aspect extraction in Hindi.

4.2.2 Constructing the Parallel Corpus

We construct a parallel corpus by translating the SemEval 2014 English aspect based sentiment analysis dataset of restaurant and laptop reviews [58]. The dataset constructed by this translation can be used as an independent Hindi dataset, or can be used such that it leverages the English dataset for aspect extraction. By using the guidelines provided below, we are able to preserve the diversity of syntactic constructions from the original dataset, making the quantitative comparisons more representative.

The final dataset constructed by this methodology consists of 5989 sentences with 5864 aspects. Not all the sentences could be translated based on our guidelines which aim at maintaining naturalness and fluency. The guidelines pertaining to the translation and aspect extraction have been discussed below, followed by the methodology of annotation. The comparative statistics of this dataset can be found in 4.1, when compared to [2] and [58].

4.2.2.1 Annotation Guidelines

The guidelines for creating this parallel corpus were twofold, translating the dataset into Hindi and identifying the aspect terms in the translations.

The translation methodology adopted for this task had to account for fluency, accuracy and style. Not only did the translated reviews had to be as semantically similar to the original review as possible, but they also had to be faithful to the style of restaurant and technology reviews in Hindi. In order to achieve a natural translation true to this style, we propose the following translation guidelines.

1. For proper nouns and other names in English, such as locations, company names and other named entities, annotators were asked to directly use Roman script. For example: *Brooklyn, 2nd Street, Sony*. We found that proper nouns in both domains indicated a property of the main topic and rarely that of an aspect, so using Roman script could aid in attribute extraction without being a problem in aspect extraction or other downstream tasks.
2. For common nouns without Hindi translations, or with very obscure translations which are not commonly used, annotators were asked to transliterate these nouns into Hindi. This was done in order to maintain consistency in the use of technical terms which could act as aspects in the Hindi sentence, while maintaining the domain-specific naturalness and fluency of the translated sentence. Word such as *keyboard, bluetooth, monitor, sake* and *soy sauce* were transliterated into Hindi.
3. Aspect descriptions often contain idiomatic constructions or other compositional phrases. Translators were asked to simplify these phrases to their meaning rather than translate word for word. Therefore, for phrases such as '*on the nose*' was translated to *yathaarth* (meaning 'obvious') rather than *naak ke upar* (literally meaning 'on or over the nose')
4. For common nouns with gender and number inflections, annotators were asked to transliterate the root word (as mentioned in rule (2)) but use the Hindi inflection markers. As English pronouns and nouns are not gender marked, the default male inflection is used whenever applicable.
5. For all other words, aspects and aspect descriptions, translate into Hindi using the most commonly used words given the appropriate context. In the case where the context is so scarce that there is no way to translate the sentence in a way that preserves meaning, do not translate the sentence.

After the translation, a different group of annotators were asked to identify aspect terms. Aspect term identification guidelines were the same as those used in the SemEval-2014 ABSA task[2] [58]. The

[2] http://alt.qcri.org/semeval2014/task4/data/uploads/semeval14_absa_annotationguidelines.pdf

annotators were asked to annotate all single or multiword terms which were a particular aspect of the target entity (i.e. 'Restaurant' or 'Laptop').

Metrics	[2]	[58]	Our Dataset
Total # Sentences	5417	**6092**	5989
Total # Aspects	4509	**6072**	5864
Total # Tokens	96140	86552	**104618**
% Sentences marked with Aspects	**61.5%**	57.7%	57.9%
Avg. # of aspects per sentence	0.81	**0.99**	0.98
Avg. # of aspects per marked	1.32	**1.72**	1.69
% of `Bs`	4.69%	**7.01%**	5.60%
% of `Is`	**3.26%**	2.96%	2.80%
% of `Os`	**92.15%**	90.22%	91.60%
No. of Domains	**12**	2	2

Table 4.1 Some basic comparative statistics between the aspect extraction and aspect based sentiment analysis datasets. We see that while the Hindi dataset has lower number of samples, fewer aspects, lower ratio of aspects per sentence and lower number of sentences with aspects. Interestingly, however, these words have been added to a much larger number of domains in Hindi and there are higher number of words with the `I` and `O` tags

4.2.2.2 Annotation Methodology

Each sentence in the [58] dataset was translated by four translators, two undergraduate and two graduate students. All translators are bilingual speakers of Hindi and English and are between the ages of 18 and 22. The translated sentences were then provided to two other annotators for the aspect extraction task. These annotators were in the same age group and of the same composition in terms of expertise in Hindi and English.

Translation was performed in two phases: *aspect-aware* and *aspect-blind* translations. In aspect-aware translation, the translator were provided the aspect terms while translating the sentence and were to retain as many aspects in the translated sentence as possible while maintaining the rules of translation mentioned above. In the aspect-blind translation, the translators were provided just the sentence to translate with no additional instructions. This two-phase translation was done to determine the fluency and naturalness of the translations with respect to one another with and without the constraint of maintaining aspects. The dataset contains the most fluent version of the annotations and those which maintain the most aspects from the source sentences in the SemEval dataset.

These translated sentences were provided to the final annotators, who were asked to identify the aspects in these sentences based on the guidelines provided above. This was compared to a direct translation of the extracted aspects in the source sentence (which were provided in the dataset).

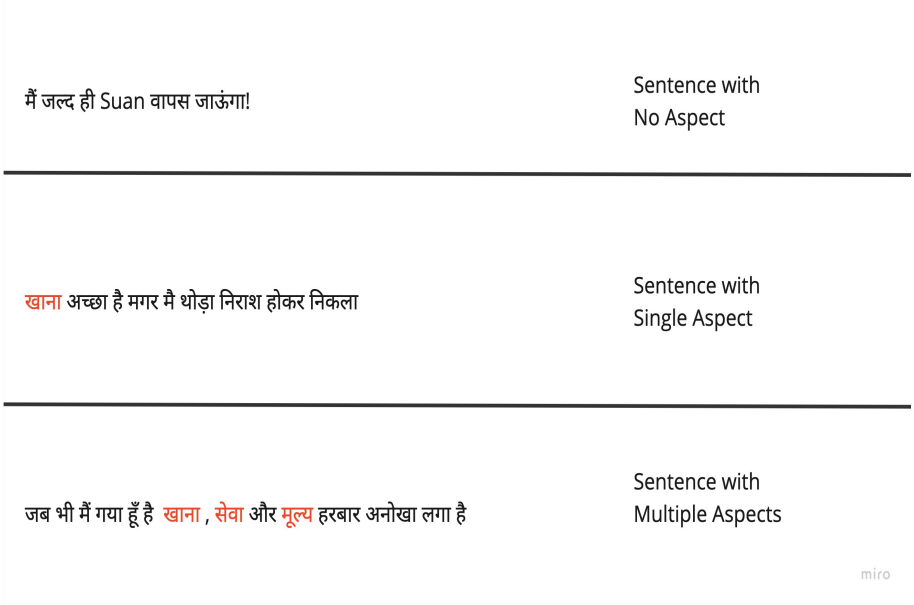

Figure 4.3 Examples of samples with No, one and Multiple aspects from our created dataset

4.2.2.3 Challenges in Annotation

Some of the main challenges in translating the data are detailed below.

1. The most common problem in translation was semantically compositional constructions such as idiomatic phrases. Phrases such as "*boy oh boy*", "*don't look down your nose*" etc. were descriptive of a given aspect in the corpus, but could not be easily translated due to a lack of natural corollaries for these phrases in Hindi.
2. Constructions with puns and aspects embedded in the compositional constructions were the biggest challenge to the translation. For example: '*But that wasn't the icing on the cake: a tiramisu that resembled nothing I have ever had*' had the aspect '*icing on the cake*' which is both literal and metaphorical in this sentence. In the final version of the data, these sentences have not been included due to very high disparity between the translations and the difficulty in extracting aspects.
3. Elided references were a concern for translators. For example, a sentence such as '*A cheap eat for NYC, but not for dosa.*' uses the term '*eat*' to refer to a '*place to eat*' which is also an aspect in this sentence. A direct translation forces this elision to be explicit, which also changes the aspect term.
4. Hindi syntax has relatively free word order, which affords fragmentation of noun and verb phrases by adjectives and adverbs respectively. The aspect-aware and aspect-blind translations often differed in such cases, as the aspect-aware translation is not fragmented, but is also generally unnatural according to the annotators. For example, the phrase "*everything bagel with lox spread*" has the annotated aspect *bagel with lox spread*, but gets translated to *lauks spred ke saath evarithing begal* (where the word "everything" fragments the aspect term).

5. Certain aspect terms translate only based on context, which is not always provided in the data. An example of this is ... *mine was well done and dry* without a subject in reference, where the term *well done* can have different translations in different contexts (such as a well-done steak versus an actual compliment).

