A Multiple Classifier System for Predicting Best-Selling Amazon Products

Michael Kranzlein

Follow this and additional works at: https://digitalcommons.kennesaw.edu/cs_etd

Part of the Other Computer Sciences Commons

Recommended Citation

https://digitalcommons.kennesaw.edu/cs_etd/18

This Thesis is brought to you for free and open access by the Department of Computer Science at DigitalCommons@Kennesaw State University. It has been accepted for inclusion in Master of Science in Computer Science Theses by an authorized administrator of DigitalCommons@Kennesaw State University. For more information, please contact digitalcommons@kennesaw.edu.
A MULTIPLE CLASSIFIER SYSTEM FOR PREDICTING BEST-SELLING AMAZON PRODUCTS

A Thesis Presented to
The Faculty of the Department of Computer Science

By

Michael M. Kranzlein

In Partial Fulfillment
of Requirements for the Degree
Master of Science in Computer Science
in the
College of Computing and Software Engineering

Kennesaw State University
May 2018

Copyright © Michael M. Kranzlein 2018
In presenting this thesis as a partial fulfillment of the requirements for an advanced degree from Kennesaw State University, I agree that the university library shall make it available for inspection and circulation in accordance with its regulations governing materials of this type. I agree that permission to copy from, or to publish, this thesis may be granted by the professor under whose direction it was written, or, in his absence, by the dean of the appropriate school when such copying or publication is solely for scholarly purposes and does not involve potential financial gain. It is understood that any copying from or publication of, this thesis which involves potential financial gain will not be allowed without written permission.

Michael Kranzlein
Notice To
Borrowers

Unpublished theses deposited in the Library of Kennesaw State University must be used only in accordance with the stipulations prescribed by the author in the preceding statement.

The author of this thesis is:

Michael M. Kranzlein
307 Marden Court
Smyrna, GA 30082

The director of this thesis is:

Dan Chia-Tien Lo
1100 South Marietta Parkway
Marietta, Georgia 30060

Users of this thesis not regularly enrolled as students at Kennesaw State University are required to attest acceptance of the preceding stipulations by signing below. Libraries borrowing this thesis for the use of their patrons are required to see that each user records here the information requested.

Name of user    Address    Date    Type of use (examination only or copying)
ACKNOWLEDGEMENTS

I would like to express my gratitude to my advisor, Dr. Dan Chia-Tien Lo, for his continued support, guidance, and encouragement throughout the process of writing this thesis. I would also like to thank Dr. Mingon Kang and Dr. Kai Qian for being members of my thesis committee. Finally, I owe a great deal of thanks to the faculty of the Department of Computer Science at Kennesaw State University under whom I have studied and grown as a student and researcher.
# TABLE OF CONTENTS

Acknowledgments ................................................................. iv

List of Tables ................................................................. viii

List of Figures ................................................................. ix

Chapter 1: Introduction and Background ................................. 1
  1.1 Overview ............................................................... 1
  1.2 Structure of Work .................................................... 2

Chapter 2: Related Works .................................................... 3
  2.1 Binary Classification .................................................. 3
    2.1.1 Text Classification .............................................. 3
    2.1.2 Image Classification .......................................... 3
  2.2 Ecommerce ............................................................ 3
    2.2.1 Economic Impact of Product Ratings and Reviews .......... 4
    2.2.2 Economic Impact of Product Descriptions .................. 5
    2.2.3 How Image Quality and Other Media Affect Sales .......... 5

Chapter 3: Dataset and Preprocessing .................................... 6
  3.1 Updated SNAP Dataset Overview .................................. 6
Chapter 4: Methods

4.1 MCS Overview

4.1.1 Text CNNs and Other Models

4.2 Model Architectures

4.2.1 Text Models

4.2.2 Image CNN

4.2.3 Image CNN

4.3 Model Combination

Chapter 5: Results

5.1 Single-Model Experiment Results

5.1.1 Books

5.1.2 Home and Kitchen

5.1.3 Sports and Outdoors

5.1.4 Examining the models

5.1.5 Evaluating Text Model Consistency

5.2 Model Combination Results

5.3 Analysis and Comparison

Chapter 6: Conclusion

6.1 Interpretation of Results
6.2 Future Work ......................................................... 30

Appendix A: Libraries, Frameworks, and Guides ......................... 33

References ............................................................. 40
LIST OF TABLES

3.1 Fields and Values of a Sample Metadata Record (Excluding Description) from the Updated SNAP Dataset .......................... 6

3.2 Dataset Categories by the Numbers .................................. 9

5.1 Single-Model Results for Books Category: BSP vs. NBSP ........ 17

5.2 Single-Model Results for Books Category: BSP vs. WSP .......... 17

5.3 Single-Model Results for Home and Kitchen Category: BSP vs. NBSP ... 17

5.4 Single-Model Results for Home and Kitchen Category: BSP vs. WSP ... 18

5.5 Single-Model Results for Sports and Outdoors Category: BSP vs. NBSP . 18

5.6 Single-Model Results for Sports and Outdoors Category: BSP vs. WSP . 18

5.7 10-Fold Cross Validation of Text Models ............................ 24

5.8 MCS Results - BSP vs. NBSP ............................................ 25

5.9 MCS Results - BSP vs. WSP .............................................. 25
# LIST OF FIGURES

3.1 Sample product images from the Books category ........................................... 9  
3.2 Sample product images from the Home and Kitchen category ......................... 10  
3.3 Sample product images from the Sports and Outdoors category ....................... 10  
4.1 Architecture of LeNet-5, One of the First Widely Successful CNNs .................. 14  
5.1 Best and Worst Tokens for Books - BSP vs. NBSP (Count-Based Vectorization) .... 20  
5.2 Best and Worst Tokens for Books - BSP vs. WSP (Count-Based Vectorization) .... 20  
5.3 Best and Worst Tokens for Books - BSP vs. NBSP ........................................ 21  
5.4 Best and Worst Tokens for Books - BSP vs. WSP ........................................ 21  
5.5 Best and Worst Tokens for Home and Kitchen - BSP vs. NBSP ....................... 21  
5.6 Best and Worst Tokens for Home and Kitchen - BSP vs. WSP ....................... 22  
5.7 Best and Worst Tokens for Sports and Outdoors - BSP vs. NBSP ................... 22  
5.8 Best and Worst Tokens for Sports and Outdoors - BSP vs. WSP ................... 22  
5.9 Filters for First Convolutional Layer ......................................................... 23  
5.10 Test Images After Applying Filters from the First Convolutional Layer ............ 23
ABSTRACT

