PREDICTING SENTIMENT-MENTION ASSOCIATIONS IN PRODUCT REVIEWS

by

VISHWAS VASWANI

B.E., Rajiv Gandhi Technical University, India, 2009

A THESIS

submitted in partial fulfillment of the
requirements for the degree

MASTER OF SCIENCE

Department of Computing and Information Sciences
College of Engineering

KANSAS STATE UNIVERSITY
Manhattan, Kansas
2012

Approved by:

Major Professor
Dr. Doina Caragea

Abstract

With the rising trend in social networking, more people express their opinions on the web. As a consequence, there has been an increase in the number of blogs where people write reviews about the products they buy or services they experience. These reviews can be very helpful to other potential customers who want to know the pros and cons of a product, and also to manufacturers who want to get feedback from customers about their products. Sentiment analysis of online data (such as review blogs) is a rapidly growing field of research in Machine Learning, which can leverage online reviews and quickly extract the sentiment of a whole blog. The accuracy of a sentiment analyzer relies heavily on correctly identifying associations between a sentiment (opinion) word and the targeted mention (token or object) in blog sentences.

In this work, we focus on the task of automatically identifying sentiment-mention associations, in other words, we identify the target mention that is associated with a sentiment word in a sentence. Support Vector Machines (SVM), a supervised machine learning algorithm, was used to learn classifiers for this task. Syntactic and semantic features extracted from sentences were used as input to the SVM algorithm. The dataset used in the work has reviews from car and camera domain.

The work is divided into two phases. In the first phase, we learned domain specific classifiers for the car and camera domains, respectively. To further improve the predictions of the domain specific classifiers we investigated the use of transfer learning techniques in the second phase. More precisely, the goal was to use knowledge from a source domain to improve predictions for a target domain. We considered two transfer learning approaches: a feature level fusion approach and a classifier level fusion approach.

Experimental results show that transfer learning can help to improve the predictions made using the domain specific classifier approach. While both the feature level and classifier level fusion approaches were shown to improve the prediction accuracy, the classifier level fusion approach gave better results.

Table of Contents

List of Figures

List of Tables

Acknowledgments

There is a Chinese Proverb:

"*When eating bamboo sprouts, remember the man who planted them.*"

With this proverb in mind, I would like to thank everyone who helped me in accomplishing this work and providing their invaluable suggestions, guidance, insights and support.

First and foremost, I would like to thank my advisor, Dr. Doina Caragea. She has been the person who supported and encouraged me throughout my M.S. at Kansas State University. Successful completion of this thesis is the result of her timely comments, suggestions and patience, which helped me stay on course. I appreciate the time and effort she spent in thoroughly reviewing this work not once, but multiple times. She has been a great mentor who guided me not only through my academic issues, but also my personal and job related issues.

I would also like to thank Dr. Gurdip Singh for being part of my thesis committee and for the knowledge that he provided in the operating systems course. I offer my gratitude to Dr. Torben Amtoft for providing me with an understanding of algorithms and databases in his classes, which helped me in implementing this work.

I thank the Bioinformatics Center at KSU and Dr. Susan Brown for giving me part-time working opportunity during my M.S. studies. This job provided me with a lot of experience and ability to face and overcome challenges.

I thank my friends and colleagues, Swapnil, Ana, Sandeep, Sri, Karthik and Rohit for helping me in this work and during my Masters. And last, but not the least, I would like to thank my parents, Rajendra Vaswani and Ekta Vaswani, and my brother, Rishab Vaswani, for their constant support and faith in me. I deeply appreciate the sacrifices they made for my future.

Chapter 1

Introduction

With the increase of social networking and blogging websites, there has been a sudden outburst of information on the web. Much of the information that is available on these websites consists of untapped user opinions in the form of reviews, recommendations, ratings etc. Customers who are in the marketplace to purchase a product or service can conduct research about the product and the company through these opinions. Thus, online social media websites can have a deep effect on the way customer does his purchasing.

The raw data that is available on *blogs* (e.g. LiveJournal [1], Google blogs), *e-commerce sites* (e.g. Amazon [2], eBay [3]), *review sites* (e.g. CNET [4], PC Magazine [5]), *discussion forums* (e.g. Craigslist forums [6]) etc., is of extreme importance from both individual's and business's prospectives [Ganesan and Kim, 2008]. The individual (customer) wants to know the pros and cons of a product and also to get some information about the manufacturer. The company (manufacturer) might want to know whether customers liked or disliked a product, the complaints and suggestions about that product.

An overwhelming amount of data is already available on the web and with new information constantly being added, it becomes difficult for an individual or a company to go

[1] *http://www.livejournal.com/*
[2] *http://www.amazon.com/*
[3] *http://www.ebay.com/*
[4] *http://www.cnet.com/*
[5] *http://www.pcmag.com/*
[6] *http://www.craigslist.org/about/sites*

through each bit. **Automated Sentiment Analysis** emerged as a new field in computer science, a field which could help customers and companies in analyzing online opinions and reviews. In other words, **Automated Sentiment Analysis** can be seen as a task in machine learning, where we train a classification model, using *Machine Learning* [Mitchell, 1997] and *Natural Language Processing (NLP)* techniques [Manning and Schutze, 1999], to identify and classify sentiments in online content.

In Section 1.1 we will discuss sentiment analysis in more details and also its types. Then, in Section 1.2, we will discuss some application of sentiment analysis. Section 1.3 explains in brief the objective of this work and the approach used to address it.

1.1 Sentiment Analysis

Sentiment analysis refers to the process of identifying and extracting the opinion or sentiment of a person towards an object or a topic. These opinions or sentiments are available abundantly on the web in the form of text. Machine Learning can be used to conduct sentiment analysis on the web. Machine Learning techniques can also be used to train a classification model that identifies sentiment words in a sentence and classifies the sentiments according to their polarity. As datasets used in sentiment analysis comprise of opinions from the social media websites (mostly blogs), they do not have any fixed format. Below are listed some writing styles that bloggers use, which make the task of sentiment analysis challenging:

- Opinions on the web can be in the form of plain text summary, where there is no separation between positive and negative reviews.

  For example: *"I loved the exteriors of new 2012 Honda CRV that debuted in the Detroit auto show this year. But the interiors still felt cheap."*

- They might include a nice structure, where positive and negative features are separated in points, but a lexical structure (sentence form) might be missing, which makes it difficult to get the semantics of the opinion mentioned.

For example: *"Review 2012 Honda CRV:*

Pros: Nice exterior.

Cons: Cheap interiors."

- Identification of active or passive voice in an opinion with plain text poses another challenge.

  For example: *Active voice: "Yesterday, I drove the peppy Volkswagen golf."*

  Passive voice: "The peppy Volkswagen golf was driven by me yesterday."

- A sentence might be objective (having no sentiment) or subjective (opinionated).

  For example: *Subjective: "Yesterday, I drove the peppy Volkswagen golf."*

  Objective: "The Volkswagen golf starts at $25,000."

- The sentence could contain direct opinions or comparisons between two products.

  For example: *Direct: "The Nissan Cube is an ugly car. Honda Fit is good-looking."*

  Comparison: "Unlike the ugly Nissan Cube, Honda Fit is a far good-looking car."

All these factors are common to documents in sentiment analysis and make the problem difficult.

Apart from these challenges, *Sentiment Analysis* includes various subtasks that need to be addressed in order to precisely predict the sentiment. Following are some subtasks in Sentiment Analysis:

- *Sentiment Identification*, where the sentiment words in a sentence (if present) are tagged.

- *Sentiment-Mention Association*, where all possible evaluative entities or products (mentions) are tagged and, amongst these, the mention that is targeted by a sentiment word is identified. A mention can be any physical entity or object, for example

"Honda Accord" is a mention of type vehicle and "engine" is a mention of type vehicle part. And a sentiment is an expression of thought or opinion towards a person or thing. (These terms are explained in more details in Section 3.1.)