Due to these challenges in dataset annotation and the lack of context to make an informed translation which was natural and fluent, some sentences and aspects could not be translated into Hindi. Therefore the Hindi dataset has a few fewer sentences that the English dataset. The final translated dataset consists of 5,989 sentences with 5,864 aspect terms.

4.3 Dataset Analysis

In this section, we show some basic statistical analyses of the dataset including the annotator performance in translation and aspect term extraction. For translation performance, we compare the ROUGE-L scores across the translators, while for the annotation task, we use the Fleiss' Kappa metric.

We evaluate these translation based on the ROUGE-L metrics as the average review length is no more than 15 words, and most words have only one (or very few) variations in translation. Given the stringent translation/transliteration guidelines, lack of extensive vocabulary in the descriptions and less number of words per sentence, the ROUGE-L metric is a decent approximation of the translation quality provided by the annotators. The ROUGE-L metric also accounts for the relative free-word order nature and constituent rearrangement [43].

ROUGE-L is the comparison of the longest common subsequence between two translated phrases. Given the translations X of length m and Y of length n, the ROUGE-L score is given by:

$$R_{lcs} = \frac{\text{LCS}(X, Y)}{m} \tag{4.1}$$

$$P_{lcs} = \frac{\text{LCS}(X, Y)}{n} \tag{4.2}$$

$$F_{lcs} = \frac{(1 + \beta^2) R_{lcs} P_{lcs}}{R_{lcs} + \beta^2 P_{lcs}} \tag{4.3}$$

where $\beta = \dfrac{P_{lcs}}{R_{lcs}}$ when $\dfrac{\partial F_{lcs}}{\partial R_{lcs}} = \dfrac{\partial F_{lcs}}{\partial P_{lcs}}$. This value is an F-measure. In Table 4.2 we show the comparison between the ROUGE-L scores of the aspect-aware and aspect-blind translations, by taking a weighted average over the entire dataset based on the number of words in the source and target sentence. We also show the score of the translation with the highest ROUGE-L score with the rest of the translations which has been used in the dataset.

Aspect extraction is treated as a sequence labeling task and is evaluated using the Fleiss Kappa metric [22]. Fleiss' Kappa is a multiclass inter-annotator agreement score which is computed as follows:

$$\kappa = \frac{\bar{P} - \bar{P}_e}{1 - \bar{P}_e} \tag{4.4}$$

Comparison	ROUGE-L	Fleiss' Kappa
Aspect-aware	0.8994	0.8961
Aspect-blind	0.8722	0.9244
Overall	0.8960	0.9130

Table 4.2 The average ROUGE-L and Fleiss' Kappa score in the translation and annotation tasks respectively

where $P - P_e$ is the actual degree of agreement achieved and $1 - P_e$ is the degree of agreement above chance. Given N tokens to be annotated and n annotators, with k categories to annotate the data. We first calculate the proportion of annotations in the j^{th} domain as:

$$p_j = \frac{1}{Nn} \sum_{i=1}^{N} n_{ij} \ , \quad 1 = \sum_{j=1}^{k} p_j \tag{4.5}$$

We then calculate P_i, the degree of agreement with the i^{th} annotator as:

$$P_i = \frac{1}{n(n-1)} \sum_{j=1}^{k} n_{ij}(n_{ij} - 1) \tag{4.6}$$

$$= \frac{1}{n(n-1)} \left[\left(\sum_{j=1}^{k} n_{ij}^2 \right) - n \right] \tag{4.7}$$

Finally we calculate $\bar{P}$ and $\bar{P}_e$ as:

$$\bar{P} = \frac{1}{N} \sum_{i=1}^{N} P_i \tag{4.8}$$

$$\bar{P}_e = \sum_{j=1}^{k} p_j^2 \tag{4.9}$$

The Fleiss' Kappa scores of the aspect-aware, aspect-blind and overall translations are provided in table 4.2. The high Fleiss' Kappa scores indicate the confidence in the aspect identification guidelines.

Note that the ROUGE-L score of the aspect-aware translation is higher than the overall as well as the aspect-blind translations, as translators often resorted to word-for-word translations in order to preserve each and every aspect of the sentence with its associated semantic information. Note that ROUGE-L is the weighted average of the F-measure taken over all the sentences in the dataset, weighted based on the number of words in the source and target sentences. For the overall ROUGE score, the weighted average was taken over the dataset, weighted based on the number of words in the sentence which gave the highest comparative score for each translation.

Another important insight into the corpus is the difference in aspect coverage between the aspect-aware and the aspect-blind translations. As mentioned in 4.2.2, aspect-blind translations often dropped aspects due to constraints in syntactic representation or incoherent translation due to sentence semantics, such as due to complex idiomatic phrases. The difference in aspect coverage was seen in about 6% of the corpus, specifically, 358 sentences overall.

4.4 Dataset Evaluation

In this section, we detail the evaluation of our translated aspect extraction dataset. We evaluate our dataset using multiple monolingual and multilingual models. The monolingual models are trained and tested on the individual language datasets while the multilingual models involve the use of transfer learning from the SemEval-2014 dataset to the dataset we have created.

4.4.1 Monolingual Aspect Extraction

Model	[2]	[58]	Our Dataset
Baselines			
CRF	22.08	**54.97**	47.07
BiLSTM	20.71	**61.01**	54.77
BILSTM-CRF	34.71	**62.61**	50.26
Neural SoTA Models			
BiLSTM-CNN-CRF	36.56	**73.03**	67.08
DeCNN	38.21	**77.67**	68.35
Seq2Seq4ATE	35.04	**78.86**	68.61

Table 4.3 F1 scores of established models on the monolingual aspect extraction task

We evaluate our dataset against the existing Hindi dataset and the SemEval 2014 dataset using the following baselines:

- **CRF**: We use a conditional random field with basic features[3] such as word form and POS tag.
- **BiLSTM**: We use a vanilla BiLSTM as a baseline model for aspect extraction as it is an established baseline in seq2seq tasks [46].
- **BiLSTM-CRF**: We use a BiLSTM to encode the input sentence and a conditional random field for the sequence labeling. This is a commonly used baseline for sequence tagging tasks [32].

We also use the following neural models for our analysis:

[3] `https://sklearn-crfsuite.readthedocs.io/en/latest/tutorial.html`

- **BiLSTM-CNN-CRF**: The state-of-the-art in neural named entity recognition. The architecture uses both character and word level features in a CNN and BiLSTM respectively, and using a CRF for sequence labeling tasks [63]. We use a slightly modified version where word embeddings are generated by concatenating character embeddings, as done by [59] for event detection in Hindi.
- **DeCNN**: The commonly adopted model for aspect extraction specifically, this model uses a combination of general and domain based embeddings in multiple convolutional layers and a fully connected layer with softmax for label prediction [73].
- **Seq2Seq4ATE**: This model is a sequence-to-sequence model for aspect terms extraction. The model uses a BiGRU encoder and a position aware attention variant of gated unit networks as a decoder with softmax for label prediction [48].

For consistency, in all the above mentioned models, we use the FastText embeddings for word as well as character embeddings for both English and Hindi [14, 50, 27]. For the English dataset, we use the [58] train-test split (3045 training to 800 test sentences and 2000 training to 676 test sentences in the 'Laptop' and 'Restaurant' domains respectively). For the Hindi dataset, we use a train-test split of 4062 train to 1355 test sentences based on [3]. For the LSTM based models, we use 128 unit LSTM layers, with a hidden size of 1024, and a dropout of 0.4 over 50 epochs. For the CNN based model, we use a 128 filter network with a kernel size of 5 and hidden embeddings of size 100 and dropout of 0.4 over 50 epochs.

We find that the Seq2Seq4ATE model is the best performing model for this task across the datasets. We see that the model performance on our dataset is close to that on the English dataset. While the human aspect extraction baseline shows that there is a lot more work to be done in this task, our dataset provides an adequate baseline for this task, similar to those in the SemEval Aspect Extraction subtask [58].

4.4.2 Leveraging Parallel Data

As mentioned in section 4.2, the corpus we have developed aims to be a parallel corpus, which allows us to use language invariant, transfer learning based models for aspect extraction in Hindi. We use the BERT mutilingual sentence embeddings [19] as the sentence representations for the English and Hindi on the (a) BiLSTM, (b) BiLSTM-CNN-CRF and (c) the Seq2Seq4ATE models, mentioned in Section 4.4.1. The BERT multilingual embeddings have been used for a variety of tasks in Hindi including machine comprehension [28] and named entity recognition [55], among other sequence labeling tasks. [55] showcases the model efficacy in using monolingual corpora for zero-shot code-mixed tasks as well, which would be useful for our corpus.

We design three experiments for evaluating our dataset using M-BERT, which are detailed below.

1. *M-BERT baseline* where we train and test on the Hindi sentences and aspects from our dataset directly, using the M-BERT embeddings. This has been done to establish a baseline for our experiments that follow for leveraging the English data.