In this work, I examine a dataset of Amazon product metadata and propose a heterogeneous multiple classifier system for the task of identifying best-selling products in multiple categories. This system of classifiers consumes the product description and the featured product image as input and feeds them through binary classifiers of the following types: Convolutional Neural Network, Naïve Bayes, Random Forest, Ridge Regression, and Support Vector Machine. While each individual model is largely successful in identifying best-selling products from non best-selling products and from worst-selling products, the multiple classifier system is shown to be stronger than any individual model in the majority of cases of identifying best-selling products from non best-selling products, and achieves up to 83.3% accuracy, depending on the product category. To my best knowledge, this research is the first application of ensemble learning to Amazon product data of this type and the first use of product images and Convolutional Neural Networks to predict product success.
CHAPTER 1
INTRODUCTION AND BACKGROUND

1.1 Overview

Across global markets, ecommerce is rapidly expanding. At the end of 2015, The Pew Research Center reported that 79% of U.S. adults now shop online at least occasionally, and 28% shop online frequently [1]. However, the ecommerce marketplace is not just being flooded with consumers. It is also seeing a large influx of new vendors, ranging from single individuals to multinational corporations. In 2015, Amazon reported 50% growth in the number of sellers on its fulfillment platform [2].

To provide these vendors and consumers with additional decision-making tools, and to advance current understanding of how the attributes of product listings affect sales, I present a new ensemble learning approach built on product descriptions and images to classify a product as a best-selling product (“BSP”), a non best-selling product (“NBSP”), or a worst-selling product (“WSP”). This approach involves five models whose results are combined to create a multiple classifier system (“MCS”). Four of these models operate on product descriptions, and the remaining model operates on product images as input. The product description models include a Naïve Bayes Classifier, a Random Forest Classifier, a Ridge Regression Classifier, and a Support Vector Machine (“SVM”). Finally, a Convolutional Neural Network (“CNN”) is used for product image classification. Using a simple majority voting scheme, the outputs of these models are combined to boost performance even further. Ultimately, the MCS identifies a BSP correctly up to 83.3% of the time. While the image CNN performs better than random guessing, it is the weakest of the five models, and I discuss whether it’s worth including in later sections.

This new contribution has multiple practical implications. While some tools are cur-
rently available to vendors, they lack the capacity to identify deeper, non-intuitive patterns in product listings. For example, Jungle Scout is a rapidly growing company that offers a number of products designed to help vendors on the Amazon platform. One of these products is a grading lister that can help sellers gauge and improve the quality of their product listings, but it only accounts for descriptive, easily measured factors such as the length of the product description or the number of images included in the listing [3]. It does not evaluate product listings on less obvious indicators of quality, such as which keywords perform the best in the product category. Using the proposed MCS, sellers on ecommerce platforms can tailor their product descriptions and images even further to patterns that aren’t readily apparent in order to drive increased sales.

This approach can help consumers, too. It is well-known and documented that consumers’ decision making is strongly informed by product reviews [4, 5, 6]. However, in cases where, for example, a recently launched product has not been on the market long enough to fetch meaningful feedback, buyers can use the proposed MCS to help inform their choices. Since sales rank might be interpreted as a proxy for quality, a consumer may feel more comfortable buying a product if the model shows it is likely to become a best-seller. Conversely, a consumer may be able to avoid ending up the guinea pig who leaves a one-star review and warns other users to stay away from a lower-quality product. This helps make online shopping a less painful experience for consumers and helps newer vendors selling quality products gain market share more quickly.

1.2 Structure of Work

Chapter 2 presents a survey of related works. Chapter 3 provides details on the datasets and preprocessing steps for the experiments. Chapter 4 discusses the component models of the MCS. Chapter 5 highlights the results of the individual classifiers and of the MCS, and Chapter 6 offers an interpretation of the results and a discussion of future work.
CHAPTER 2
RELATED WORKS

2.1 Binary Classification

2.1.1 Text Classification

Text classification is a problem that has been studied for decades, and applications of text classification grew rapidly alongside the rise of the internet and the field of information retrieval [7, 8]. A text classifier can be employed for tasks such as identifying spam [9], determining document relevance [10], or detecting sentiment [11].

2.1.2 Image Classification

Image classification has also been studied extensively, and as the cost of computation has decreased, image classification has found applications in optical character recognition [12], facial recognition [13], autonomous driving [14], remote sensing and hyperspectral imaging [15, 16], medical imaging [17], and more. Binary classification represents a subfield of this domain, in which outputs are constrained to a positive and negative class.

2.2 Ecommerce

Historically, machine learning research in the domain of ecommerce has focused on text models built on user-provided product feedback (i.e. ratings and reviews), with many related goals such as identifying fraudulent and spam reviews, measuring the effects reviews have on consumer behavior, detecting review sentiment, and more. Only recently has research started to tap into the trove of information that exists in product descriptions and other listing information. The following works show interesting results in the domain of product feedback and listing analysis and helped to inform my own approach to this work.
2.2.1 Economic Impact of Product Ratings and Reviews

Hu et al. examine the effects of ratings (the 1 through 5-star metric associated with reviews) and review sentiment on product sales [18]. They found that ratings indirectly impact sales through the sentiments of associated reviews. They also found that review helpfulness and recency play an important role in improving sales. The significance of their work and its relation to this work lies in the conclusion that “information that is easily accessible and cognitive effort-reducing heuristics play a role in online purchase decisions” [18]. Product descriptions and images may be considered additional examples of the “effort-reducing heuristics” consumers rely on. Ling et al. engaged in similar research on recommender systems to “apply topic modeling techniques on the review text and align the topics with rating dimensions to improve prediction accuracy” [19]. Chong et al. also found that the interplay between ratings, reviews, and other factors is important for predicting success [20].