- *Sentiment Classification*, where the sentences are identified as positive, negative or neutral. Some sentences could also contain both positive and negative sentiment words.

Sentiment classification can be performed at three levels:

- Sentence level - the goal is to perform classification of a sentence as subjective or objective and positive or negative. A sentence can be neutral if it has more than one sentiment expression of opposite polarity.

  For example: *"This is a beautiful car."*

  The above sentence is subjective as it has at least one sentiment in it. Here "beautiful" is the only sentiment in the sentence which is associated with "car". As polarity of "beautiful" is positive, the polarity of the sentence is also positive.

- Document level - the goal is to perform classification of the whole text document as positive, negative or neutral. The polarity of the document is determined by the overall sentiment orientation of the document.

- Feature level - the goal is to perform classification of each object feature (mention) in the sentence independently. Here also, the polarity can be positive, negative or neutral. The task is to identify the features (mentions), determine the sentiments associated with them and then assign a polarity to each of the sentiment-mention pairs.

  For example: *"This is a beautiful car with luxurious interiors."*

  Here we have two sentiment words "beautiful" and "luxurious", two mentions "car" and "interiors" where "car" is an object and "interiors" is a feature of the object. Also, "beautiful" is associated with "car" and "luxurious" is associated with "interiors". Polarity of both "beautiful" and "luxurious" is positive, hence the polarity of both mentions "car" and "interiors" is also positive.

1.2 Applications of Sentiment Analysis

The process of sentiment analysis can be seen as the transformation of the user's or customer's implicit, untapped emotions and opinions about a product or service into valuable information, which is almost free of cost. It is important both for businesses/organizations as well as individuals. The organizations will be interested in reviews of their products, market intelligence, survey on a topic etc. It could also help them with ads placement as they can suggest a similar product by placing an advertisement of the new product if the customer liked the previous product he purchased. For individuals, this analysis could be useful when they want to purchase a product, use a service or want to make a decision about any topic. The sentiment analysis provides them with input about the product or the service, in other words it provides them with the overall sentiment polarity assigned to the feature of the product they are about to purchase, hence saving a ton of their valuable time.

The industries where sentiment analysis is being used or can be used includes automobiles, entertainment, electronics, books, health, food, travel, finance, online services (including gaming services, tools, maps, online stores), fashion etc.

1.3 Overview of the Thesis

In this work, we focus on the subtask of *Sentiment-Mention Association* in sentiment analysis. In other words, we work on the problem of getting the correct associations between sentiment expressions and possible mentions in a sentence. We use a supervised machine learning algorithm, *Support Vector Machines (SVM)* [Cortes and Vapnik, 1995] to train a classifier using the semantic and syntactic features from the sentence. A supervised learning framework is a technique that uses the labeled instances to learn a model that is then used to predict the class of new unseen instances. Support vector machine is a supervised learning approach that considers each instance as a point in a multi-dimensional space and builds a hyperplane to separate the data into two classes. It is mostly used for binary classification and regression [Mitchell, 1997].

The dataset used in this work is the *JDPA sentiment corpus* [Kessler and Nicolov, 2010] that has customer reviews for *car* and *camera* domains.

This work is divided in two phases. In the first phase, we identify the mentions and sentiment words from sentences in both car and camera reviews. Each mention in a sentence can potentially form a pair with each sentiment in the sentence and vise versa.

For example: There are 2 sentiments ("beautiful", "luxurious") and 2 mentions ("car", "interiors"). There will be 4 possible sentiment-mention pairs: (beautiful, car), (beautiful, interiors), (luxurious, car) and (luxurious, interiors).

Syntactic and semantic features are extracted for these instances of possible sentiment-mention pairs. These features have been used previously in related work by Kessler et al. [2009]. We used SVM to train domain specific classifiers for each domains separately using labeled instances from the respective domains, and then predict the classes for unlabeled instances from the same domain.

Given the small amount of labeled data available, we also explored transfer learning (aka Cross-Domain) approaches in the second phase of our work. The transfer learning approach uses the previously learned knowledge from a task and applies it to a different but related task. We compared the results for two different transfer learning approaches. The first approach is *Feature Level Fusion approach* [Li and Zong, 2008] and the second one is *Classifier Level Fusion approach* [Li and Zong, 2008]. The approach and implementation details are discussed in Chapter 3 and Chapter 4, respectively.

To summarize, the rest of the document is organized as follows: Chapter 2 describes the related works that have been conducted in the sentimental analysis field. Chapter 3 describes the problem addressed and also the approaches we used to solve the problem. In Chapter 4, we discuss the experimental setup and explain the implementation details. Chapter 4 also lists the experiments we conducted. The results of these experiments are then shown and discussed in Chapter 5. The conclusion and the future work are included in Chapter 6.

Chapter 2

Related Work

As the number of social networking and blog sites has increased in the past decade, there has been considerable attention given to the task of sentiment analysis recently. The problem of associating objects (sentiment expressions) with targets has also been attempted by many researchers. This chapter provides a review of the related works conducted previously. One of the approaches that attempted to address the problem of getting associations between targets and sentiments was a *proximity based approach* [Nicolov et al., 2008]. The authors tried to semantically relate a token (mention) and the closest sentiment. The approach worked well for some sentences but not for all, as in some cases the closest sentiment-mention pairs were not semantically related. An example of such a case is provided below: "Beast! Yes, that's the first word that comes in mind when I hear the *engine* of this super-luxurious *Roles Royce*".

Here "Beast", sentiment, should be associated with "engine" but according to the proximity based approach "super-luxurious" (also a sentiment) is only 2 tokens away ("of" and "this") from it as compared to "Beast" which is 13 tokens away. Hence, it will incorrectly associate "super-luxurious" with "engine".

The *n-token window* was another approach suggested in [Nicolov et al., 2008]. In this approach the mentions occurring in a window of size n from the sentiment expression are considered as the target mentions. This approach could also result in incorrect predictions for sentiment-mention pairs, which are far away from each other in a sentence. The sentence below is provided as an example for such a case.

"The *1945 Ford Coupe* owned by *my* school friend who lives in *Florida*, is a vintage".

Here the sentiment expression "vintage" targets the mention "Ford Coupe" which is about 11 tokens far from it. Depending on the window size, it might happen that the target mention falls outside the window. Also, there are 2 more mentions "my" and "Florida" which are closer to the sentiment expression and could be predicted as a possible target mention.

Work by Bloom et al. [2007] links the target to the attitude in the appraisal expression framework. An appraisal expression as defined in [Bloom et al., 2007] is "a textual unit expressing an evaluative stance towards some target" and it is similar to the sentiment expression in our work. The appraisal expression consists of a source, an attitude (sentiment) and a target (mention). The linking of the attitude to the target is done by the dependency representation of each sentence and a ranked list of these representations is obtained. Bethard [2004] also tried to link the targets with the "opinion denoting verbs". An example of such a sentence can be:

"I believe in the *system*".

Here "believe" is a verb but it also expresses opinion about the system. The authors only considered the "opinion verbs" that were prepositions.

Kobayashi et al. [2006] finds the association between the "aspect" (mention) and "evaluation" (sentiment) in a Japanese corpus of product reviews. It uses a method by Iida et al. [2003], where the aspects are linked to the evaluations, based on comparing two candidate aspects from the sentence and ranking them. The aspect with highest likelihood is then compared to the other unseen possible candidate aspect and this is carried on for all the aspects until a final winner at the end is obtained. This approach uses the syntactic and semantic features from the sentences in the reviews.

Our work is closely related to the work by Kessler et al. [2009]. The authors of this paper first used syntactic and semantic features from reviews in the JDPA sentiment corpus [Kessler and Nicolov, 2010] to get the associations between sentiments and mentions in a sentence.