Training	Model	F1-score
Baseline	BiLSTM	41.06
	BiLSTM-CNN-CRF	54.92
	Seq2Seq4ATE	43.51
Zero-shot	BiLSTM	40.72
	BiLSTM-CNN-CRF	56.16
	Seq2Seq4ATE	42.08
Fine-tuned	BiLSTM	57.37
	BiLSTM-CNN-CRF	62.12
	Seq2Seq4ATE	**66.28**

Table 4.4 F1-score of the models by leveraging English aspect extraction data using M-BERT. The baseline score is based on using Hindi for training as well as testing

2. *Zero shot aspect extraction for Hindi* where we train using the English dataset and evaluate the model performances on the Hindi data, in order to estimate how much aspect information can be extracted about aspect representation in this data which can be applied on the Hindi dataset directly.
3. *Fine tuned aspect extraction for Hindi* where we train the models on the Hindi and a small part of the English dataset and test on the translated Hindi test set. In this experiment, we augment the training data and therefore showcase the use of the English representation of aspect terms in the dataset. This is done with the motivation to boost the token representation of English tokens, as the Hindi data contains English tokens in the form of proper nouns. These tokens are aspects in a part of the corpus and therefore introducing this experiment improves the representation and extraction of these aspect tokens.

Table 4.4 provides the F1-scores of the various models described above. We use the pretrained BERT Mulitilingual cased model. The best performing model is the fine-tuned Seq2Seq4ATE model with an F1 of 66.28. We also see that the zero-shot performance of the BiLSTM-CNN-CRF is better than the baseline, and that fine-tuning using English data definitely helps the model.

4.5 Conclusion

In this chapter, we detailed the state of aspect extraction in Hindi by thoroughly analyzing and evaluating the currently available baseline dataset for this task. By understanding the flaws in that dataset, we explain its inadequacy in terms of lack of uniformity, high domain sparsity and incorrect aspect annotations. We further compare its performance with existing models to show that it performs very poorly as compared to the existing English dataset.

We then explain the mechanism of creating a SemEval style corpus for aspect extraction in Hindi, by translating the English SemEval 2014 aspect based sentiment analysis corpus. We provide a detailed list of guidelines in order to make this task as replicable as possible. We also focus on maintaining the naturalness and fluency of the translations using transliteration wherever necessary. Our translation and annotation methodology is evaluated on the ROUGE-L and Fleiss' Kappa metrics respectively.

We use this dataset to show performance on basline statistical and neural sequence labeling models, as well as the current state-of-the-art models in neural aspect extraction such as DeCNN and Seq2Seq4ATE. We show that while the models don't perform nearly as well on the published Hindi dataset, we provide results comparable to the performance of those models on Gold Standard English Dataset. Since we have a parallel corpus, we also leverage the English data for improving aspect extraction in Hindi using multilingual BERT. And leave room for further research on cross-lingual transfer learning approaches.

We conclude that this created dataset should be the new gold standard for ABSA in Hindi. In the chapters to follow, we focus on improving the state of modelling for ABSA in low-resource languages, focusing on Hindi.

Chapter 5

Exploring Context Localization for Aspect Based Sentiment Analysis

"With limited training data, a more constrained model tends to perform better."

— Christopher Manning

In this chapter, we explore modelling techniques for learning the task of Aspect Based Sentiment Analysis, specifically in Hindi.

5.1 Introduction

Aspect-based sentiment analysis (ABSA) is a fine-grained opinion mining task for the extraction and polarity detection of topic-specific aspects from a review or opinion. This chapter introduces the Context-Localization and Pooling (CLaP) architecture, a nested sequence labeling, end-to-end model for ABSA.

Our architecture relies on convolving the representations of aspect-aware and aspect-blind sentences and combining them to detect the polarity of the aspect term. Our architecture outperforms the current state-of-the-art in English, Hindi, and Chinese in aspect term extraction and polarity classification on gold standard data using no additional training data. We analyze our model's performance with several ablations and show the importance of accounting for aspect-specific context for end-to-end ABSA.

5.1.1 Previous efforts to solve Aspect Based Sentiment Analysis

Aspect based sentiment analysis has become a prevalent natural language understanding task, especially with the datasets and baselines established by SemEval tasks [58, 57, 56]. The best performing models in SemEval tasks[1], such as [71, 65] used feature engineered statistical systems such as Kernel SVMs with Bag-of-Words, CRFs, and semi-Markov taggers. used a neural architecture for other subtasks.

[1]SemEval 2014 was the only task which featured aspect specific polarity detection. Systems in 2015-16 refer to opinion target expression

Earlier works in ABSA rely on statistical feature engineered systems [71, 29]. More recently, deep learning methods such as CNNs [53, 60], RNNs and LSTMs [52, 47], and attention-based architectures [16, 10] have been used for various subtasks in ABSA. With the increasing prominence of contextualized word representations such as BERT [20], most of the current state-of-the-art models use BERT to encode the sentence for aspect term extraction and polarity detection [30, 74]. However, most of these approaches are either disjoint or treat polarity detection as a classification task.

5.1.1.1 Aspect Term Extraction

Aspect term extraction (ATE) gained relevance as an independent sequence labeling task with baselines that use simple recurrent neural networks and their variations for an NER-like task framework [46]. Advances in architectures for ATE include [73], which uses a domain-specific embedding, [40], which retains opinion summaries for each aspect, and [48], which learns syntactic cues for a review by training positional aware attention.

5.1.1.2 Co-extracting Aspect Term and Polarity

Aspect term extraction and polarity detection are often performed together, either using joint learning or end-to-end models. Architectures such as [76] (a gated convolutional mechanism) and [39] (an attention-LSTM target detection framework) construct a single end-to-end architecture. On the other hand, systems such as MIN [42] and MATEPC [52] treat ATEPC as a joint learning task. [30, 41] are the baseline BERT-based models which fine-tune BERT for classification and sequence labeling, respectively. Further work has been done to construct auxiliary sentence construction [68] and post-training BERT using domain-specific and task-specific context information [74].

5.1.1.3 ABSA for Hindi and Chinese

In Hindi, [2] is the gold-standard dataset for aspect-based sentiment analysis. [1] used a suite of feature engineered model for aspect term extraction and aspect specific polarity classification, while [4] employs a hybrid CNN-SVM and multi-objective feature selection model. Finally, [3] performs both joint and end-to-end modeling of aspect term extraction and polarity classification using a BiLSTM-CNN model, which is the current state-of-the-art in Hindi.

5.1.2 Proposed Approach: Overview

Driven by this motivation, In this chapter, we introduce Context-Localization-and-Pooling (CLaP) architecture, a BERT-based, end-to-end model for aspect term extraction and polarity classification (ATEPC). We use a *nested sequence labeling* approach to this task, as we consider ATE and APC individual sequence tagging tasks such that the output of the ATE tagger effects the input of the APC tagger. We first use BERT to encode the input sentence for aspect term extraction, and use the sentence

and the extracted aspects to create an aspect-aware sentence representation. We then convolve both these representations to localize the contexts specific to each aspect. Our model's output is a sequence corresponding to the polarity of *each word*, from which we extract the polarity of the aspect terms.

We train and evaluate our model on the English SemEval 2014 dataset [58], the Hindi [2] ABSA dataset, and the Chinese multi-domain ABSA datasets [15]. Our model outperforms the existing state-of-the-art aspect term extraction and polarity classification. We perform model ablations to show the importance of accounting for aspect-level and sentence-level information.

5.2 Model Description

In this section, we introduce and describe the architecture of our model designed for Context-Localization-and-Pooling (CLaP). In our model, we treat aspect term extraction and polarity classification as a nested sequence labeling task. The overall structure of the model is provided below.

- First, we generate an encoded, *aspect-blind* representation of the input sentence via BERT, and leverage it for labelling the aspects (§5.2.1).
- Then, we append these extracted aspects to the input sequence, creating an *aspect-aware* representation (§5.2.2).
- Finally, we use a combination of this aspect-aware sentence representation and the aspect-blind representation to generate a polarity sequence for each word in a sentence (§5.2.3).

Figure 5.1 shows the architecture of the proposed model. We detail each part of the model in the subsections below. The experiment details and hyperparameter information is provided in §5.3.2.

5.2.1 Aspect Term Extraction

Language	**Dataset**	**Sentences**	**Aspects**		
			Positive	Negative	Neutral
English	Restaurant	3041 / 800	2164 / 728	805 / 196	633 / 196
	Laptop	3045 / 800	987 / 341	866 / 128	460 / 169
Chinese	Car	921 / 230	708 / 164	213 / 66	- / -
	Phone	1988 / 497	1319 / 341	668 / 156	- / -
	Notebook	496 / 123	328 / 88	168 / 35	- / -
	Camera	1743 / 435	1197 / 322	546 / 113	- / -
Hindi	Overall	4334 / 1083	1589 / 397	455 / 114	1531 / 383

Table 5.1 The dataset statistics for the three languages, written in a `Train / Test` format. We have not divided the Hindi data as it consists of 12 domains. There are no neutral aspects in the Chinese datasets

We first employ a BERT encoder with N transformer layers to the given input sequence $\mathbf{x}_{\text{blind}} = \{x_1, \ldots x_l\}$ of length l. Therefore, we compute the hidden representation $\mathbf{x}_{\text{ATE}} = \{h_1^N, \ldots, h_l^N\} \in \mathbb{R}^{N \times \text{dim}_h}$, where dim_h is the number of hidden dimensions. $\mathbf{x}_{\text{ATE}}$ is the *aspect-blind* representation of the sentence.