Other researchers acquired access to Alibaba data and studied another facet of sales prediction—vendor reputation—finding that sellers who reward feedback receive more reviews and achieve higher sales, though this practice raises important ethical questions that need to be considered [21]. Bao and Chung examine the gap between traditional and social media feedback on Amazon and find that “multiple earned media produce combined sales effects greater than those resulting from the sum of their parts” [22]. Finally, Wulff et al. studied how people make decisions based on description or on experience. They noted consumers’ “reliance on relatively small samples of information and overweighting of recently sampled information” in an ecommerce setting [23]. The growing body of work around the economic impact of product ratings and reviews underscores the importance of research in this domain. Expansion into analysis of the rest of the product listing (i.e. descriptions, images, and even other components) seems to be a natural progression.
2.2.2 Economic Impact of Product Descriptions

In 2017, Stanford researchers quantified for the first time the effects of product descriptions on sales in an ecommerce marketplace [24]. Specifically, Pryzant et al. mined “90,000+ product descriptions on the Japanese e-commerce marketplace Rakuten...” and identified “...actionable writing styles and word usages that are highly predictive of consumer purchasing behavior” [24]. The authors found that keywords in the categories of “Informativeness”, “Authority”, “Seasonality”, and “Politeness” were most influential on sales. This research is of particular note due to its consideration and elimination of confounding factors on product sales such as brand reputation.

2.2.3 How Image Quality and Other Media Affect Sales

Zakrewsky et al. highlight the importance of image quality characteristics, including “color, simplicity, scene composition, texture, style, aesthetics and overall quality” [25] and show an improvement in model performance using feature vectors designed to capture these characteristics. Xu et al. go beyond just product images and consider a newer medium in the ecommerce space: video. These researchers study video reviews and how they compare to more traditional review media such as images and text. They conclude that “the presentation format of online reviews has a substantive and nuanced impact on consumer perceptions” and that “product type significantly moderates the effect of presentation format on consumer perceptions” [26].
CHAPTER 3
DATASET AND PREPROCESSING

3.1 Updated SNAP Dataset Overview

Unfortunately, there aren’t many widely available public ecommerce datasets. Actual sales numbers are often closely guarded secrets to help ecommerce platforms maintain their competitive advantage. One of the few public datasets that does exist is the SNAP Amazon Review Dataset. This is a large dataset that was originally part of SNAP, the Stanford Network Analysis Project [27]. I use an updated version of this dataset that contains millions of reviews and associated product metadata including Amazon Standard Identification Numbers (ASIN) [28], descriptions, image URLs, and more for Amazon products in a wide range of categories [29]. Although the dataset doesn’t include explicit sales numbers, it does include the publicly available Amazon Best Sellers Rank [30], which is used in this work as a metric to identify best-selling products. Table 3.1 showcases the types of product metadata available in the dataset.

Table 3.1: Fields and Values of a Sample Metadata Record (Excluding Description) from the Updated SNAP Dataset

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>ASIN</td>
<td>0000031852</td>
</tr>
<tr>
<td>Title</td>
<td>Girls Ballet Tutu Zebra Hot Pink</td>
</tr>
<tr>
<td>Price</td>
<td>3.17</td>
</tr>
<tr>
<td>Also Bought</td>
<td>B00JHONN1S, B002BZX8Z6, B00D2K1M3O, ...</td>
</tr>
<tr>
<td>Also Viewed</td>
<td>B002BZX8Z6, B00JHONN1S, B008F0SU0Y, ...</td>
</tr>
<tr>
<td>Bought together</td>
<td>B002BZX8Z6</td>
</tr>
<tr>
<td>Sales rank</td>
<td>Toys &amp; Games: 211836</td>
</tr>
<tr>
<td>Brand</td>
<td>Coxlures</td>
</tr>
<tr>
<td>Categories</td>
<td>Sports &amp; Outdoors, Other Sports, Dance</td>
</tr>
</tbody>
</table>
3.1.1 Disadvantages of the Dataset

There are disadvantages in the dataset that are important to acknowledge. Two of the most pernicious factors are the overall age of the dataset and the fact that sales ranks are non-temporal.

Let’s consider the age of the dataset first. The dataset includes “142.8 million reviews spanning May 1996 - July 2014” [29]. As such, the model may not capture new, emerging patterns that could complement the longstanding, established patterns that can be detected. In addition, it is known that ecommerce revenues, Amazon’s in particular, are rapidly increasing [31]. This means the years of data the dataset is missing (the most recent ones) would also likely be the most informative to the model, as total sales and the number of products available on ecommerce platforms continue to grow.

A second problem is that the metadata are undated. This applies to sales ranks, too, which means we can’t account for or derive insight from the evolution of a product’s performance over long periods of time. Furthermore, there may be multiple products that share the same sales rank. Perhaps product x was ranked #1 in a category on one day that data were captured, but product y was ranked #1 in a category the next day that data were captured.

A tangential issue is introduced in that products are only unique according to their ASIN. In some cases, a seller may have two very similar products under two different listings, each with its own ASIN. This is less an issue with this dataset in particular and more of a problem with scraping data from Amazon in general. In future work, the use of automated tools to identify and purge these near duplicates would improve the cleanliness of the data and therefore the reliability of the models as well.
3.2 Sampling the Dataset

As discussed previously, five models are incorporated into the MCS. The models were trained independently on data sampled from three Amazon product categories. Products can belong to multiple categories, so these categories do not necessarily contain disjoint products, though these instances are exceptions to the rule. The categories are as follows:

- Books
- Home and Kitchen
- Sports and Outdoors

In each case, I selected the top 20,000 products (by sales rank) as best-selling products (BSPs) and sampled 20,000 non best-selling products (NBSPs) from a random uniform distribution of the rest of the products in the category. Finally, I sampled the bottom 20,000 products in the category as worst-selling products (WSPs). The selected categories were chosen out of 24 available categories in the dataset. While each of the 24 categories has a large number of products, \( n > 30648 \), not all products are ranked. Furthermore, some rankings exist in the dataset while others may not. As an example, there may be a product ranked 17, but no product ranked 18 in a given category. In order to generate high-quality BSP and WSP datasets, only categories with a sufficient number of ranked products and a sufficient distribution of rankings could be used. After manual exploration of the various categories, I found the selected categories listed above to be the most viable options.

With that said, there are a few observations to note:

1. The highest sales rank in a BSP dataset may be greater than 20,000, since some sales ranks are missing.

2. The highest sales rank in a BSP dataset may be less than 20,000, since some sales ranks are claimed multiple times by different products.
3. The highest sales rank in a NBSP or WSP dataset may be greater than the number of products in the dataset, since not all products are included in the dataset.