Kessler et al. [2009] analyzed *JDPA sentiment dataset* and found some interesting facts about it. Assuming that the targets lie on either the left or the right of the sentiment expression will not be a good way to predict their association, as 51% of the targets appeared on the left and 49% on right. They also noticed that 91% of the target mentions are in the same sentence as the sentiment expression, so breaking the reviews in sentences and analyzing them can guarantee high performance accuracy. To address the problem, they used the RankSVM [Joachims, 2006] (a variant of SVM) that ranks the possible sentiment-mention associations based on the likelihood score of the association.

Our work is an extension to the work by Kessler et al. [2009], where we not only learned individual classifiers for both the car and camera domains from JDPA datasets, but we also learned cross-domain classifiers, using two approaches. First, we merged the features from both datasets to learn a combined classifier for both domains (Feature Level Fusion Approach). Second, learned separate classifiers from the source and target domains, and combined the classifiers to further enhance the prediction for the target domain (Classifier Level Fusion Approach). Experiments were carried out considering car as target and camera as source domain, and vice versa.

The classifier level fusion approach also known as an ensemble of classifiers has been studied in many works including [Li and Zong, 2008] and [Aue et al., 2005]. These studies show that classifier fusion gives better results when cross-domain or transfer learning is conducted in sentiment analysis tasks.

There are other approaches that have been used in the sentiment analysis e.g., a tree based approach in [Jiang et al., 2010], a spectral feature alignment approach in [Pan et al., 2010], etc.

Chapter 3

Problem Definition and Approaches

This chapter describes the task of finding associations between target mentions and sentiment expressions in a sentence from the product reviews available on the web. It also explains the feature set, classifiers and approaches used in this work. The problem of determining which sentiment word in the sentence is related to which mention is the main task for performing opinion mining for any data or reviews. We address this problem by using two different approaches: feature level fusion approach and classifier level fusion approach.

We begin this chapter by providing a detailed problem definition with the help of some examples in Section 3.1. In Section 3.2, we describe the feature set that we have used in this work and also how the features are used to effectively train a classifier for the problem stated. Section 3.3 describes the two phases of our work. In the first phase we explain in brief, the working of Support Vector Machines (SVM) [Cortes and Vapnik, 1995] which we have used to learn the domain specific classifiers. Section 3.3 also explains the second phase of our work where we used the transfer learning approaches to improve the accuracy of the classifier. Transfer Learning can help improve the classifier predictions in a target domain having small amount of labeled data or instances available. To further improve the classifier's predictions for the target domain, we include some more training data from source domain. In our case, car dataset was considered as the source domain and camera dataset was considered as the target domain, and vice versa. We use two different approaches to perform transfer learning, these approaches are also described in Section 3.3.

3.1 Predicting Mention-Sentiment Associations

As we described above, to accurately predict the positive or negative polarity of a mention in a review, first we need to decide which sentiment expression is associated with that mention. Prior to introducing the problem definition, we explain the basic terms that have been or will be used frequently in this work.

A **mention** in a sentence is any concrete entity or a part or feature of that entity that we can evaluate. For example, in our dataset we have car reviews. Here, a car like *Honda Accord*, a car part like *Engine*, an organization like *Mercedes-Benz*, a year like *1985*, a place like *Germany* etc. can be called a mention. Any person name or pronouns like *I, my, you* etc. are also considered mentions. For each mention we also have a mention class or type associated with them. Types can include *Vehicles, Time, Organizations, Car Part, Person, Places, Vehicles.SUV* etc. The next term that is relevant to this work is **sentiment** which is an expression of thought or a feeling or opinion towards a person or a thing. The evaluative expression containing the sentiment and targeting the mention in a sentence is known as a **sentiment expression**.

Now, we give examples of various types of sentences that can be found in the customer reviews.

- The car has <u>good</u> *interiors.*

  In this example the mention is "interiors" which is a car feature and the sentiment associated with it is "good".

- Both *Accord* and *CRV* are <u>best-selling</u> cars in their own segment.

  In this sentence we have two mentions "Accord" and "CRV" associated with one sentiment expression "best-selling".

- *I* <u>liked</u> the *Accord* with <u>powerful</u> *engine*, more than the *Camry.*

  In this sentence there are four mentions "I", "Accord", "Camry" and "engine" (italics), and two sentiment words "liked" and "powerful" (underlined). In principle, the sentiment word "liked" can now be associated with all four mentions in the sentence, forming 4 possible sentiment-mention pairs. Similarly the sentiment word "powerful" also forms four possible sentiment-mention pairs. But, ("liked", "Accord") and ("powerful", "engine") are the only two pairs out of eight, that are correct associations.

We need to find out from sentences like those shown above, which sentiment expression targets which mention. There are some simple approaches (*proximity based approach* [Nicolov et al., 2008] and *n-token window approach* [Nicolov et al., 2008]) that seem attractive, but as mentioned in the related work section, there are several types of sentences that these approaches cannot handle. Therefore, in our work, we use a machine learning approach, as opposed to simply proximity or window based approaches. In the next section, we describe the features used with our machine learning approach.

3.2 Feature Set

For the first phase of this work, we learned domain-specific classifiers using the syntactic and semantic features extracted from sentences in the reviews from the dataset. These domain-specific classifiers were then used to predict the sentiment-mention association for the respective domains.

The syntactic features in a sentence are the rules that are used to build or construct a sentence in natural languages [Ganesan and Kim, 2008]. The words that are the building blocks of a sentence in any natural language are further divided into categories called *Parts of Speech* or POS. Some examples of POS categories are noun, verb, adjective, pronoun, determiner etc. These syntactic features are used to predict the association between the sentiment expressions and mentions because these syntactic features help in extracting the structure of the sentence which in turn can determine the type of relation between sentiment and candidate mention.

Below is a sentence from the car domain reviews in the dataset. We will use this sentence as an example in rest of this work.

Example:

I liked the *Accord* with powerful *engine*, more than the *Camry*.

The syntactic features that we have used are listed below. We use the sentiment-mention pair "liked" and "engine" from the example to illustrate the features.

- *Lexical Distance*: It represents the number of tokens (words) between the sentiment expression and the mention.

  In our example: 4 (the number of words or tokens between "liked" and "engine" is 4)

- *Lexical Path*: It is the set of tokens between the mention and the sentiment expression.

  In our example: *"the Accord with powerful"* (these are the words between "liked" and "engine").

- *Lexical Stem Path*: It is the set of stems (base or root form) of the tokens in the lexical path.

  In our example: *"the Accord with power"* ("powerful" is derived from "power", we replace it).

- *Lexical POS Path*: It is the set of part-of-speech (POS) tags or the linguistic category for the tokens in the lexical path.

  The POS tagging for our example sentence is:

  I/PRP liked/VBD the/DT Accord/NNP with/IN powerful/JJ engine/NN ,/, more/RBR than/IN the/DT Camry/NNP ./.

  As in our example the lexical path is "the Accord with powerful" the lexical POS path translates to *DT NNP IN JJ*, where DT - Determiner, NNP - Proper Noun singular, IN - preposition, JJ - Adjective (these are the POS categories for the token in Lexical Path labeled by the Stanford Dependency Parser [Marneffe et al., 2006]).

- *POS relation*: It is given by the POS categories (as labeled by the Stanford Dependency Parser) of the sentiment-mention pair whose features are being extracted.

  In our example: *VBD-NN* where VBD - verb past tense, NN - Noun singular. The POS relation represents the relation or dependency between the sentiment and the mention. In our example the sentiment is "liked" which is a verb in past tense and the mention is "engine", which is a singular noun.

The semantics of a sentence in natural language refers to its meaning and implies extracting the relation between the words or phrases of a sentence [Ganesan and Kim, 2008]. The semantic analysis of a sentence is required as it can derive the relations or dependencies between the words in the sentence. Some of the semantic features we have used in this work are:

- *Dependency Path*: It is the shortest dependency path between the sentiment expression and the mention where both are treated as nodes of the graph whose links or connections are determined by the dependencies returned by the Stanford Dependency parser [Marneffe et al., 2006].