This representation is then fed to the sequence tagging layer, a feed-forward neural network layer to predict the sequence of tags $\mathbf{y}_{\text{ATE}} = \{y_1, \ldots y_l\}$, where $y_{\text{ATE}_i} = \mathbf{w}\mathbf{x}_i + b_i$ for $y_{\text{ATE}_i} \in \mathbf{y}_{\text{ATE}}$ and $\mathbf{x}_i \in \mathbf{x}_{\text{ATE}}$. The possible values of tags of y_{ATE_i} can be `B`, `I`, and `O`, used to denote the beginning, inside and outside of the span of an aspect term respectively.

We then extract the list of aspects $A = \{a_1, \ldots, a_n\}$, from the sentence based on the output sequence $\mathbf{y}_{\text{ATE}}$.

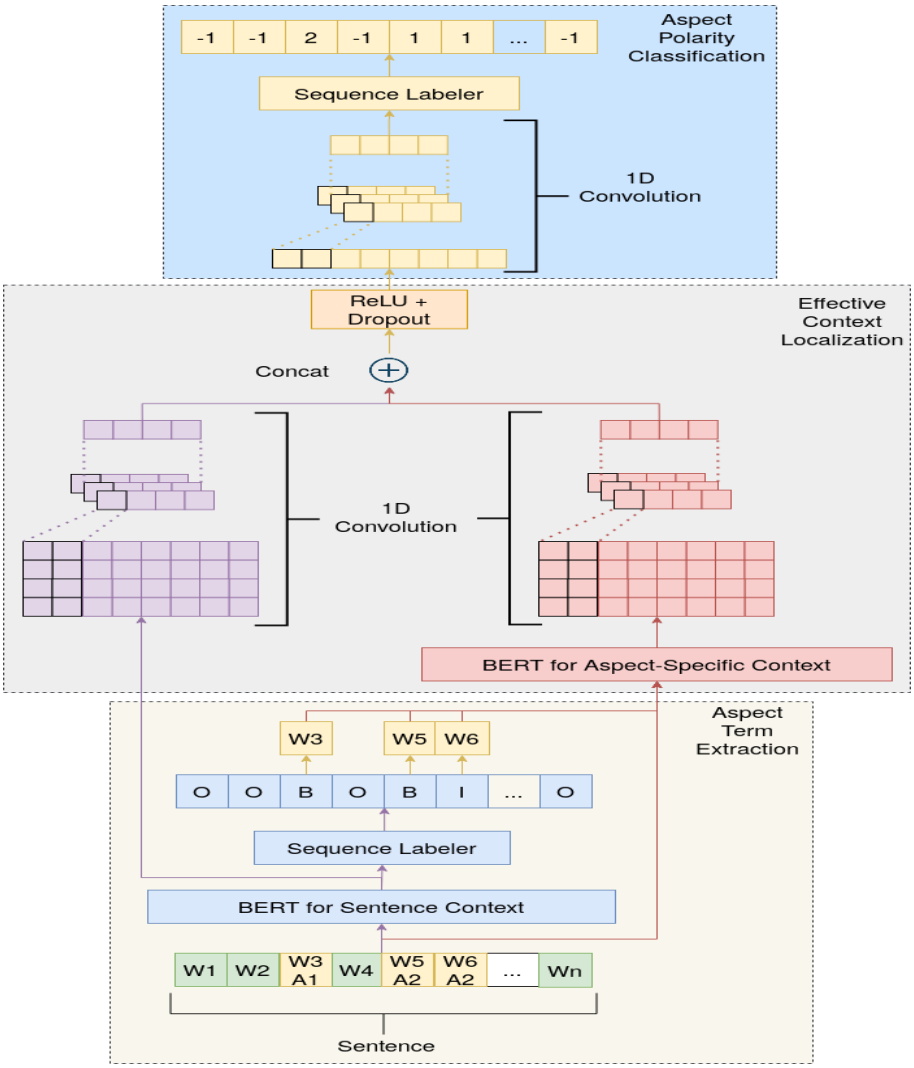

Figure 5.1 The architecture of the proposed model

5.2.2 Extracting Context and Effective Pooling using CNNs

For performing aspect term polarity classification, we first generate the *aspect-aware* sentence representation, by appending the aspects extracted above to the original sentence with a `[SEP]` token. Therefore, we generate $\mathbf{x}_{\text{aware}} = \mathbf{x}_{\text{blind}} \cup A = \{x_1, \ldots, x_l, a_1, \ldots, a_n\}$. We then employ a second BERT encoder to create the aspect-aware representation of the sentence. Therefore, we compute the representation $\mathbf{x}_{\text{APC}} = \{h_1^N, \ldots, h_{(l+n)}^N\} \in \mathbb{R}^{N \times \text{dim}_h}$.

The representations $\mathbf{x}_{\text{ATE}}$ and $\mathbf{x}_{\text{APC}}$ are passed through variable-kernel 1-D convolutions using two kernel sizes. The outputs of the convolutions of the same filter size are concatenated. This is summarized by the equations below:

$$\mathbf{y}_{f_1} = \mathbf{b}_1 + \sum_{k=0}^{f_1-1} \mathbf{w}_k * \mathbf{x}_{\text{ATE}_k} \odot \mathbf{b}_2 + \sum_{k=0}^{f_1-1} \mathbf{w}_k * \mathbf{x}_{\text{APC}_k}$$
$$\mathbf{y}_{f_2} = \mathbf{b}_3 + \sum_{k=0}^{f_2-1} \mathbf{w}_k * \mathbf{x}_{\text{ATE}_k} \odot \mathbf{b}_4 + \sum_{k=0}^{f_2-1} \mathbf{w}_k * \mathbf{x}_{\text{APC}_k}$$

where the terms $\mathbf{b}_i$ is the bias term for each convolution, and f_1 and f_2 are the kernel sizes.

The outputs of the convolutions, $\mathbf{y}_{f_1}$ and $\mathbf{y}_{f_2}$ are passed through a ReLU and a dropout layer, before being concatenated to form $\mathbf{y}_{\text{CLaP}}$. This final output tensor contains the effectively context-localized information, pooled and concatenated into a single representation.

5.2.3 Sequence Labeling for Aspect Polarity Classification

First, we perform a simple 1-D convolution on $\mathbf{y}_{\text{CLaP}}$ with a single, fixed filter size. This is then passed to a feed-forward neural network, in order to generate the final output sequence $\mathbf{y}_{\text{APC}}$. Finally, we model aspect term polarity classification as a sequence labeling task by using the representation extracted above to generate a sequence of polarity values. These steps are summarized in the equations below:

$$\mathbf{y} = \mathbf{b} + \sum_{k=0}^{f-1} \mathbf{w}_k * \mathbf{y}_{\text{CLaP}}$$
$$y_{\text{APC}_i} = \mathbf{w}\mathbf{y}_i + b_i$$

where $y_{\text{APC}_i} \in \mathbf{y}_{\text{APC}}$, $\mathbf{y}_i \in \mathbf{y}$ and f is the size of the filter. The possible values of tags of y_{APC_i} can be `-1`, `0`, `1`, and `2`, used to denote a non-polarity term, negative, neutral, and positive polarities respectively.

5.3 Experimental Setup

In this section, we provide information about the experiments and results, such as the details of the data (§5.3.1) and the hyperparameters used (§5.3.2).

5.3.1 Data

We selected the gold-standard datasets for aspect term extraction and polarity classification for English, Hindi, and Chinese. The distribution of positive, negative, and neutral aspects is provided in Table 5.1.

5.3.1.1 English

For English, we trained and evaluated our model on the SemEval-2014 Restaurant and Laptop reviews [58]. The dataset has 3041 training and 800 test sentences in the Restaurant reviews, and 3045 training and 800 test sentences in the Laptop reviews.

5.3.1.2 Chinese

For Chinese, we train and evaluate on car, phone, camera, and notebook review datasets [15]. The datasets have 921 training and 230 test sentences in the car reviews, 1988 training and 497 test sentences in the phone reviews, 1743 training and 435 test sentences in the camera reviews, and 496 training and 123 test sentences in the notebook reviews. These datasets have no neutral or conflict tags.

5.3.1.3 Hindi

For Hindi, we use the [2] dataset, which is a corpus of reviews spanning 12 distinct domains, including laptops, home appliances, speakers, smartwatches, movies, and travel. The dataset has 5417 sentences and 4509 aspects. We also try our model on the gold standard dataset we introduced in the previous chapter.