4. The NBSP sample groups may contain WSPs, but they will never contain BSPs.

5. The fewer products in a category, the more WSPs and NBSPs will overlap in NBSP sample groups.

6. There is a one-to-one relationship between product descriptions and product images in all of the sample groups: BSP, NBSP, WSP. That is, each sample, identified by its ASIN, will have a non-empty product description and one product image.

Table 3.2: Dataset Categories by the Numbers

<table>
<thead>
<tr>
<th>Category</th>
<th>Products</th>
<th>Eligible Products</th>
<th>Best Sales Rank</th>
<th>Worst Sales Rank</th>
</tr>
</thead>
<tbody>
<tr>
<td>Books</td>
<td>2370617</td>
<td>905267</td>
<td>4</td>
<td>14651960</td>
</tr>
<tr>
<td>Home and Kitchen</td>
<td>450500</td>
<td>182065</td>
<td>14</td>
<td>5314124</td>
</tr>
<tr>
<td>Sports and Outdoors</td>
<td>540695</td>
<td>312835</td>
<td>4</td>
<td>2975253</td>
</tr>
</tbody>
</table>

Table 3.2 provides a high-level overview of the characteristics of each category, and Figures 3.1, 3.2, and 3.3 show examples of BSP, NBSP, and WSP product images (in that order) from the three categories. For the purposes of Table 3.2, an eligible product is one with a non-zero-length product description, a sales rank in the applicable category, a product title, and an image URL.

Figure 3.1: Sample product images from the Books category
For each category, I conducted binary classification experiments (detailed in the next chapter) between BSPs and NBSPs to test how the models perform against the general population of products. I also conducted experiments between BSPs and WSPs to see how the models handle more polarized data. These models perform better according to concrete measures, as expected, but they are more limited in their utility. For example, The BSP vs. WSP models can help a seller or consumer predict whether the product is similar to WSPs and if it’s likely to flop, but these models may be less useful for predicting how the product will fare against the general population of products in the category and whether it will supersede sales of competing products. This will also be discussed in greater detail in the next chapters.
CHAPTER 4
METHODS

4.1 MCS Overview

As components of the proposed MCS, five classifiers operate on two input sources: product descriptions and product images. The MCS includes four text-based models and one image-based model:

- Image: CNN
- Text: Naïve Bayes
- Text: Support Vector Machine
- Text: Random Forest
- Text: Ridge Regression

Each of these models was selected for its reliable, strong performance for text classification, or image classification in the case of the CNN. As previously stated, the models that make up the MCS operate on two fundamentally different inputs: product descriptions and product images. There is a one-to-one relationship between the ASIN—which identifies the product listing—and the product image. This same relationship exists between each ASIN and product description. This means each product listing in the dataset has valid input for the text models and the image model. Ensuring these one-to-one relationships is important for evaluating and combining the models, so product listings in the dataset with blank descriptions or unavailable images are omitted.
4.1.1 Text CNNs and Other Models

CNNs are also well-suited to the task of sentence classification as shown in [32] and [33], but I opted not to include a text-based CNN as this approach showed weak results on the dataset. Product descriptions in the dataset span a much greater range of lengths than individual sentences, where text-based CNNs excel. A product description may be just a couple or words or more than 2,000 characters. This range proved to be unsuitable for the text-based CNN approaches I studied. I explored two of the common techniques to fixing this problem: padding and truncation. Both techniques lead to unimpressive results. Padding introduces significant sparsity, which affected filters negatively, and truncation is inherently lossy. The text-based CNN model showed worse—or at the very least, inefficient—performance across all categories. On average, it performed slightly worse than the image CNN.

In future work, Recurrent Neural Networks and Reinforcement Learning approaches should be considered as well, as research into using of these techniques for text classification has surged in recent years [18, 34, 35, 36, 37, 38]. Pryzant et al. have also demonstrated that adversarial neural networks are well-suited to this problem [24]. In addition, further tinkering may yield improved results with CNNs for text, and alternative feature representations for this type of model should be considered [25].
4.2 Model Architectures

4.2.1 Text Models

For all four text models, product descriptions are vectorized using term-frequency inverse document-frequency (“TF-IDF”). This system of feature representation is similar to a count-based vectorization approach, but it accords greater importance to terms that appear frequently in an individual document and do not appear frequently in the rest of the collection. For each of the four models, TF-IDF is calculated as follows and the results are scaled with the l2-norm [39]:

\[
\text{TF-IDF}(t, d) = (1 + \log(tf)) \times \log \frac{1 + n}{1 + df(d, t)} + 1
\]

(4.1)

where:

\( t \) = term

\( d \) = document

\( n \) = number of documents in the collection

\( tf \) = number of times \( t \) appears in \( d \)

\( df \) = number of documents in the collection that contain \( t \)

A Naïve Bayes Classifier, which employs a probabilistic approach, is included in the MCS for its reliably good performance and its popularity as a baseline for text classification [36, 40]. While overshadowed by the other text classifiers in the MCS, the Naïve Bayes Classifier is quick to train and contributes model diversity to the system. In the MCS, the classifier used is a Multinomial Naïve Bayes Classifier.

A Ridge Regression Classifier is a linear model regularized by the Euclidean norm that is very successful in situations with high-dimensional data [41]. While simplistic, this model, optimized with stochastic average gradient descent, ends up being the strongest individual classifier of the MCS for multiple experiments.

Random Forest Classifiers represent a form of ensemble learning on their own. Random
forests combine multiple decision trees (64 trees in this MCS), and in doing so protect against the problem of overfitting [42]. In multiple instances, random forests have been shown to excel in text classification problems [43, 44, 45].

Support Vector Machines are efficient and consistently perform well for text classification [45, 46] and have been shown to be effective as part of hybrid models [47, 48]. SVMs are also the subject of many current research efforts related to ecommerce, including topics such as knowledge transfer and sentiment analysis [49, 50]. They seek to find a maximum-margin hyperplane, or decision boundary, so that predictions on unseen data can be made with maximum confidence. SVMs can be trained quickly by taking advantage of the “kernel trick” [51, 52, 53]. For all of the SVM experiments in this work, I use a linear kernel to find the optimal separating hyperplane in the data.