  In our example: *"prep pobj"* where prep stands for prepositional modifier, pobj stands for object of preposition.

  As we can see in Figure 3.1, the nodes are the tokens or words of the sentence in our example. The directional arrows in the graph depicts the relation or dependency between the tokens. The dependency path between "liked" and "engine" is "prep pobj".

- *Sentiment Expression in Path*: It represents the number of sentiment expressions in the path.

  In our example: 0 (there is no sentiment word in the shortest dependency path obtained from the dependency graph).

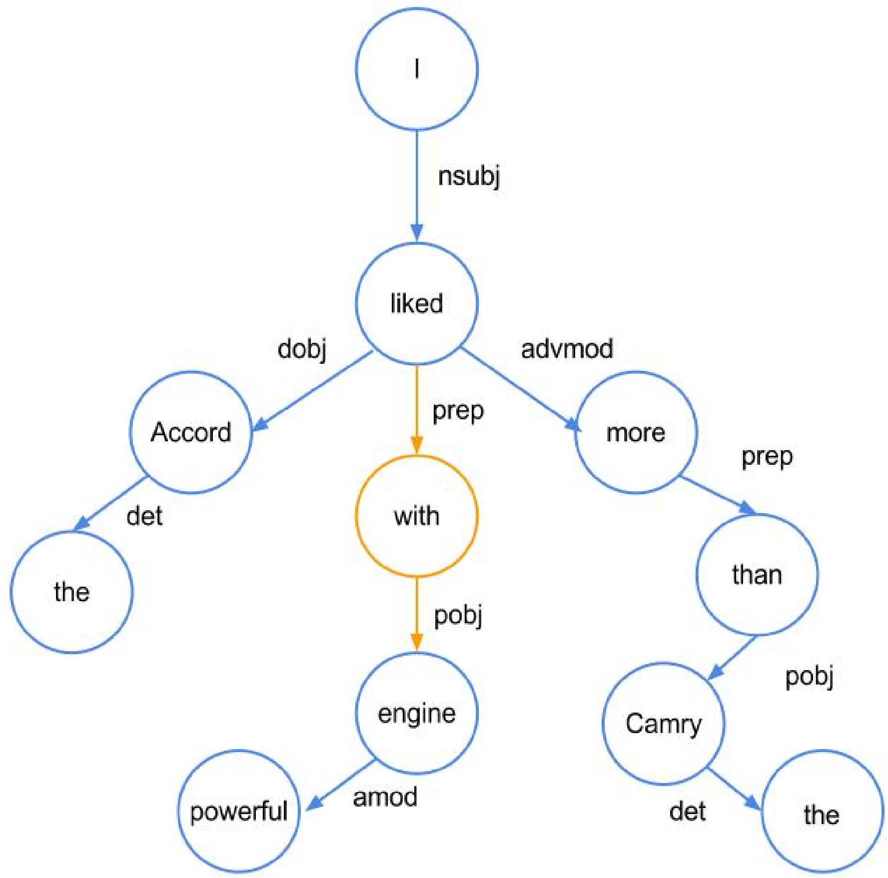

Figure 3.1: *Dependency Graph*

- <u>*Mention in Path*</u>: It represents the number of mentions in the path.

  In our example: 0 (there is no mention in the shortest dependency path obtained from the dependency graph).

- <u>*Mention Type*</u>: It represents the semantic type of the mention.

  In our example: *Vehicles.part* (the mention "engine", is a part of vehicle. Hence the class - "Vehicles.part").

- <u>*Stem Dependency Path*</u>: It is the stem (base or root form) of the sentiment expression concatenated to the dependency path.

In our example: *like prep pobj* (the stem or root of "liked" is "like" which is concatenated to the dependency path "prep pobj").

All of the features except Sentiment Expression in Path, Mention in Path are encoded as binary features (0/1 based on their appearance in the sentence).

Example: Sentiment Expression in Path - 0, Mention in Path - 0.
For our example sentence, the feature vector is {4, (the, Accord, with, powerful), (the, Accord, with, power), (DT, NNP, IN, JJ), (VBD-NN), (prep, pobj), 0, 0, (Vehicles.part), (like, prep, pobj)}.

3.3 Approaches

3.3.1 Learning Domain-Specific Classifiers

In the first phase of this work we extract the features that were discussed above and then train a classifier for each domain (car and camera) independently. We test each classifier on the test data from its respective domain. The algorithm used in this phase was *Support Vector Machine (SVM)* which is a supervised machine learning algorithm. SVM takes labeled examples as an input and learns a model based on those inputs which then classifies new examples as belonging to one class or another. SVM considers each example or instance as a data point in a high dimensional space. It then builds a hyperplane that can divide or classify those points with high accuracy. The accuracy of classification of an instance (data point) is determined by the distance to the separating hyperplane. The larger the distance, the better is the accuracy and vice-versa [Cortes and Vapnik, 1995].

3.3.2 Learning Cross-Domain Classifiers

In the second phase of the work, we aim to improve the predictions for the associations that were generated in the first phase. Transfer learning was a good choice for us to improve the predictions because we can make use of the labeled data available in a source domain to train the target domain classifier where labeled data is scarce. Formally speaking, transfer learning is a technique in which we acquire knowledge by solving a task on one domain and then applying this previously learned knowledge to solve the task on a different but related domain [Pan and Yang, 2010]. We make use of the labeled data available in the car domain (source), and predict the labels or the correct sentiment-mention association from the target domain (camera). Then we switch the source and target domain. We first use a simple Feature Level Fusion technique (described in Section 3.3.2.1) as our baseline for the transfer learning task. The Classifier Level Fusion technique (described in Section 3.3.2.2), which includes filtering of features, is then used as the main approach to improve the predictions and is evaluated by comparing with the predictions by the baseline approach.

3.3.2.1 Feature Level Fusion Approach

The task of transfer learning can be accomplished by a simple "feature level fusion approach" where we train a single classifier using training data from all the domains available, in our case car and camera domains. As it is depicted in Figure 3.2 the training data set consists of the labeled training data from both, car and camera domains. A classifier is then learned based on this training dataset and two experiments are conducted to predict lables for the two unlabeled test dataset from the camera and car dataset, respectively. As this classifier contains features from multiple domains it will not be domain specific and hence can enhance or reduce the accuracies for predictions depending on how closely related the two domains are. The predictions from this approach were considered as baseline and used to evaluate the classifier level fusion approach described next.

3.3.2.2 Classifier Level Fusion Approach

This is a more sophisticated approach than feature level classification. In the classifier level fusion approach, we train two individual classifiers for the source and target domains respectively, using the labeled data from each domain. The features for the source domain