We preprocess all seven datasets by converting the XML format into a two-tiered format. For aspect term extraction, we tag it with the `BIO` format (detailed in Section 5.2.1) and for aspect-specific polarity classification, we annotate each word with the tags `-1`, `0`, `1`, or `2` (explained in Section 5.2.3). We do not consider the aspect terms with a conflict polarity.

5.3.2 Hyperparameter and Training Details

In this section, we explain the hyper-parameters used in our model for the reproducibility of results. We train our model for five epochs, with a learning rate of 3×10^{-5}. The model performs best with

a dropout of 0.3 for regularization and uses Adam optimizer [38]. We tuned the hyperparameters on a holdout dataset of 100 samples from each corpus.

We use pretrained variants of BERT-like contextualized word representations for encoding the sentences in our model. For the English datasets, we use uncased $\text{BERT}_{\text{BASE}}$, while for Hindi and Chinese corpora, we use Multilingual and Chinese BERT, respectively.[2] The sequence tagging layer for aspect term extraction is a single-layer feed-forward neural network of input size 768 units. We use two CNNs for context localization with variable kernel sizes 3 and 5 (f_1 and f_2 in Section 5.2.2), with 256 filters in each layer of both CNNs. We use a third CNN for aspect polarity detection, which has a kernel size 5, with 512 filters. The final sequence tagging layer for aspect polarity classification has 512 hidden units. We compute cross entropy loss between the original and predicted labels for both aspect term extraction and aspect polarity classification sequences. The total loss of the model is the sum of the losses of both components.

5.3.3 Variants of CLaP

The major feature of the CLaP model is the variable kernel size multi-layer CNNs, which allow us to effectively localize and pool context information from both the aspect-aware and the aspect-blind sentence representations.To test the efficacy of this system, we introduce the following variations of the CLaP model:

- **CLaP-0**: CLaP-0 eliminates the CNNs for context localization from the model. In this vanilla variant, we use a linear layer to extract aspect terms, append them to the sentence, and use a linear layer to predict the aspect polarities.
- **CLaP-F**: CLaP-F tests the effectiveness of variable kernel sizes in the model. In this variant, we fix the kernel size of all the CNNs in the model to a single value of 5.
- **CLaP-B**: CLaP-B evaluates the need for an aspect aware representation of the sentence. In this variant, we do not append the aspects to the sentence after aspect term extraction and use the aspect-blind representation for aspect polarity classification.

5.3.3.1 Adaptation of Domain Knowledge

The above model variants aim to understand the importance of various parts of our model architecture. While incorporating aspect-specific context is useful for aspect polarity classification, domain knowledge post-training has proven to be useful for ABSA as a whole [74]. Therefore, we introduce a model variant, **CLaP + DK**, which aims to further improve task performance by using pretrained, domain-specific BERT embeddings. We only tested this variant with the [58] datasets, as pretrained domain-specific sentence representations are not available for Chinese and Multilingual BERT.

[2] `https://github.com/huggingface/transformers`

Domain	Model	ATE	APC	
		F1	Acc.	Macro F1
Laptop	CLaP	**84.30**	**79.95**	**76.34**
	CLaP-0	74.46	71.95	71.04
	CLaP-F	81.19	74.57	73.20
	CLaP-B	**84.30**	74.60	73.66
	CLaP + DK	**85.48**	77.85	**76.62**
Restaurant	CLaP	**88.95**	**85.17**	**82.41**
	CLaP-0	76.02	73.45	72.06
	CLaP-F	86.64	80.05	79.70
	CLaP-B	**88.95**	83.17	81.06
	CLaP + DK	**90.28**	**85.58**	**86.70**

Table 5.2 Comparison of results with the model variants. "CLaP" represents the original model described in Section 5.2

We provide the results for the model ablations for the [58] in Table 5.2. The ablation model performance for the Chinese and Hindi corpora are provided in Appendix A.[3]

5.3.3.2 Quantitative Analysis

We detail a few observations from the results shown in Table 5.2 below.

1. Among the models that use the same language representations for encoding the sentence (i.e. all models *except* CLaP + DK), we find that the original model variant is the best performing one, while CLaP-0 shows the lowest F1-measure and accuracy scores in both domains across both tasks. This difference in performance indicates that our model can effectively extract aspect-specific context.
2. While all the model ablations are made to effect context localization and therefore the performance of the aspect polarity classification 'subtask', CLaP is an end-to-end model. This implies that errors made in aspect polarity classification also effect the aspect term extraction task. However, the F1 score of aspect term extraction for CLaP and CLaP-B (aspect-blind, i.e. aspects not appended for polarity classification) are the same for both domains.
3. We see that the standard model variant outperforms CLaP-F (fixed kernel size) by an average of 2.71 F1-score across both domains and tasks. This points to the efficacy of variable kernel size as a method of localizing contexts more effectively.
4. The CLaP + DK model variant outperforms the ablations which use standard BERT, which indicates that using domain-specific embeddings is useful. However, these representations are expensive to train as they require more than a million post-training examples for each domain [74].

[3] We also perform the results of our model and its comparisons on the Twitter English corpus in Appendix B

5.4 Results

In this section, we compare the results of our model, CLaP for two tasks, aspect term extraction, and aspect polarity classification, against the current state-of-the-art for the three languages, English, Hindi, and Chinese.

5.4.1 English

For English, we compare our method against the following results for aspect term extraction and polarity classification:

- **DeCNN**: [73] uses domain specific embeddings to achieve state-of-the-art results in *aspect term extraction* on the Restaurant reviews dataset.
- **Seq2Seq4ATE**: [48] models *aspect term extraction* as a sequence-to-sequence task and achieves state-of-the-art results on the Laptop reviews dataset.
- **AEN-BERT**: [67] proposed an attention encoder network for *aspect polarity classification*.
- **BERT-E2E**: [41] uses BERT as an encoder, and establishes baselines for using BERT in end-to-end ABSA, i.e. *joint extraction of aspect terms and polarities*
- **BERT-PT**: [74] uses a post-training architecture in order to fine-tune BERT for reading comprehension and *aspect term extraction and polarity classification*.

Domain	**Model**	**ATE**	**APC**	
		F1	**Acc.**	**Macro F1**
Laptop	DeCNN	81.59	-	-
	Seq2Seq4ATE	80.31	-	-
	AEN-BERT	-	79.93	76.31
	BERT-E2E	82.57	74.40	74.24
	BERT-PT	84.26	78.07	75.08
	CLaP	84.30	**79.95**	76.34
	CLaP + DK	**85.48**	77.85	**76.62**
Restaurant	DeCNN	74.37	-	-
	Seq2Seq4ATE	75.14	-	-
	AEN-BERT	-	83.12	73.76
	BERT-E2E	88.60	82.66	74.13
	BERT-PT	77.97	84.95	76.96
	CLaP	88.95	85.17	82.41
	CLaP + DK	**90.28**	**85.58**	**86.70**

Table 5.3 Comparison of results for English for aspect term extraction (ATE) and accuracy score for aspect polarity classification (APC). "-" represents that the paper does not perform that task, or provide a score for it

The results for the comparison are presented in Table 5.3. In the Table, we provide aspect term extraction with ATE, and we compare the F1 scores, while for aspect polarity classification (APC), we compare the accuracy and macro-F1 scores.[4]

We see that our original model variant, CLaP, performs marginally better at aspect term extraction with an average F1 score increase of 0.19, while the ablation CLaP + DK shows an average increase of 1.45 across both domains. CLaP also marginally outperforms the current state-of-the-art in aspect polarity classification by an average of 2.74 F1 score across both domains, the increase being noticeably larger in the Restaurant corpus, which could be attributed to the larger number of training samples and the higher number of aspects per sentence.

5.4.2 Hindi

For Hindi, we compare our model against the current state-of-the-art results on the dataset, which include:

- [2] introduced the ABSA dataset for Hindi used feature engineered statistical systems for this task.
- [6] studied ABSA using cross-lingual and multilingual word embedding in order to increase coverage.
- [5] provided a language-agnostic method for aspect extraction across six languages, which also included Hindi.
- [3] considered a joint modeling and an end-to-end approach to aspect-based sentiment analysis for both English and Hindi. Both these results are presented in Table 5.4.

Model	ATE F1- Akthar Dataset	APC Acc- Akthar Dataset	ATE F1- Our Dataset	APC Acc- Our Dataset
[2]	41.07	54.05	39.4	55.2
[6]	50.03	63.64	52.06	65.42
[5]	53.50	66.90	54.2	68.4
[3] Joint	43.90	76.12	55.8	78.01
[3] E2E	52.03	63.84	61.07	70.21
CLaP	**73.92**	**77.42**	**77.07**	**79.4**

Table 5.4 Comparison of our results for Hindi. We present the F1 score for aspect term extraction (ATE) and accuracy score for aspect polarity classification (APC)

Table 5.4 provides a comparison of the aspect term extraction F1 score and the aspect polarity classification accuracy scores, in keeping with the convention in existing literature. We find that our model outperforms the current models by an F1-score of 10.42 in aspect term extraction, and an accuracy score of about 1.10 points in aspect polarity classification model.