4.2.2 Image CNN

Figure 4.1: Architecture of LeNet-5, One of the First Widely Successful CNNs

Convolutional Neural Networks and their variations are widely recognized as state-of-the-art performers for the task of image classification [54, 55]. LeNet-5, proposed by LeCun et al. and depicted in Figure 4.1, is one of the network architectures that lead to the widespread adoption of CNNs for image classification and other tasks. Convolutional neural networks can “learn complex, high-dimensional, non-linear mappings” and “they have no built-in invariance with respect to translations, or local distortions of the inputs,” so they are ideal for image classification [56]. 4.1 shows the essential components of a
CNN: convolutions, subsampling, and one or more fully connected layers. For the purpose
of this work, I use a modified architecture with $128 \times 128$ inputs, 3 convolutional layers
with the ReLu activation function and 64, 32, and $32 \times 3$ filters, respectively, followed by
a 64-neuron fully-connected layer, with softmax applied to generate the final outputs. The
activation maps of each layer use $2 \times 2$ max-pooling (subsampling) and the filters have a
stride of 1. 0-value padding is used around the edges.

4.2.3 Image CNN

4.3 Model Combination

I employ a parallel ensemble learning scheme with simple majority voting, or averaging, for
model combination, wherein the binary classification results are summed and the output of
the MCS depends on the sum crossing a threshold of three. A sum of three to five indicates
a consensus among the models that a sample belongs to the positive class—BSP. A sum of
zero to two indicates a consensus among the models that a sample belongs to the negative
class—NBSP or WSP, depending on the context.

$$H_{MCS}(x) = \begin{cases} 1, & \text{if } \sum_{i=1}^{5} h_i(x) \geq 3 \\ 0, & \text{otherwise} \end{cases}$$  \hspace{1cm} (4.2)

While there are many approaches to model combination, preliminary investigation into
some approaches—stacking, for example—found little difference compared to a majority
voting approach. Majority voting is especially appealing given the tie-breaking nature of
a five-model approach. Since the models rely on fundamentally different learning algo-
rithms rather than repetitions of the same type of classifiers, the MCS can be considered a
heterogeneous ensemble.
5.1 Single-Model Experiment Results

Multiple binary classification experiments were performed on subsets of the data from the selected Amazon product categories. For each model (except the Image CNN), one class of experiments operated on a set of 20,000 BSPs as samples of the positive class and 20,000 NBSPs as samples of the negative class. The second class of experiments operated on 20,000 BSPs as in the previous experiments, but used 20,000 WSPs instead of NBSPs for negative samples. These product images and descriptions are more disparate in relation to the BSPs than they are in relation to the NBSPs, and accordingly, we observe better results.

In each experiment iteration, 36,000 samples (90%) were used for training, and the remaining 4,000 samples (10%) were used for testing. 10-fold cross-validation was performed on the four text models. The same experiments were performed with the Image CNN, but with a reduced set of 10,000 training samples and the same 4000 test samples. Tables 5.1 through 5.6 below summarize single-model performance in terms of F1 score of positive (BSP) classification, F1 score of negative (NBSP or WSP, depending on the circumstance) classification, and average F1 score. These results are discussed in detail in Section 5.3.
5.1.1 Books

Table 5.1: Single-Model Results for Books Category: BSP vs. NBSP

<table>
<thead>
<tr>
<th>Classifier</th>
<th>F1 (Positive)</th>
<th>F1 (Negative)</th>
<th>F1 (Average)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Image CNN</td>
<td>.4407</td>
<td>.6579</td>
<td>.5492</td>
</tr>
<tr>
<td>Naïve Bayes</td>
<td><strong>.6985</strong></td>
<td>.6109</td>
<td>.6547</td>
</tr>
<tr>
<td>Random Forest</td>
<td>.6677</td>
<td>.6752</td>
<td>.6714</td>
</tr>
<tr>
<td>Ridge Regression</td>
<td>.6812</td>
<td><strong>.6747</strong></td>
<td><strong>.6780</strong></td>
</tr>
<tr>
<td>SVM</td>
<td>.6764</td>
<td>.6670</td>
<td>.6717</td>
</tr>
</tbody>
</table>

Table 5.2: Single-Model Results for Books Category: BSP vs. WSP

<table>
<thead>
<tr>
<th>Classifier</th>
<th>F1 (Positive)</th>
<th>F1 (Negative)</th>
<th>F1 (Average)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Image CNN</td>
<td>.7113</td>
<td>.5442</td>
<td>.6277</td>
</tr>
<tr>
<td>Naïve Bayes</td>
<td>.7966</td>
<td>.7458</td>
<td>.7712</td>
</tr>
<tr>
<td>Random Forest</td>
<td>.7936</td>
<td>.7949</td>
<td>.7942</td>
</tr>
<tr>
<td>Ridge Regression</td>
<td><strong>.8310</strong></td>
<td><strong>.8275</strong></td>
<td><strong>.8287</strong></td>
</tr>
<tr>
<td>SVM</td>
<td>.8239</td>
<td>.8211</td>
<td>.8225</td>
</tr>
</tbody>
</table>

5.1.2 Home and Kitchen

Table 5.3: Single-Model Results for Home and Kitchen Category: BSP vs. NBSP

<table>
<thead>
<tr>
<th>Classifier</th>
<th>F1 (Positive)</th>
<th>F1 (Negative)</th>
<th>F1 (Average)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Image CNN</td>
<td>.5482</td>
<td>.5857</td>
<td>.5669</td>
</tr>
<tr>
<td>Naïve Bayes</td>
<td>.6814</td>
<td>.6457</td>
<td>.6636</td>
</tr>
<tr>
<td>Random Forest</td>
<td>.7010</td>
<td><strong>.6898</strong></td>
<td><strong>.6954</strong></td>
</tr>
<tr>
<td>Ridge Regression</td>
<td><strong>.7022</strong></td>
<td>.6885</td>
<td>.6953</td>
</tr>
<tr>
<td>SVM</td>
<td>.6989</td>
<td>.6879</td>
<td>.6934</td>
</tr>
</tbody>
</table>
Table 5.4: Single-Model Results for Home and Kitchen Category: BSP vs. WSP

<table>
<thead>
<tr>
<th>Classifier</th>
<th>F1 (Positive)</th>
<th>F1 (Negative)</th>
<th>F1 (Average)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Image CNN</td>
<td>.6357</td>
<td>.4541</td>
<td>.5449</td>
</tr>
<tr>
<td>Naïve Bayes</td>
<td>.7852</td>
<td>.7571</td>
<td>.7711</td>
</tr>
<tr>
<td>Random Forest</td>
<td><strong>.8352</strong></td>
<td><strong>.8234</strong></td>
<td><strong>.8293</strong></td>
</tr>
<tr>
<td>Ridge Regression</td>
<td>.8231</td>
<td>.8147</td>
<td>.8189</td>
</tr>
<tr>
<td>SVM</td>
<td>.8242</td>
<td>.8156</td>
<td>.8199</td>
</tr>
</tbody>
</table>