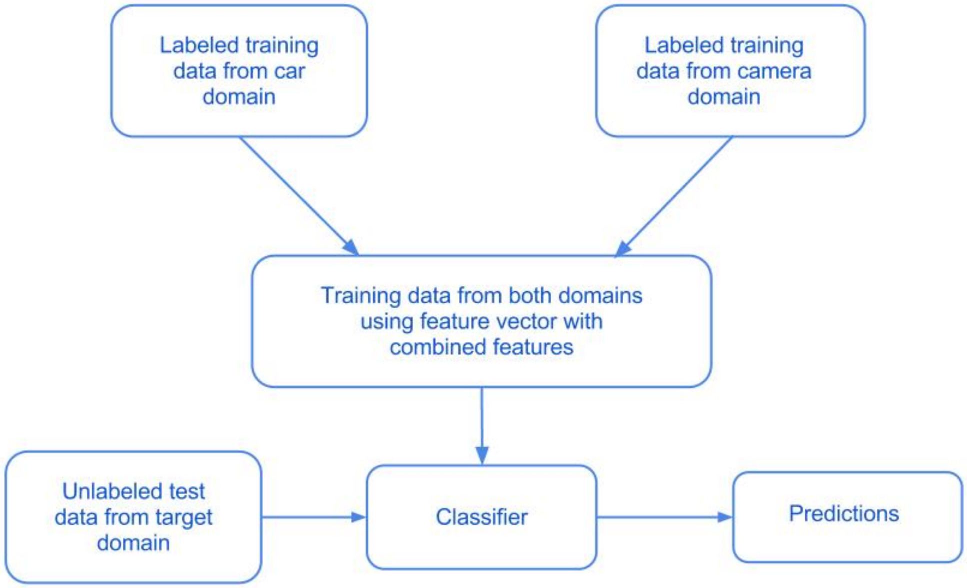

Figure 3.2: *The architecture of the feature-level fusion approach*

instances are filtered out and are limited to those observed in the target domain. In other words, the target domain instances are represented by all features in the target domain and the source domain instances are represented only with the common features between the target and source domains. Predictions are then made on a small test set of instances from the target domain by both classifiers that we learned. Once we have the probability scores from individual classifiers we use a weighted probability scoring scheme to assign final, improved scores to the test instances from the target domain (refer to Figure 3.3).This type of filtered feature set helps as the classification algorithm does not have to deal with those features that are not in the target domain, hence improving the overall prediction of the instances from the test set.

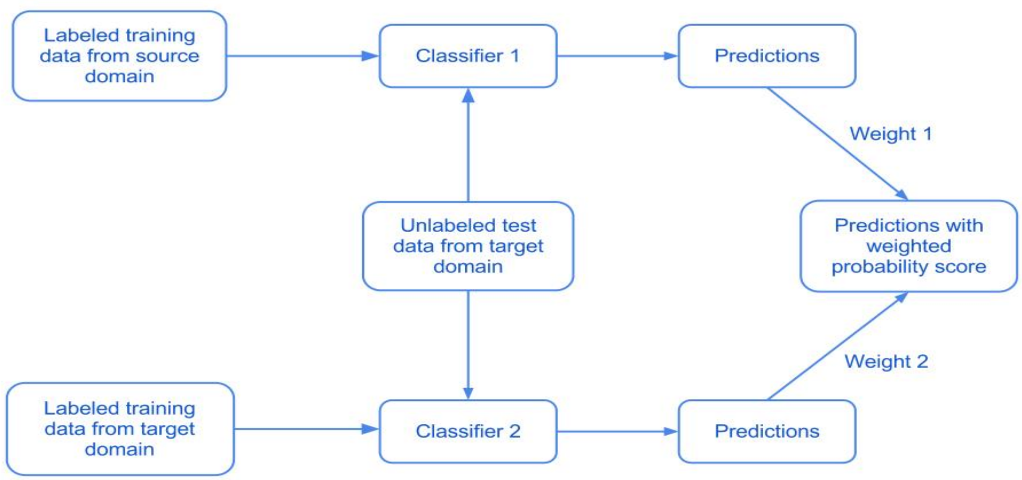

Figure 3.3: *The architecture of the classifier-level fusion approach*

Chapter 4

Experimental Setup

In this chapter, we describe the dataset used in this work and the experiments that were performed to predict the associations of the sentiment-mention pairs. We compare the performance of the transfer learning approach with the independent classifier approach. We have conducted various experiments with LibSVM [Chang and Lin, 2011] implementation of SVM classifiers to investigate its performance in predicting the associations between sentiments and mentions in the *JDPA Sentiment Corpus* [Kessler and Nicolov, 2010], which consists of documents having product reviews.

This Chapter is organized as follows: Section 4.1, provides the explanation of the SVM classifier implementation that we have used. The dataset used in this work is described in Section 4.2. As the dataset was from user product reviews, it included some unwanted symbols that required cleaning. Also we needed to tokenize the sentences and extract the mentions and sentiments from them. This was done in the preprocessing step which is also described in Section 4.2. Finally, Section 4.3 lists all the experiments that we have conducted.

4.1 Support Vector Machines (SVM) Implementation

- We used LibSVM implementation in this work. LibSVM is an integrated software for support vector classification (C-SVC, nu-SVC), regression (epsilon-SVR, nu-SVR) and distribution estimation (one-class SVM). It supports multi-class classification too [Chang and Lin, 2011].

4.2 Dataset Description and Preprocessing

The dataset that we used for this work is the *JDPA Sentiment Corpus* provided by authors Jason S. Kessler, Miriam Eckert, Lyndsie Clark, and Nicolas Nicolov [Kessler and Nicolov, 2010]. They presented this dataset at the 4th International AAAI Conference on Weblogs and Social Media Data Challenge Workshop (ICWSM-DCW 2010), 2010, Washington, D.C.

The dataset consists of user-generated content (blog posts) containing reviews about automobiles and digital cameras. The text files containing the reviews have been manually annotated for named and pronominal mentions of entities. Entities are marked with the sentiment polarity expressed toward them in the document. The dataset consists of 515 documents (blog posts) covering 330,762 tokens (words) which make up 19,322 sentences, out of which 87,532 mentions and 15,637 sentiment expressions are annotated.

The opinions are further divided into different batches. Each batch has user opinion in .txt files and corresponding annotations in .xml files. The .xml files have a unique id assigned to each sentiment or mention word. The .xml files also contains the start and end coordinates of a mention or a sentiment word, with its class and the linked sentiment polarity for some of the mentions. Figure 4.1 shows an extract of one of the .xml files from the dataset.

- As part of the preprocessing step, we cleaned the data set to remove any unwanted symbols (such as #, ! etc). Also, those blog posts that were not in the form of sentences were ignored.
- The next step in preprocessing involved tokenization. Each sentence from both corpora was tokenized and POS tagged.
- The mentions and sentiments were extracted from the text files in the corpus according to their span provided in the .xml files. The span of a mention or sentiment is the

```xml
<?xml version='1.0' encoding='us-ascii'?>
<annotations>
  <annotation>
    <mention id="car-004-1-structural-sentiment_Instance_0" />
    <annotator id="1">1</annotator>
    <span end="7" start="0" />
  </annotation>
  <classMention id="car-004-1-structural-sentiment_Instance_0">
    <mentionClass id="Mention.GeoPolitical.City">Mention.GeoPolitical.City</mentionClass>
    <hasSlotMention id="car-004-1-structural-sentiment_Instance_8" />
  </classMention>
  <annotation>
    <mention id="car-004-1-structural-sentiment_Instance_11" />
    <annotator id="1">1</annotator>
    <span end="12" start="9" />
  </annotation>
  <classMention id="car-004-1-structural-sentiment_Instance_11">
    <mentionClass id="Mention.GeoPolitical">Mention.GeoPolitical</mentionClass>
    <hasSlotMention id="car-004-1-structural-sentiment_Instance_19" />
  </classMention>
  <annotation>
    <mention id="car-004-1-structural-sentiment_Instance_10966" />
    <annotator id="1">1</annotator>
    <span end="29" start="15" />
  </annotation>
  <classMention id="car-004-1-structural-sentiment_Instance_10966">
    <mentionClass id="Mention.Time.Date">Mention.Time.Date</mentionClass>
  </classMention>
  <annotation>
    <mention id="car-004-1-structural-sentiment_Instance_10998" />
    <annotator id="1">1</annotator>
    <span end="55" start="48" />
  </annotation>
  <classMention id="car-004-1-structural-sentiment_Instance_10998">
    <mentionClass id="SentimentBearingExpression">SentimentBearingExpression</mentionClass>
    <hasSlotMention id="car-004-1-structural-sentiment_Instance_11000" />
  </classMention>
```

Figure 4.1: *XML File Sample from the JDPA Sentiment Corpus*

difference between start and end coordinates provided in the .xml files. The mention class was also extracted for each mention as the class of the mention is also one of the features in the feature set.