[4]Our model, CLaP is presented here with an ablation CLaP + DK, which is explained in Section 5.3.3.

5.4.3 Chinese

Domain	Model	ATE	APC	
		F1	Acc.	Macro F1
Car	ATSM-S	-	82.94	64.18
	GANN	-	83.71	77.66
	AS-Reasoner	-	85.52	79.22
	CLaP	87.19	**97.36**	**95.41**
Phone	ATSM-S	-	84.86	75.35
	GANN	-	89.17	88.16
	AS-Reasoner	-	89.17	88.02
	CLaP	90.59	**97.14**	**95.97**
Notebook	ATSM-S	-	75.59	60.09
	GANN	-	82.65	82.16
	AS-Reasoner	-	85.95	84.41
	CLaP	84.17	**94.44**	**90.92**
Camera	ATSM-S	-	82.88	72.50
	GANN	-	87.99	86.75
	AS-Reasoner	-	89.71	88.66
	CLaP	87.13	**97.5**	**94.91**

Table 5.5 Comparison of our results for Chinese. We present the F1 score for aspect term extraction (ATE) and accuracy score for aspect polarity classification (APC). "-" represents the model does not perform that task, or does not provide a score for it

For Chinese, we compare our models to the state-of-the-art for Chinese ABSA on these datasets. All the tasks provide scores only for aspect polarity classification, so we also provide the baseline score for aspect term extraction for these datasets. We compare our model against the following methods:

- **ATSM-S**: [54] provides a baseline model for attention-based sentence and aspect encoding at word-level granularities.
- **AS-Reasoner**: [45] is an attention-based reasoning architecture that introduces degrees as a method of capturing word-level aspects polarity.
- **GANN**: [44] provides a gated alternate neural network as a fine-grained analysis of sentiment using both RNNs and CNNs.

Table 5.5 presents the result of CLaP model's comparison against the current state-of-the-art for the Chinese ABSA datasets.[5] We outperform the current state-of-the-art by an average macro F1 score of 9.99 across all four domains.

[5]We also establish the neural state-of-the-art for aspect term extraction for these corpora, as current literature only provide aspect term polarity classification scores on these corpora.

English	Original	The ⟨**spicy mussels**⟩$^{+}$ are a highlight.							
	Predicted	The spicy ⟨**mussels**⟩$^{-}$ are a highlight.							
Hindi	Original	⟨**HDD**⟩$^{+}$	⟨***rale***⟩^{0}	*kaa*	*upyog*	*karke*	*surakshit*	*kartaa*	*hai*
	Predicted	⟨**HDD**	***rale***⟩$^{+}$	*kaa*	*upyog*	*karke*	*surakshit*	*kartaa*	*hai*
	Gloss	HDD	rail	of	use	doing	safe	does	is
	Translation	It uses the ⟨**HDD rail**⟩^{0} to keep [it] safe.							
Chinese	Original	机送的⟨刻件⟩^{0} 播放件毒件也不							
	Predicted	机送的⟨刻件⟩$^{+}$ 播放件毒件也不							
	Gloss	machine includes recording software and Norton antivirus are not bad							
	Translation	The ⟨**recording software**⟩^{0} that comes with the machine, the playback softwares, and the Norton antivirus software are not bad.							

Table 5.6 Some cases in which our model does not correctly identify the aspect terms and polarities in reviews. Aspect terms are bounded by ⟨ and ⟩, and are provided with polarity markers, where **+** represents positive, **-** negative, and **0** neutral. We also provide the gloss and the translation for Hindi and Chinese reviews

5.5 Discussion

In this section, we discuss our approach to the task using qualitative analyses. We use this qualitative analysis to hypothesize why our model performs well.

When analyzing the corpora used for this task, we found that aspect terms are generally nouns, which are modified either by adjectives, nominal modifiers, or descriptive predicates to copular constructions. Since all the three languages which we work with are head final languages, any modifier to the aspect noun would occur to in direct neighbourhood of the aspect term. In Hindi (which is an SOV language) and in Chinese (where the copular verb is eliminated), the descriptive copula are directly *after* the aspect term, while in English, the descriptive predicate is separated from the aspect term by just the copular verb. This close neighbourhood of aspect terms and their descriptions allows us to localize aspect-specific context for aspect polarity classification easily using word-level CNNs, without having to incorporate the direction of the dependencies as in the case of RNNs and LSTMs.

Therefore, we find that the model can make an error in aspect polarity classification stemmed from either:

- the model misidentifies the polarity of the description of a given aspect,
- the description was not in the model's context of the aspect term, or
- the model incorrectly identifies the span of the aspect term.

Table 5.6 shows examples which were incorrectly tagged by our model, one in each language. We find that the span of the aspect term includes a modifier (as in the first example), in some cases the aspect term might not include the modifier, therefore using that as the context for aspect polarity classification. In other cases, as in the third example, the aspect and its description are not in a context window that

can be easily associated. Some cases in Hindi lead to splitting in the span of the aspect term due to inconsistent code mixing, which can lead to errors in getting the correct span for aspect terms as well as their polarity classification.

5.6 Conclusion

In this chapter, we have introduced a novel approach for aspect term extraction and polarity classification: Context Localization-and-Pooling (CLaP). We use a nested sequence labeling model for this task, wherein the output of aspect term extraction is appended to the input for aspect polarity detection, both of which are sequence tagging tasks. We describe our model architecture and explore the need for various components by designing ablation studies that modify and remove various parts of the model.

Our architecture is designed in three phases. First, we encode the input sentence using BERT language representation, and use a linear layer sequence tagger to predict the aspects. We append these aspects to the input sentence, and embed this now-aspect-aware sentence using another BERT encoder. We use CNNs with variable kernel sizes to localize aspect-specific context, and use a third, joint CNN to pool the resulting outputs. This pooled representation can be used to detect the polarity of each word in the sentence, where non-aspect terms are identified with a unique label.

We find that this relatively simple model captures aspect descriptions in English, Hindi and Chinese review corpora, outperforming the current state-of-the-art in both aspect term extraction and polarity classification. We also show how using domain-specific language representations might be useful for this approach to ABSA. In the future, expanding review datasets for Hindi and Chinese, in order to train such language representations, as well as adopting a nested sequence labeling approach in models that account for long range dependencies might help further boost task performance and produce robust ABSA models.

Chapter 6

Conclusions and Future Work

The first chapter explains how most deep learning methods are predominantly data-driven, which becomes a problem for development in languages with a resource constraint. A primary example of that would be Indian Languages, especially Hindi.

We start with exploring the possibility of performing downstream tasks, like sentiment analysis in a low resource setting. Our approach was targeted at avoiding the usage of cross-lingual resources, which are hard to come by. Next, we conduct experiments to try and extract language invariant features that can be used to perform a downstream task. We use datasets from an input set of languages to learn these features and apply the learned features to a target language. The technique developed gave very promising results and achieved state-of-the-art performance on the datasets experimented with. It also displayed competitive results compared to the existing supervised and unsupervised methods that leverage cross-lingual supervision.

Next, we explore the state of another downstream task, but this time, the task chosen was Aspect Based Sentiment Analysis. The primary difference in the nature of the task compared to sentiment analysis was that the former is more sensitive to location information within the text span itself than the latter, which only requires good contextualized aggregation of the whole text.

In our exploration of the state of current research in Aspect Based Sentiment Analysis for Hindi, we identified certain fallacies with the existing gold standard dataset, which have been described in great detail in chapter 4. The same also served as our motivation to consider creating a resource that can further enhance the state of ABSA in Hindi.

We take a ground-up approach to design and develop a dataset that would serve as the new baseline dataset of experimental research of ABSA in Hindi. We elucidate the same in chapter 4, along with but not restricted to best practices and challenges of creating a dataset for ABSA. We decided to develop a corpus parallel to the gold standard dataset for English. The motivation for the same was to streamline the evaluation of models designed for the same task across a broader language spectrum provided by the resource-rich nature of English while preserving the complexities of the Hindi language. The new dataset is then evaluated with state-of-the-art context-aware models for aspect extraction that were initially designed for English. We found these models to give satisfactory results. It also helped us better

understand the reasons for the success of these models in English and failures in Hindi on a case-to-case basis. The following exercise provides the starting point for the further work that we do to come up with a Deep Learning architecture that was better suited to solve the case of ABSA in Hindi.