5.1.3 Sports and Outdoors

Table 5.5: Single-Model Results for Sports and Outdoors Category: BSP vs. NBSP

<table>
<thead>
<tr>
<th>Classifier</th>
<th>F1 (Positive)</th>
<th>F1 (Negative)</th>
<th>F1 (Average)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Image CNN</td>
<td>.6092</td>
<td>.4670</td>
<td>.5380</td>
</tr>
<tr>
<td>Naïve Bayes</td>
<td>.7405</td>
<td>.6636</td>
<td>.7020</td>
</tr>
<tr>
<td>Random Forest</td>
<td><strong>.7700</strong></td>
<td><strong>.7906</strong></td>
<td><strong>.7803</strong></td>
</tr>
<tr>
<td>Ridge Regression</td>
<td>.7626</td>
<td>.7532</td>
<td>.7579</td>
</tr>
<tr>
<td>SVM</td>
<td>.7623</td>
<td>.7509</td>
<td>.7568</td>
</tr>
</tbody>
</table>

Table 5.6: Single-Model Results for Sports and Outdoors Category: BSP vs. WSP

<table>
<thead>
<tr>
<th>Classifier</th>
<th>F1 (Positive)</th>
<th>F1 (Negative)</th>
<th>F1 (Average)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Image CNN</td>
<td>.5490</td>
<td>.5973</td>
<td>.5731</td>
</tr>
<tr>
<td>Naïve Bayes</td>
<td>.7949</td>
<td>.7464</td>
<td>.7707</td>
</tr>
<tr>
<td>Random Forest</td>
<td>.8351</td>
<td>.8287</td>
<td>.8319</td>
</tr>
<tr>
<td>Ridge Regression</td>
<td>.8343</td>
<td>.8255</td>
<td>.8299</td>
</tr>
<tr>
<td>SVM</td>
<td><strong>.8379</strong></td>
<td><strong>.8294</strong></td>
<td><strong>.8336</strong></td>
</tr>
</tbody>
</table>

18
5.1.4 Examining the models

Text Models

Figures 5.3 through 5.8 show the tokens with the highest coefficients (tending to predict BSP) in black and the lowest coefficients (tending to predict NBSP or WSP) in red for the six SVM experiments with TF-IDF vectorization. Figures 5.1 and 5.2 show a sample of how the best and worst tokens of the books category SVMs might vary with count-based—or plain bag-of-words—vectorization instead of TF-IDF. Count-based vectorization in this category lead to a decrease in F1 score of roughly 4 percentage points. While some tokens like seuss, 2012, and 2013 in the books category make sense intuitively, others are more obscure. To elaborate, this dataset extends through 2014, and one can speculate that recent books (i.e. those from 2012 or 2013) would generally sell better than less recent ones. Furthermore, Dr. Seuss is a well-known author. Many of the other tokens are less explainable and likely appear in relatively few samples. The text models may generalize better with a minimum document frequency threshold set during the vectorization process to exclude tokens that appear infrequently.
Figure 5.1: Best and Worst Tokens for Books - BSP vs. NBSP (Count-Based Vectorization)

Figure 5.2: Best and Worst Tokens for Books - BSP vs. WSP (Count-Based Vectorization)
Figure 5.3: Best and Worst Tokens for Books - BSP vs. NBSP

Figure 5.4: Best and Worst Tokens for Books - BSP vs. WSP

Figure 5.5: Best and Worst Tokens for Home and Kitchen - BSP vs. NBSP
Figure 5.6: Best and Worst Tokens for Home and Kitchen - BSP vs. WSP

Figure 5.7: Best and Worst Tokens for Sports and Outdoors - BSP vs. NBSP

Figure 5.8: Best and Worst Tokens for Sports and Outdoors - BSP vs. WSP
Image CNN

Figure 5.9 shows the filter weights for all 64 of the $3 \times 3$ filters in the first convolutional layer of the CNN, run on BSPs vs. NBSPs for the sports and outdoors category. Figure 5.10 shows two test images, the first a racket, and the second a spool a fishing line, after the filters have been applied to the images.

Figure 5.9: Filters for First Convolutional Layer

Figure 5.10: Test Images After Applying Filters from the First Convolutional Layer
5.1.5 Evaluating Text Model Consistency

In order to ensure the stability of the models and the consistency of their results, 10-fold cross-validation was applied to all models except for the image CNN, due to computational costs. Table 5.7 shows the average of positive F1 score and negative F1 score across all ten folds for each category with both experiments (BSP vs. NBSP and BSP vs. WSP) run for all four text models, a total of 240 experiments. Average performance of the models across the ten folds tends to be within a couple of hundredths of fold zero, whose results are fed into the MCS.