- As no labeled sentiment-mention pairs were provided with the dataset, we formed pairs of all possible sentiments-mentions of a sentence and manually labeled the pair as 1 or 0 corresponding to whether the pair is a valid sentiment pair or not.

For example: *I* liked the *Accord* with powerful *engine*, more than the *Camry*.

In this sentence, there are four mentions "I", "Accord", "Camry" and "engine" (italics) and two sentiment words: "liked" and "powerful" (underlined). The next task was to determine the mentions which the sentiments, "liked" and "powerful", target.

Manual Labeling:

Below we have all the possible sentiment-mention pairs from the sentence in the example and we also have the classes (0,1) to which the pair belongs.

```
liked:  I: 0
liked:  Accord:  1
liked:  Engine:  0
liked:  Camry:  0
powerful:  I: 0
powerful:  Accord:  0
powerful:  Engine:  1
powerful:  Camry:  0
```

Next, we require to extract the semantic and syntactic features for all the possible sentiment-mention pairs. For this, we used the *Stanford Dependency Parser* [Marneffe et al., 2006] for parsing the text files and then POS tagging was done for each sentence. The output of the Stanford Dependency Parser had POS tags attached to each word or token and it also gave the dependency relations between the tokens in the sentence.

POS Tagging:

I/*PRP* liked/*VBD* the/*DT* Accord/*NNP* with/*IN* powerful/*JJ* engine/*NN* ,/, more/*RBR* than/*I* the/DT Camry/*NNP* ./.

Typed Dependency output:

```
nsubj (liked-2, I-1)
det (Accord-4, the-3)
dobj (liked-2, Accord-4)
prep (liked-2, with-5)
```

```
amod (engine-7, powerful-6)
pobj (with-5, engine-7)
advmod (liked-2, more-9)
prep (more-9, than-10)
det (Camry-12, the-11)
pobj (than-10, Camry-12)
```

One of the semantic features included dependency paths for which each word in the sentence was assumed as a node of the graph and each dependency as an edge between the nodes. We wrote an implementation for *Dijkstra algorithm* that was used to calculate the shortest dependency path between the sentiment expression and mention pairs. The dependency graph for the typed dependency output shown above is depicted in Figure 3.1.

Once we have extracted all the features for each sentiment-mention pair, we preformed experiments that are described in Section 4.3.

4.3 Experiments

For this work we conducted experiments in two phases. In the first phase we learned individual classifiers based on blog posts from the car and camera domains separately. The second phase was the transfer learning phase, where we made use of the labeled data from the source domain to help predicting the correct associations in the target domain. Both car and camera domain were treated as source and target respectively, in different experiments. The experiments were conducted to answer the following research questions:

- How does the transfer learning (cross-domain) approach improve the predictions from the independent classifier (domain-specific) approach?
- Which approach gives better predictions: feature level fusion or classifier level fusion?
- In which direction does transfer learning works more effectively: car (source) to camera (target) or vice-versa?

4.3.1 Domain-Specific Classifiers

- In the first experiment, we used the LibSVM [Chang and Lin, 2011] to learn the predictive model from the car domain data. We manually labeled 2549 possible sentiment-mention pairs. Syntactic and semantic features were generated and used; the size of the feature set was 3460. The test set consists of 600 instances from the car domain. This experiment is henceforth referred to as Experiment 1.

- In the second experiment, we used the camera dataset to learn a predictive model using LibSVM [Chang and Lin, 2011]. We manually labeled 2134 possible sentiment-mention pairs from the camera dataset. Similar syntactic and semantic features were generated and used; the size of the feature set in this case was 2819. The test dataset consisted of 497 unlabeled sentiment-mention pairs from the camera domain. This experiment is henceforth referred to as Experiment 2.

4.3.2 Cross-Domain Classifiers

In the second phase, we performed the transfer learning where we used the source domain labeled training data to further enhance the predictions of the models for the target domain data. We used two techniques for this, feature level fusion approach and classifier level fusion approach.

Feature Level Fusion

In this experiment, we combined the features from both the car and camera data sets and learned a classifier using LibSVM [Chang and Lin, 2011]. A total of 4683 possible sentiment-mention pairs were used while training. The size of the feature set was 5544 (including 735 common features). In Experiment 3, the test dataset consisted of 497 unlabeled sentiment-mention pairs from the camera domain. Experiment 4 is similar to Experiment 3, the difference being the test dataset which contained 600 possible sentiment-mention pairs from the car domain.

Classifier Level Fusion

- Car as source & camera as target domain: In this experiment, we learned two individual classifiers, one from the labeled data from the camera domain with all available features and another one from the labeled data from car domain but the features are restricted to those present also in the camera dataset. The test set consists of 497 instances from the camera domain. We predict the probability or class distribution score for these test instances using both car and camera classifiers. Then, we assign a combined weighted probability score to each sentiment-mention pair. This new score is then used to decide the class of the pair. This experiment is henceforth referred to as Experiment 5.

- Camera as source & car as target domain: Similar to experiment 5, here also we learned two individual classifiers, one from the labeled data from the car domain and another one from the labeled data from camera domain. The features are restricted to those in the target (car) dataset. The test set consists of 600 instances from the car domain. We predict the probability or class distribution score for these test instances using both the classifiers. Then, we assign a combined weighted probability score to each sentiment-mention pair. This new score is then used to decide the class of the pair. This experiment is henceforth referred to as Experiment 6.

LibSVM [Chang and Lin, 2011] implementation from WEKA [Hall et al., 2009] was used as the classifier for all the experiments. We also tuned the parameters for best performance in each experiment. Parameter tuning was conducted in WEKA where the gamma parameter in the radial basis function of SVM was varied from 0.1 to 1.0 in steps of 0.1. We used 10 fold cross-validation to tune the parameters for all the experiments that we conducted. A separate validation dataset (subset of the respective training set) was used for each tuning exercise.

Chapter 5

Results

In this chapter, we discuss the results of the experiments discussed in Section 4.3. The chapter is organized as follows: In Section 5.1 we discuss the results that were obtained with the domain-specific classifiers for both domains considered. Section 5.2 describes the results of the experiments for feature level fusion approach and classifier level fusion approach.

5.1 Results of the Domain-Specific Classifiers

As described in Section 4.3.1, we conducted two experiments in the first phase of this work, where we learned independent classifiers for car and camera domain instances. In Experiment 1, we manually labeled 2549 possible sentiment-mention pairs from car domain and extracted 3460 features from these sentences. The test set consists of 600 unlabeled instances from the car domain. For Experiment 2, we labeled 2134 sentiment-mention pairs from the camera dataset and the number of features from these instances were 2819. The test dataset consisted of 497 unlabeled sentiment-mention pairs from the camera domain. We used LibSVM [Chang and Lin, 2011] implementation of SVM to learn the individual classifiers and used 10 fold cross-validation for tuning the parameters and evaluation. The results of both these experiments are shown in Table 5.1.

We can see that the precision and recall obtained from the car dataset with a larger number of features, is greater than that obtained from the camera dataset.

Table 5.1: *Results of domain-specific classifiers (Experiment 1 and Experiment 2)*

Experiment #	1	2
Domain	Car	Camera
Possible Sentiment-Mention Pairs	2549	2134
Number of Features	3460	2819
Precision	87.5%	84.0%
Recall	86.0%	85.5%
F1 Measure	86.74%	84.74%

5.2 Results of the Cross-Domain Classifiers

For experiment 3, feature level fusion, as explained in Section 4.3.2 we combined the labeled training data from both the domains, which together resulted in 4683 possible sentiment-mention pairs and 5544 features. We then learned a classifier using LibSVM [Chang and Lin, 2011] and first tested it on a test set from camera domain, having 497 possible sentiment-mention pairs. The results of the experiment are shown in Table 5.2.