Finally, we hypothesize that a model that can pool the local context of an aspect in the text, along with the text's global context, is effective for learning features required to perform the task of aspect-based sentiment analysis. And based on the following hypothesis, we propose a novel architecture called CLaP (Context-Localization and Pooling).We show that a relatively simple model captures aspect descriptions in English, Hindi, and Chinese review corpora, outperforming the current state-of-the-art in both aspect term extraction and polarity classification. We also show how using domain-specific language representations might be helpful for this approach to ABSA.

We conclude thus our investigation of Sentiment Analysis in a low resource setting with the findings that for tasks that require lesser contextual information it's possible to leaveage data from multiple lanugages to extract language invariant features that give a rather good performance on such tasks. However, for tasks that require intrinsic detail and shared contexts from different parts of the text to be weighted, it's important to have a decent-sized high-quality dataset. We also conclude that it's possible to build an absa model that is language agnostic and can perform well on the task given a wide range of languages from divergent language groups.

6.1 Future Work

This thesis presents work in multiple directions that show promising results and provide direction for researchers to explore further. One of them would be to explore the possibility of extracting language invariant features as a general paradigm. The following could be possible with the advent of new-age larger transformers like GPT-3.

The findings of this thesis demand more research into understanding the CLaP framework for extracting context-aware features for various downstream tasks like question answering that need localized context information concerning the question in the target text from which the answer would be generated.

Related Publications

1. Allen J Antony, Arghya Bhattacharya, Jaipal Singh Goud and Radhika Mamidi, **Leveraging Multilingual Resources for Language Invariant Sentiment Analysis**. (Accepted at the European Association for Machine Translation 2020 (EAMT 2020)) [12]
2. Arghya Bhattacharya, Alok Debnath, Manish Shrivastava **Improving Aspect Extraction in Hindi**. (ACL 2020, Workshop on NLP in E-comm · Aug 6, 2021) [7]

Bibliography

[1] M. S. Akhtar, A. Ekbal, and P. Bhattacharyya. Aspect based sentiment analysis: category detection and sentiment classification for hindi. In *International Conference on Intelligent Text Processing and Computational Linguistics*, pages 246–257. Springer, 2016.

[2] M. S. Akhtar, A. Ekbal, and P. Bhattacharyya. Aspect based sentiment analysis in hindi: resource creation and evaluation. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, pages 2703–2709, 2016.

[3] M. S. Akhtar, T. Garg, and A. Ekbal. Multi-task learning for aspect term extraction and aspect sentiment classification. *Neurocomputing*, 2020.

[4] M. S. Akhtar, A. Kumar, A. Ekbal, and P. Bhattacharyya. A hybrid deep learning architecture for sentiment analysis. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pages 482–493, 2016.

[5] M. S. Akhtar, A. Kumar, A. Ekbal, C. Biemann, and P. Bhattacharyya. Language-agnostic model for aspect-based sentiment analysis. In *Proceedings of the 13th International Conference on Computational Semantics-Long Papers*, pages 154–164, 2019.

[6] M. S. Akhtar, P. Sawant, S. Sen, A. Ekbal, and P. Bhattacharyya. Improving word embedding coverage in less-resourced languages through multi-linguality and cross-linguality: A case study with aspect-based sentiment analysis. *ACM Transactions on Asian and Low-Resource Language Information Processing (TALLIP)*, 18(2):1–22, 2018.

[7] A. Antony, A. Bhattacharya, J. Goud, and R. Mamidi. Leveraging multilingual resources for language invariant sentiment analysis. In *Proceedings of the 22nd Annual Conference of the European Association for Machine Translation*, pages 71–79, Lisboa, Portugal, Nov. 2020. European Association for Machine Translation.

[8] M. Z. Asghar, A. Khan, S. R. Zahra, S. Ahmad, and F. M. Kundi. Aspect-based opinion mining framework using heuristic patterns. *Cluster Computing*, 22(3):7181–7199, 2019.

[9] R. K. Bakshi, N. Kaur, R. Kaur, and G. Kaur. Opinion mining and sentiment analysis. In *2016 3rd International Conference on Computing for Sustainable Global Development (INDIACom)*, pages 452–455. IEEE, 2016.

[10] L. Bao, P. Lambert, and T. Badia. Attention and lexicon regularized lstm for aspect-based sentiment analysis. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics: Student Research Workshop*, pages 253–259, 2019.

[11] N. Bel, C. H. A. Koster, and M. Villegas. Cross-lingual text categorization. In T. Koch and I. T. Sølvberg, editors, *Research and Advanced Technology for Digital Libraries*, pages 126–139, Berlin, Heidelberg, 2003. Springer Berlin Heidelberg.

[12] A. Bhattacharya, A. Debnath, and M. Shrivastava. Enhancing aspect extraction for Hindi. In *Proceedings of the 4th Workshop on e-Commerce and NLP*, pages 140–149, Online, Aug. 2021. Association for Computational Linguistics.

[13] J. Blitzer, R. McDonald, and F. Pereira. Domain adaptation with structural correspondence learning. In *Proceedings of the 2006 Conference on Empirical Methods in Natural Language Processing*, pages 120–128, Sydney, Australia, July 2006. Association for Computational Linguistics.

[14] P. Bojanowski, E. Grave, A. Joulin, and T. Mikolov. Enriching word vectors with subword information. *Transactions of the Association for Computational Linguistics*, 5:135–146, 2017.

[15] W. Che, Y. Zhao, H. Guo, Z. Su, and T. Liu. Sentence compression for aspect-based sentiment analysis. *IEEE/ACM Transactions on audio, speech, and language processing*, 23(12):2111–2124, 2015.

[16] 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*, pages 452–461, 2017.

[17] X. Chen, Y. Sun, B. Athiwaratkun, C. Cardie, and K. Weinberger. Adversarial deep averaging networks for cross-lingual sentiment classification. *Transactions of the Association for Computational Linguistics*, 6:557–570, 2018.

[18] J. A. Chevalier and D. Mayzlin. The effect of word of mouth on sales: Online book reviews. *Journal of marketing research*, 43(3):345–354, 2006.

[19] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

[20] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, 2019.

[21] K. T. Durant and M. D. Smith. Predicting the political sentiment of web log posts using supervised machine learning techniques coupled with feature selection. In O. Nasraoui, M. Spiliopoulou, J. Srivastava, B. Mobasher, and B. Masand, editors, *Advances in Web Mining and Web Usage Analysis*, pages 187–206, Berlin, Heidelberg, 2007. Springer Berlin Heidelberg.

[22] J. L. Fleiss and J. Cohen. The equivalence of weighted kappa and the intraclass correlation coefficient as measures of reliability. *Educational and psychological measurement*, 33(3):613–619, 1973.

[23] L. Frermann and A. Klementiev. Inducing document structure for aspect-based summarization. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 6263–6273, 2019.

[24] R. R. R. Gangula and R. Mamidi. Resource creation towards automated sentiment analysis in Telugu (a low resource language) and integrating multiple domain sources to enhance sentiment prediction. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan, May 2018. European Language Resources Association (ELRA).

[25] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio. Generative adversarial nets. In *Advances in neural information processing systems*, pages 2672–2680, 2014.

[26] J. S. Goud, P. Goel, A. J. Antony, and M. Shrivastava. Leveraging multilingual resources for open-domain event detection. In *Proceedings 15th Joint ACL-ISO Workshop on Interoperable Semantic Annotation*, pages 76–82, 2019.

[27] E. Grave, P. Bojanowski, P. Gupta, A. Joulin, and T. Mikolov. Learning word vectors for 157 languages. In *Proceedings of the International Conference on Language Resources and Evaluation (LREC 2018)*, 2018.

[28] S. Gupta and N. Khade. Bert based multilingual machine comprehension in english and hindi. *arXiv preprint arXiv:2006.01432*, 2020.

[29] H. Hamdan, P. Bellot, and F. Bechet. Lsislif: Crf and logistic regression for opinion target extraction and sentiment polarity analysis. In *Proceedings of the 9th international workshop on semantic evaluation (SemEval 2015)*, pages 753–758, 2015.

[30] M. Hoang, O. A. Bihorac, and J. Rouces. Aspect-based sentiment analysis using bert. *NoDaLiDa 2019*, page 187, 2019.

[31] S. Hochreiter and J. Schmidhuber. Long Short-Term Memory. *Neural Computation*, 9(8):1735–1780, 11 1997.

[32] Z. Huang, W. Xu, and K. Yu. Bidirectional lstm-crf models for sequence tagging. *arXiv preprint arXiv:1508.01991*, 2015.

[33] M. Iyyer, V. Manjunatha, J. Boyd-Graber, and H. Daumé III. Deep unordered composition rivals syntactic methods for text classification. 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)*, pages 1681–1691, Beijing, China, July 2015. Association for Computational Linguistics.

[34] T. Joachims. Text categorization with support vector machines. *Proc. European Conf. Machine Learning (ECML'98)*, 01 1998.

[35] H. Kanayama, T. Nasukawa, and H. Watanabe. Deeper sentiment analysis using machine translation technology. In *COLING 2004: Proceedings of the 20th International Conference on Computational Linguistics*, pages 494–500, Geneva, Switzerland, aug 23–aug 27 2004. COLING.