<table>
<thead>
<tr>
<th>Category</th>
<th>Classifier</th>
<th>Experiment</th>
<th>F1 (Average)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Books</td>
<td>Naïve Bayes</td>
<td>BSP vs. NBSP</td>
<td>.6629</td>
</tr>
<tr>
<td>Books</td>
<td>Naïve Bayes</td>
<td>BSP vs. WSP</td>
<td>.7605</td>
</tr>
<tr>
<td>Books</td>
<td>Random Forest</td>
<td>BSP vs. NBSP</td>
<td>.6750</td>
</tr>
<tr>
<td>Books</td>
<td>Random Forest</td>
<td>BSP vs. WSP</td>
<td>.7837</td>
</tr>
<tr>
<td>Books</td>
<td>Ridge Regression</td>
<td>BSP vs. NBSP</td>
<td>.6868</td>
</tr>
<tr>
<td>Books</td>
<td>Ridge Regression</td>
<td>BSP vs. WSP</td>
<td>.8218</td>
</tr>
<tr>
<td>Books</td>
<td>SVM</td>
<td>BSP vs. NBSP</td>
<td>.6802</td>
</tr>
<tr>
<td>Books</td>
<td>SVM</td>
<td>BSP vs. WSP</td>
<td>.8176</td>
</tr>
<tr>
<td>Home and Kitchen</td>
<td>Naïve Bayes</td>
<td>BSP vs. NBSP</td>
<td>.6641</td>
</tr>
<tr>
<td>Home and Kitchen</td>
<td>Naïve Bayes</td>
<td>BSP vs. WSP</td>
<td>.7696</td>
</tr>
<tr>
<td>Home and Kitchen</td>
<td>Random Forest</td>
<td>BSP vs. NBSP</td>
<td>.6993</td>
</tr>
<tr>
<td>Home and Kitchen</td>
<td>Random Forest</td>
<td>BSP vs. WSP</td>
<td>.8249</td>
</tr>
<tr>
<td>Home and Kitchen</td>
<td>Ridge Regression</td>
<td>BSP vs. NBSP</td>
<td>.6876</td>
</tr>
<tr>
<td>Home and Kitchen</td>
<td>Ridge Regression</td>
<td>BSP vs. WSP</td>
<td>.8211</td>
</tr>
<tr>
<td>Home and Kitchen</td>
<td>SVM</td>
<td>BSP vs. NBSP</td>
<td>.6841</td>
</tr>
<tr>
<td>Home and Kitchen</td>
<td>SVM</td>
<td>BSP vs. WSP</td>
<td>.8207</td>
</tr>
<tr>
<td>Sports and Outdoors</td>
<td>Naïve Bayes</td>
<td>BSP vs. NBSP</td>
<td>.6971</td>
</tr>
<tr>
<td>Sports and Outdoors</td>
<td>Naïve Bayes</td>
<td>BSP vs. WSP</td>
<td>.7756</td>
</tr>
<tr>
<td>Sports and Outdoors</td>
<td>Random Forest</td>
<td>BSP vs. NBSP</td>
<td>.7783</td>
</tr>
<tr>
<td>Sports and Outdoors</td>
<td>Random Forest</td>
<td>BSP vs. WSP</td>
<td>.8281</td>
</tr>
<tr>
<td>Sports and Outdoors</td>
<td>Ridge Regression</td>
<td>BSP vs. NBSP</td>
<td>.7549</td>
</tr>
<tr>
<td>Sports and Outdoors</td>
<td>Ridge Regression</td>
<td>BSP vs. WSP</td>
<td>.8286</td>
</tr>
<tr>
<td>Sports and Outdoors</td>
<td>SVM</td>
<td>BSP vs. NBSP</td>
<td>.7585</td>
</tr>
<tr>
<td>Sports and Outdoors</td>
<td>SVM</td>
<td>BSP vs. WSP</td>
<td>.8329</td>
</tr>
</tbody>
</table>
5.2 Model Combination Results

After model combination using the simple majority voting scheme discussed in Chapter 4, the results of the proposed MCS improved the best F1 score over any single model in three cases. In the best case, the MCS improved F1 score by up to 2.3% compared to the best-performing single model, ridge regression (books category). Meanwhile, results improved by more than 51.4% compared to the worst-performing single model, the image CNN (home and kitchen category). Granted, in the case of the home and kitchen category, the individual text models handily outperformed the image CNN as well. Tables 5.8 and 5.9 show MCS performance compared to top performers for fold zero of the ten folds.

<table>
<thead>
<tr>
<th>Category</th>
<th>Best Single Classifier</th>
<th>Best Single Classifier F1</th>
<th>MCS F1</th>
</tr>
</thead>
<tbody>
<tr>
<td>Books</td>
<td>Ridge Regression</td>
<td>.6780</td>
<td>.6935</td>
</tr>
<tr>
<td>Home and Kitchen</td>
<td>Random Forest</td>
<td>.6954</td>
<td>.7015</td>
</tr>
<tr>
<td>Sports and Outdoors</td>
<td>Random Forest</td>
<td>.7803</td>
<td>.7618</td>
</tr>
</tbody>
</table>

Table 5.9: MCS Results - BSP vs. WSP

<table>
<thead>
<tr>
<th>Category</th>
<th>Best Single Classifier</th>
<th>Best Single Classifier F1</th>
<th>MCS F1</th>
</tr>
</thead>
<tbody>
<tr>
<td>Books</td>
<td>Ridge Regression</td>
<td>.8287</td>
<td>.8329</td>
</tr>
<tr>
<td>Home and Kitchen</td>
<td>Random Forest</td>
<td>.8293</td>
<td>.8250</td>
</tr>
<tr>
<td>Sports and Outdoors</td>
<td>SVM</td>
<td>.8336</td>
<td>.8329</td>
</tr>
</tbody>
</table>

5.3 Analysis and Comparison

All four text models performed strongly in all categories, with little variance between different folds. In a broad sense, the results were similar for the books and home and kitchen categories, with the text models for the sports and outdoors category outperforming the text models for the aforementioned categories, particularly in the BSP vs. NBSP experiments.
However, although the text models performed well, the Naïve Bayes Classifier did tend to lag behind the other approaches, which all performed similarly well, with each of the three leading the group in F1 score in at least one experiment. Comparatively, the image CNN performed worse across the board, though it’s important to recognize that this is not a fair, apples-to-apples comparison with the text models. The CNN is operating on fundamentally different data and faces a more challenging task. Furthermore, the purpose of this project is to capitalize on the combined insights of different data types, not to claim which single model is best.

The improved performance for the sports and outdoors category in the task of classifying BSPs vs. NBSPs is likely due to two combined factors: low variance in BSP sales ranks in the sports and outdoors category and product overlap. The highest sales rank in the BSP set of sports and outdoors products is slightly less than 20,000. This means that multiple products share the same sales rank. In some cases, this is due to the second factor: overlap. Products like athletic t-shirts are frequently sold under multiple ASINs with very similar product descriptions, and images for these products tend to be very similar as well, except for attributes such as color. Low variance in the BSP sales ranks means that the BSPs and NBSPs are comparatively more polar than the BSPs and NBSPs in the books or home and kitchen categories [57]. If conceived as clusters, the BSP cluster for sports and outdoors would be much tighter and more clearly defined than the clusters for best-selling books and home and kitchen products. This leads to greater errors for the books category and the home and kitchen category in the BSP vs. NBSP classification task, since the class groupings are more ambiguous.