Table 5.2: *Results for the feature level fusion approach (Experiment 3 - camera domain used as target)*

Experiment #	3
Approach	Feature Level Fusion
Possible Sentiment-Mention Pairs	4683
Number of Features	5544
Precision	85.4%
Recall	87.3%
F1 Measure	86.33%

As can be seen from the results in Table 5.2 the accuracy of feature level fusion approach (camera domain used as target) was better than that of the domain-specific classifier shown in Table 5.1 (Experiment 2). This increase in accuracy can be explained by the addition of car domain instances.

Experiment 4 was similar to Experiment 3, the difference being that the test dataset was the car domain with 600 possible sentiment-mention pairs and 5544 features. The results of the experiment are shown in Table 5.3.

Table 5.3: *Results for the feature level fusion approach (Experiment 4 - car domain used as target)*

Experiment #	4
Approach	Feature Level Fusion
Possible Sentiment-Mention pairs	4683
Number of Features	5544
Precision	85.9%
Recall	86.4%
F1 Measure	86.15%

The results in Table 5.3 were worse when compared to those of the domain-specific classifier for car domain - Experiment 1 in Table 5.1. These results indicate that the camera domain labeled instances did not help in the prediction of the car instances.

For Experiment 5, classifier level fusion with car as source domain and camera as the target domain, we first learned individual classifiers for each domain using the labeled training data from the corresponding domain. For learning the camera classification model, the labeled training instances were represented by all 2819 features from the camera domain. However, for learning the car domain classification model the training instances were represented by the 735 features common to both domains. LibSVM [Chang and Lin, 2011] was used to learn the classifiers and the test set of 497 unlabeled sentiment-mention pairs from the camera dataset was used for evaluation. The results of each classifier are displayed in Table 5.4.

Table 5.4: *Results for the individual classifiers in classifier level fusion approach (Experiment 5), where car is used as source and camera is used as target*

Experiment #	Part of 5	Part of 5
Domain	Car (Source)	Camera (Target)
Possible Sentiment-Mention pairs	2549	2134
Number of Features	735	2819
Precision	87.1%	83.8%
Recall	86.7%	85.1%
F1 Measure	86.90%	84.44%

After learning the individual classifiers for both domains and testing them on the test set, we computed the weighted probability score for each of the 497 test unlabeled instances by varying the weights assigned to each of the predicted score from the car and camera domain and adding them. The results are shown in Table 5.5 by comparison with the results from feature level classification approach and domain-specific classification approach.

Table 5.5: *Comparison between domain-specific (Experiment 2), feature level fusion (Experiment 3) and classifier level fusion approaches (Experiment 5), where car is used as the source and camera is used as target*

Experiment #	2	3	5
Approach	Domain-Specific	Feature Level Fusion	Classifier Level Fusion
Sentiment-Mention pairs	2134	4683	2549(car) + 2134(camera)
Number of Features	2819	5544	735(car) + 2819(camera)
Precision	84.0%	85.4%	87.1%
Recall	85.5%	87.3%	92.6%
F1 Measure	84.74%	86.33%	89.77%

As can be seen from Table 5.5, the precision and recall obtained from classifier level fusion approach are higher by 1.7% and 5.3%, respectively, when compared to the corresponding values of the feature level fusion approach and domain-specific classifier approach results. The improvement confirms that learning individual classifiers with filtered features is helpful, as they eliminated the noise in the car (source) dataset. Also, it shows that the addition of knowledge from car (source) helped the predictions for camera (target).

We also switched the roles of the car and camera domains using car as target and camera as source in Experiment 6. We first learned individual classifiers for each domain using the labeled training data from the corresponding domain. For learning the camera classification model, the labeled training instances were represented by all 3460 features from the car domain. However, for learning the camera domain classification model the training instances were represented by the 735 features common to both domains. LibSVM [Chang and Lin, 2011] was used again to learn the classifiers and the test set of 600 unlabeled sentiment-mention pairs from the car dataset was used for evaluation. The results of each classifier are displayed in Table 5.6.

Table 5.6: *Results for the individual classifiers in classifier level fusion approach (Experiment 6), where camera is used as source and car is used as target domain*

Experiment #	Part of 6	Part of 6
Domain	Camera (Source)	Car (Target)
Possible Sentiment-Mention pairs	2134	2549
Number of Features	735	3460
Precision	83.9%	86.8%
Recall	84.9%	87.2%
F1 Measure	84.39%	87.0%

As before, after learning the individual classifiers for both domains and testing on the test set, we computed the weighted probability score for each of the 600 test unlabeled instances by varying the weights assigned to each of the predicted scores from the car and camera domains and adding them. The results are shown in Table 5.7 by comparison with the results from domain-specific classification and feature level classification approaches.

Table 5.7: *Comparison between domain-specific (Experiment 1), feature level fusion (Experiment 4) and classifier level fusion approaches (Experiment 6), where camera is used as source and car is used as target*

Experiment #	1	4	6
Approach	Domain-Specific	Feature Level Fusion	Classifier Level Fusion
Sentiment-Mention pairs	2549	4683	2134(camera) + 2549(car)
Number of Features	3460	5544	3460(camera) + 735(car)
Precision	87.5%	85.9%	87.2%
Recall	86.0%	86.4%	91.9%
F1 Measure	86.74%	86.15%	89.50%

As can be seen from Table 5.7 the precision and recall obtained from classifier level fusion approach are higher by 1.3% and 5.5% respectively, when compared to that of feature level fusion approach results. This shows that addition of the knowledge from the camera (source) domain helped the predictions for the car (target) domain.

Therefore, the classifier level fusion approach improved results in both cases, showing that knowledge from a different domain can be very useful. However, classifiers based on feature level fusion can improve the results of domain-specific classifiers but not in all cases.

Chapter 6

Conclusions and Future Work

In this chapter, we discuss the conclusions (Section 6.1) of our work based on the results from Chapter 5. In Section 6.2, we mention some of the enhancements that could be included in this work to further improve the predictions.

6.1 Conclusions

In this section we answer the research questions that were raised earlier in Section 4.3.

- How does the transfer learning approach help the predictions from the independent (domain-specific) classifier approach?

  The transfer learning approach definitely helped in improving the predictions when compared to the domain-specific classifier approach. Both feature level fusion (in some cases) and classifier level fusion produced better results than those of the domain-specific classification approach. The evidence of the claim is visible when we compare the results from Table 5.5, where car was used as source and camera as target. We can clearly see that the F1 score (measure of accuracy of the test) of the classifier level fusion is higher by 5.03 % when compared to that of the independent (domain-specific) classifier learned on camera dataset.

  The performance of feature level classification was also improved by 1.59 % as compared to the camera domain independent classifier.

The improvement in predictions from classifier domain approach is also visible from Table 5.7, where camera was used as source and car was used as target. The F1 score (accuracy) of the predictions is higher by 2.76 % when compared to that of domain-specific classifier learned on car dataset. However, the performance of feature level classifier was worse than that of the domain-specific classifier with car domain as target. The accuracy was decreased marginally by 0.59 %, showing that feature level fusion does not work in all circumstances.

- Which approach gives better predictions: feature level fusion or classifier level fusion?

  Results in Table 5.5 and Table 5.7 show that classifier level fusion performed better as compared to feature level fusion approach. The classifier level fusion approach provided improved accuracy as only the selected (common) features from the source domain were used in transferring the knowledge to the target domain, which helped remove the noise from the source domain data. On the other hand, feature level fusion used all features from the source domain and the target domain which resulted in a decrease of the performance (F1 measure) as noise hampered the predictions.

- In which direction does transfer learning works more effectively: car (source) to camera (target) or vice-versa?

  Experiment 5 was conducted to test if car (source) domain can help improve the predictions for the camera (target) domain. The addition of knowledge from car (source) to the camera (target) improved the predictions from the camera test dataset. The results from Tables 5.4 and 5.5 showed a 5.33 % increase in accuracy (F1 measure) for the classifier level fusion approach.