[36] P. Keung, Y. Lu, G. Szarvas, and N. A. Smith. The multilingual amazon reviews corpus. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing*, 2020.

[37] Y. Kim. Convolutional neural networks for sentence classification. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1746–1751, Doha, Qatar, Oct. 2014. Association for Computational Linguistics.

[38] D. P. Kingma and J. Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[39] X. Li, L. Bing, P. Li, and W. Lam. A unified model for opinion target extraction and target sentiment prediction. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 6714–6721, 2019.

[40] X. Li, L. Bing, P. Li, W. Lam, and Z. Yang. Aspect term extraction with history attention and selective transformation. In *Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence, IJCAI-18*, pages 4194–4200. International Joint Conferences on Artificial Intelligence Organization, 7 2018.

[41] X. Li, L. Bing, W. Zhang, and W. Lam. Exploiting bert for end-to-end aspect-based sentiment analysis. *W-NUT 2019*, page 34, 2019.

[42] X. Li and W. Lam. Deep multi-task learning for aspect term extraction with memory interaction. In *Proceedings of the 2017 conference on empirical methods in natural language processing*, pages 2886–2892, 2017.

[43] C.-Y. Lin and F. J. Och. Automatic evaluation of machine translation quality using longest common subsequence and skip-bigram statistics. In *Proceedings of the 42nd Annual Meeting of the Association for Computational Linguistics (ACL-04)*, pages 605–612, 2004.

[44] N. Liu and B. Shen. Aspect-based sentiment analysis with gated alternate neural network. *Knowledge-Based Systems*, 188:105010, 2020.

[45] N. Liu, B. Shen, Z. Zhang, Z. Zhang, and K. Mi. Attention-based sentiment reasoner for aspect-based sentiment analysis. *Human-centric Computing and Information Sciences*, 9(1):35, 2019.

[46] P. Liu, S. Joty, and H. Meng. Fine-grained opinion mining with recurrent neural networks and word embeddings. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 1433–1443, 2015.

[47] H. Luo, T. Li, B. Liu, and J. Zhang. Doer: Dual cross-shared rnn for aspect term-polarity co-extraction, 2019.

[48] D. Ma, S. Li, F. Wu, X. Xie, and H. Wang. Exploring sequence-to-sequence learning in aspect term extraction. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3538–3547, 2019.

[49] R. Mihalcea, C. Banea, and J. Wiebe. Learning multilingual subjective language via cross-lingual projections. In *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics*, pages 976–983, Prague, Czech Republic, June 2007. Association for Computational Linguistics.

[50] T. Mikolov, E. Grave, P. Bojanowski, C. Puhrsch, and A. Joulin. Advances in pre-training distributed word representations. In *Proceedings of the International Conference on Language Resources and Evaluation (LREC 2018)*, 2018.

[51] P. Nakov, T. Zesch, D. Cer, and D. Jurgens. Proceedings of the 9th international workshop on semantic evaluation (semeval 2015). In *Proceedings of the 9th International Workshop on Semantic Evaluation (SemEval 2015)*, 2015.

[52] H. Nguyen and K. Shirai. A joint model of term extraction and polarity classification for aspect-based sentiment analysis. In *2018 10th International Conference on Knowledge and Systems Engineering (KSE)*, pages 323–328. IEEE, 2018.

[53] X. Ouyang, P. Zhou, C. H. Li, and L. Liu. Sentiment analysis using convolutional neural network. In *2015 IEEE international conference on computer and information technology; ubiquitous computing and communications; dependable, autonomic and secure computing; pervasive intelligence and computing*, pages 2359–2364. IEEE, 2015.

[54] H. Peng, Y. Ma, Y. Li, and E. Cambria. Learning multi-grained aspect target sequence for chinese sentiment analysis. *Knowledge-Based Systems*, 148:167–176, 2018.

[55] T. Pires, E. Schlinger, and D. Garrette. How multilingual is multilingual bert? In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4996–5001, 2019.

[56] M. Pontiki, D. Galanis, H. Papageorgiou, I. Androutsopoulos, S. Manandhar, A.-S. Mohammad, M. Al-Ayyoub, Y. Zhao, B. Qin, O. De Clercq, et al. Semeval-2016 task 5: Aspect based sentiment analysis. In *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016)*, pages 19–30, 2016.

[57] M. Pontiki, D. Galanis, H. Papageorgiou, S. Manandhar, and I. Androutsopoulos. Semeval-2015 task 12: Aspect based sentiment analysis. In *Proceedings of the 9th international workshop on semantic evaluation (SemEval 2015)*, pages 486–495, 2015.

[58] M. Pontiki, D. Galanis, J. Pavlopoulos, H. Papageorgiou, I. Androutsopoulos, and S. Manandhar. SemEval-2014 task 4: Aspect based sentiment analysis. In *Proceedings of the 8th International Workshop on Semantic Evaluation (SemEval 2014)*, pages 27–35, Dublin, Ireland, Aug. 2014. Association for Computational Linguistics.

[59] S. Prabhu, P. Goel, A. Debnath, and M. Shrivastava. A language invariant neural method for timeml event detection. In *Proceedings of International Conference on NLP (ICON)*, 2019.

[60] W. Quan, Z. Chen, J. Gao, and X. T. Hu. Comparative study of cnn and lstm based attention neural networks for aspect-level opinion mining. In *2018 IEEE International Conference on Big Data (Big Data)*, pages 2141–2150. IEEE, 2018.

[61] L. A. Ramshaw and M. P. Marcus. Text chunking using transformation-based learning. In *Natural language processing using very large corpora*, pages 157–176. Springer, 1999.

[62] Y. R. Regatte, R. R. R. Gangula, and R. Mamidi. Dataset creation and evaluation of aspect based sentiment analysis in telugu, a low resource language. In *Proceedings of The 12th Language Resources and Evaluation Conference*, pages 5017–5024, 2020.

[63] N. Reimers and I. Gurevych. Reporting score distributions makes a difference: Performance study of lstm-networks for sequence tagging. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 338–348, 2017.

[64] S. Rosenthal, N. Farra, and P. Nakov. Semeval-2017 task 4: Sentiment analysis in twitter. In *Proceedings of the 11th international workshop on semantic evaluation (SemEval-2017)*, pages 502–518, 2017.

[65] I. San Vicente, X. Saralegi, and R. Agerri. Elixa: A modular and flexible absa platform. In *Proceedings of the 9th International Workshop on Semantic Evaluation (SemEval 2015)*, pages 748–752, 2015.

[66] R. Socher, A. Perelygin, J. Wu, J. Chuang, C. D. Manning, A. Ng, and C. Potts. Recursive deep models for semantic compositionality over a sentiment treebank. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 1631–1642, Seattle, Washington, USA, Oct. 2013. Association for Computational Linguistics.

[67] Y. Song, J. Wang, T. Jiang, Z. Liu, and Y. Rao. Targeted sentiment classification with attentional encoder network. In *International Conference on Artificial Neural Networks*, pages 93–103. Springer, 2019.

[68] C. Sun, L. Huang, and X. Qiu. Utilizing bert for aspect-based sentiment analysis via constructing auxiliary sentence. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 380–385, 2019.

[69] K. S. Tai, R. Socher, and C. D. Manning. Improved semantic representations from tree-structured long short-term memory networks. 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)*, pages 1556–1566, Beijing, China, July 2015. Association for Computational Linguistics.

[70] Z. Toh and J. Su. Nlangp at semeval-2016 task 5: Improving aspect based sentiment analysis using neural network features. In *Proceedings of the 10th international workshop on semantic evaluation (SemEval-2016)*, pages 282–288, 2016.

[71] Z. Toh and W. Wang. Dlirec: Aspect term extraction and term polarity classification system. *SemEval 2014*, page 235, 2014.

[72] X. Wan. Co-training for cross-lingual sentiment classification. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP*, pages 235–243, Suntec, Singapore, Aug. 2009. Association for Computational Linguistics.

[73] H. Xu, B. Liu, L. Shu, and S. Y. Philip. Double embeddings and cnn-based sequence labeling for aspect extraction. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 592–598, 2018.

[74] H. Xu, B. Liu, L. Shu, and S. Y. Philip. Bert post-training for review reading comprehension and aspect-based sentiment analysis. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 2324–2335, 2019.

[75] R. Xu and Y. Yang. Cross-lingual distillation for text classification. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1415–1425, Vancouver, Canada, July 2017. Association for Computational Linguistics.

[76] W. Xue and T. Li. Aspect based sentiment analysis with gated convolutional networks. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2514–2523, 2018.

[77] D. Yarowsky and G. Ngai. Inducing multilingual POS taggers and NP bracketers via robust projection across aligned corpora. In *Second Meeting of the North American Chapter of the Association for Computational Linguistics*, 2001.