Predicting product success solely from product images is an inherently more difficult problem. Imagine the challenge a human would face in trying to predict how well a product would sell based on one image from the listing. Nonetheless, this an interesting problem to explore, and the overall results for the image CNN were better than random guessing. To understand this situation better, consider a case where two sellers are both selling pillows
of differing quality with similar featured images. One seller may do a substantially better job of conveying the quality and comfort of the pillow they sell via the keywords they choose to put in the product description. Such differences could reliably be captured by any of the four text models included in the MCS. However, it is much less likely that a seller would be able to convey those same attributes of higher quality in the image of their pillow, especially in a way that could be captured by a relatively simple CNN architecture. Observant consumers may notice and be influenced by certain aspects of image quality such as sharpness or lighting, but a CNN excels at object recognition. In the case of two pillows, it would be hard for a CNN to make a strong prediction based solely on the product images.

On the flip side, the CNN does much better for books. This is natural given the fact that “books” is inherently more restrictive of a category than sports and outdoors. Products in the sports and outdoors category may be basketballs, chairs, cleats, an RV accessory, or one of many other objects. The same applies to the home and kitchen category. Just like with the pillows example, two chairs that look very similar, sold by different companies, may have wildly different sales ranks. On the contrary, the product image for a book will usually feature a colorful, rectangular object with text, and unusually similar covers are more likely to indicate a brand or common publisher (whose sales ranks can tend toward being tightly clustered—see *elsevier* token in Figure 5.4), rather than a “subcategory” of product such as “chair” or “pillow.” This leads to a greater capacity to detect patterns that are influential to sales via a CNN. Since most book images fit a basic archetype, the CNN is detecting features on a more even playing field, whereas in a category like sports and outdoors, the CNN may be discriminating on the basis of product subcategory or object type. It may come to associate a generic, easily-recognizable object such as a basketball with success while disfavoring more obscure products with drastically different image footprints.

Since the image CNN was the worst performer, it invites the question, “Why use it at all?” To answer this question, I recalculated the MCS results on a reduced majority of
three text classifiers (for tie-breaking) and found that the F1 scores tended to be worse by a couple of percentage points. This leads me to conclude that the image CNN, while less impressive in terms of F1 score, plays a valuable role in contributing model diversity to the system and may be particularly useful as a tie-breaking model in the system.

With regard to image quality, there are image processing techniques to detect image quality characteristics as proposed by Zakrewski et al. [25]. In one-to-one product comparisons within a subcategory—say the hypothetical “pillows” subcategory of the parent home and kitchen category—where there isn’t much useful information for distinguishing one product from another, CNNs that don’t account for more complex feature representation may not be the best choice. However, a CNN may excel at identifying larger trends within a general category. For example, from the photos in the BSP and WSP classes of the sports and outdoors category, it seems athletic clothing outperforms handgun holsters in terms of sales rank. An image CNN can reliably tell these objects apart and make a prediction based on the type of item being sold. This insight is certainly more brutish than insight gleaned from the details of a product description, but nonetheless may be useful to a seller with available capital trying to identify a new market to enter within a specific category.
CHAPTER 6
CONCLUSION

6.1 Interpretation of Results

Each of the individual models (except the image CNN) performed well across all tasks. Model performance improved when comparing BSPs to WSPs as expected, and the MCS showed consistent, strong results in all cases, even though it was beaten by a small margin by the top-performing individual model in some situations. The consistency of the experiment results—the MCS was never worse than second place—suggests that sellers and consumers could benefit from a MCS approach instead of opting for a single model. Using a MCS approach ensures that insights gained from a variety of models are considered to inform a more resilient hypothesis. The system of models tested in these experiments yields favorable results and offers decision-informing results to sellers and consumers alike.

For sellers, these results come in the form of feedback predicting whether their product(s) will succeed based on furnished descriptions and images. An advantage of this system is that this feedback can be received before ever posting a product listing to be viewed by the public. An application implemented with the approach detailed herein could help sellers make sure they only ever present a potential consumer with an optimized version of their product listing. Furthermore, a model that operates on seller-provided information such as product descriptions and images rather than consumer-provided information such as product reviews can capitalize on quick adjustments and short turn-around time. A seller can update a product description on the spot, whereas implementing feedback from user reviews could require retooling an entire manufacturing process. Even after these changes are implemented, it will take considerable time before they are recognized and highlighted in new user feedback.
For potential customers, the output of the MCS is another variable that can help them make the right decision, enabling them to avoid products that don’t meet their needs, especially when additional information about that product is scarce. In a situation where a product is on its way to becoming a bestseller, the MCS could provide confirmation of quality asserted by the existing user reviews.

6.2 Future Work

One of the keys to improving work in this area is better access to data. In order to maintain consistency of the models for the selected categories, I restricted the total number of samples to 40,000 for each category—home and kitchen had the lowest number of products and was thus the weakest link. If the proposed individual models could operate on a dataset one or two orders of magnitude larger, the results could be drastically improved, and the model diversity inherent in the MCS may lead to even bigger improvements. Additionally, access to a broader range of data from more products on Amazon and from other ecommerce platforms would be useful in verifying the stability of the tested models and whether certain platforms lend themselves to wider disparities between BSPs and NBSPs or WSPs. In this work, I restricted the number of training samples to facilitate simple creation of balanced datasets. By taking advantage of techniques for dealing with imbalanced data, future models may be able to train on datasets with a larger proportion of NBSPs, though the BSPs necessarily need to stay restricted to prevent dilution of the definition of the term.

Another important consideration for future work is how to avoid abuse of the proposed system. If this MCS were to be made available at no cost or low cost for end users, the models could quickly fall victim to sellers trying to game the system with keyword spam, especially for models where feature representation is built on token frequency. Future researchers may want to consider other aspects of writing quality that would be harder for malicious agents to manipulate.

Finally, some specific approaches to improving the system that should be considered
have already been mentioned, but bear repeating. These include newer, more cutting-edge
text classification techniques such as recurrent neural networks (“RNN”) with long short-
term memory (“LSTM”), reinforcement learning, and generative adversarial neural net-
works [58, 59]. These approaches, combined with image processing techniques designed
to detect specific characteristics of quality, could yield promising results.
Appendices
The following programming libraries, frameworks, and guides were essential in performing the experiments carried out for this work:

- Scikit-learn [39]
- Tensorflow [60]
- “Implementing a CNN for Text Classification in TensorFlow” [61]
- “Tensorflow Image Classification” [62]
- “Visualising Top Features in Linear SVM with Scikit Learn and Matplotlib” [63]
REFERENCES