  Experiment 6 was the inverse of Experiment 5. Here, we used the knowledge from the camera (source) domain to improve predictions for the car (target) domain. Tables 5.6 and 5.7 indicate a 2.5 % improvement in the accuracy for the car domain test dataset.

  In spite of improvement in both directions of transfer learning (car to camera or vice-versa), the change in accuracy of the test is clearly more visible in the case of Experiment 5. It leads to the conclusion that car domain (source) more significantly helped the camera domain (target) predictions.

While conducting the experiments, we also observed the relationship between the number of possible sentiment-mention pairs and the number of features. There was almost a linear relation between them: as we increased the number of sentiment-mention pairs, the number of features kept on rising.

We also conducted some experiments with balanced datasets that had equal distribution of the correct and incorrect sentiment-mention associations, but that did not affect the results to a great extent, hence the results are not shown here.

From the answers to our research questions, we concluded that transfer learning approach certainly helped in the predictions from the domain-specific classifiers. The classifier level fusion approach performed better than the feature level and also we concluded that the car domain used as source helped the camera domain (target) predictions. These improvement in predictions confirms the correct sentiment-mention associations for the product reviews. This would in turn help the sentiment analysis task of extracting the crux of these reviews helping the potential customers and manufacturers.

6.2 Future Work

This section mentions some of the future extensions to this work which could be interesting candidates to further improve the predictions:

- We extracted all possible sentiment-mention pairs from one sentence. It would be interesting to see sentiment-mention associations between two or more sentences. In other words performing the *entity resolution* for the dataset that is determining which person, object or physical entity is referred by the references. Below is an example that explains it clearly:

"*Land-Cruiser* is a powerful and aggressive off-roader. *It* is also the most luxurious in the market."

In the example above, "powerful" and "aggressive" sentiments are associated with "Land-Cruiser". And "luxurious" sentiment is associated with "It". However, "It" (a preposition) also refers to "Land-Cruiser", this reference is known entity resolution.

- Given the large number of features in our experiments, the technique of feature selection could also be applied to the this work, where we could limit the number of features that could in turn improve the predictions. Feature selection techniques searches for a subset from the set all features as a group for suitability. It uses a search algorithm to search through the space of possible features and evaluate each subset by running a model on the subset. Various search algorithms that could be used are: greedy hill climbing, greedy forward selection, greedy backward elimination, best first search and exhaustive search.

- It would be worth trying other transfer learning approaches and other domains too. The task we considered in this work was a *transductive transfer learning*, where the source and target tasks are same, however, the source and target domains differ. Various transductive transfer learning techniques mentioned in paper [Arnold et al., 2007] includes: Iterative Feature Transformation (IFT), Transductive SVMs and source-initialized EM. More related domains (like: trucks, bikes etc.) could also be used to improve the predictions (provided dataset is available).

Bibliography

Andrew Arnold, Ramesh Nallapati, and William W. Cohen. A comparative study of methods for transductive transfer learning. In *In ICDM Workshop on Mining and Management of Biological Data*, 2007.

Aue, Anthony, Gamon, and Michael. Customizing sentiment classifiers to new domains: A case study. In *Proceedings of Recent Advances in Natural Language Processing (RANLP)*, 2005.

Kenneth Bloom, Navendu Garg, and Shlomo Argamon. Extracting appraisal expressions. In *Proceedings of Human Language Technologies/North American Association of Computational Linguists*, 2007.

Chih-Chung Chang and Chih-Jen Lin. LIBSVM: A library for support vector machines. *ACM Transactions on Intelligent Systems and Technology*, 2:27:1–27:27, 2011. Software available at http://www.csie.ntu.edu.tw/~cjlin/libsvm.

Corinna Cortes and Vladimir Vapnik. Support-vector networks. *Machine Learning*, 20:273–297, 1995. ISSN 0885-6125. URL http://dx.doi.org/10.1007/BF00994018. 10.1007/BF00994018.

Kavita Ganesan and Hyun Duk Kim. Opinion mining - a short tutorial (talk), 2008.

Mark Hall, Eibe Frank, Geoffrey Holmes, Bernhard Pfahringer, Peter Reutemann, and Ian H. Witten. The weka data mining software: An update. In *SIGKDD Explorations, Volume 11, Issue 1*, 2009. URL http://www.kdd.org/explorations/issues/11-1-2009-07/p2V11n1.pdf.

Ryu Iida, Kentaro Inui, Hiroya Takamura, and Yuji Matsumoto. Incorporating contextual cues in trainable models for coreference resolution. In *In Proceedings of the EACL Workshop on The Computational Treatment of Anaphora*, pages 23–30, 2003.

Peng Jiang, Chunxia Zhang, Hongping Fu, Zhendong Niu, and Qing Yang. An approach based on tree kernels for opinion mining of online product reviews. In *Proceedings of the 2010 IEEE International Conference on Data Mining*, ICDM '10, pages 256–265, Washington, DC, USA, 2010. IEEE Computer Society. ISBN 978-0-7695-4256-0. doi: http://dx.doi.org/10.1109/ICDM.2010.104. URL http://dx.doi.org/10.1109/ICDM.2010.104.

T. Joachims. Training linear svms in linear time. In *ACM Conference on Knowledge Discovery and Data Mining (KDD), 2006*, 2006. URL http://www.cs.cornell.edu/People/tj/publications/joachims_06a.pdf.

Jason S. Kessler and Nicolas Nicolov. The 2010 icwsm jdpa sentment corpus for the automotive domain. In *4th International AAAI Conference on Weblogs and Social Media Data Workshop Challenge (ICWSM-DWC 2010)*, 2010. URL http://www.cs.indiana.edu/~{}jaskessl/icwsm10.pdf.

Jason S. Kessler, Miriam Eckert, Lyndsie Clark, and Nicolas Nicolov. Targeting sentiment expressions through supervised ranking of linguistic configurations. In *3rd International AAAI Conference on Weblogs and Social Media (ICWSM- 2009)*, 2009. URL https://www.cs.indiana.edu/~{}jaskessl/icwsm09.pdf.

Nozomi Kobayashi, Ryu Iida, Kentaro Inui, and Yuji Matsumoto. Opinion mining on the web by extracting subject-attribute-value relations. In *Proceedings of AAAI 2006 Spring Sympoia on Computational Approaches to Analyzing Weblogs (AAAI-CAAW 2006)*, 2006.

Shoushan Li and Chengqing Zong. Multi-domain sentiment classification. In *Proceed-*

ings of ACL-08: HLT, Short Paper, 2008. URL http://aclweb.org/anthology/P/P08/P08-2065.pdf.

Manning and H. Schutze. *Foundations of Statistical Natural Language Processing*. The MIT Press, 1999.

Marneffe, MacCartney, and Christopher D. Manning. Stanford dependency parser - generating typed dependency parses from phrase structure parses. In *Generating Typed Dependency Parses from Phrase Structure Parses - LREC 2006*, 2006. URL http://nlp.stanford.edu/pubs/LREC06_dependencies.pdf.

M. T. Mitchell. *Machine learning*. McGraw-Hill Companies Inc., international edition, 1997.

Nicolas Nicolov, Franco Salvetti, and Steliana Ivanova. Sentiment analysis: Does coreference matter. In *In AISB 2008 Convention Communication, Interaction and Social Intelligence*, 2008.

Sinno Jialin Pan and Qiang Yang. A survey on transfer learning. *Knowledge and Data Engineering, IEEE Transactions on*, 22(10):1345 –1359, oct. 2010. ISSN 1041-4347. doi: 10.1109/TKDE.2009.191.

Sinno Jialin Pan, Xiaochuan Ni, Jian tao Sun, Qiang Yang, and Zheng Chen. Cross-domain sentiment classification via spectral feature alignment. In *In WWW*, 2010.