THE UNIVERSITY of EDINBURGH

This thesis has been submitted in fulfilment of the requirements for a postgraduate degree (e.g. PhD, MPhil, DClinPsychol) at the University of Edinburgh. Please note the following terms and conditions of use:

- This work is protected by copyright and other intellectual property rights, which are retained by the thesis author, unless otherwise stated.
- A copy can be downloaded for personal non-commercial research or study, without prior permission or charge.
- This thesis cannot be reproduced or quoted extensively from without first obtaining permission in writing from the author.
- The content must not be changed in any way or sold commercially in any format or medium without the formal permission of the author.
- When referring to this work, full bibliographic details including the author, title, awarding institution and date of the thesis must be given.

Weakly Supervised Sentiment Analysis and Opinion Extraction

Stefanos Angelidis

Doctor of Philosophy
Institute for Language, Cognition and Computation
School of Informatics
University of Edinburgh
2019

Abstract

In recent years, online reviews have become the foremost medium for users to express their satisfaction, or lack thereof, about products and services. The proliferation of user-generated reviews, combined with the rapid growth of e-commerce, results in vast amounts of opinionated text becoming available to consumers, manufacturers, and researchers alike. This has fuelled an increased focus on automated methods that attempt to discover, analyze, and distill opinions found in text.

This thesis tackles the tasks of fine-grained sentiment analysis and aspect extraction, and presents a unified framework for the summarization of opinions from multiple user reviews. Two core concepts form the basis of our methodology. Firstly, the use of neural networks, whose ability to learn continuous feature representations from data, without recourse to preprocessing tools or linguistic annotations, has advanced the state-of-the-art of numerous Natural Language Processing tasks. Secondly, our belief that opinion mining systems applied to real-life applications cannot rely on expensive human annotations and should mostly take advantage of freely available review data.

Specifically, the main contributions of this thesis are: (i) The creation of OPOSUM, a new Opinion Summarization corpus which contains over one million reviews from multiple domains. To test our methods, we annotated a subset of the data with fine-grained sentiment and aspect labels, as well as extractive gold-standard opinion summaries. (ii) The development of two weakly-supervised hierarchical neural models for the detection and extraction of sentiment-heavy expressions in reviews. Our first model composes segment representations hierarchically and uses an attention mechanism to differentiate between opinions and neutral statements. Our second model is based on Multiple Instance Learning (MIL), and can detect user opinions of potentially opposing polarity. Experiments demonstrate significant benefits from our MIL-based architecture. (iii) The introduction of a neural model for aspect extraction, which requires minimal human involvement. Our proposed formulation uses aspect keywords to help the model target specific aspects, and a multi-tasking objective to further improve its accuracy. (iv) A unified summarization framework which combines our sentiment and aspect detection methods, while taking redundancy into account to produce useful opinion summaries from multiple reviews. Automatic evaluation, on our opinion summarization dataset, shows significant improvements over other summarization systems in terms of extraction accuracy and similarity to reference summaries. A large-scale judgement elicitation study indicates that our summaries are also preferred by human judges.

Lay Summary

In recent years, online reviews have become the foremost medium for users to express their satisfaction, or lack thereof, about products and services. The proliferation of user-generated reviews, combined with the rapid growth of e-commerce, results in vast amounts of opinionated text becoming available to consumers, manufacturers, and researchers alike. This has fuelled an increased focus on automated methods that attempt to discover, analyze, and distill opinions found in text. This thesis tackles two core tasks related to understanding user reviews: (a) the detection of positive, neutral or negative sentiment in short expressions found in user reviews, and (b) the identification of the aspect of the reviewed entity being discussed in each of these expressions. We introduce novel methods, based on machine learning techniques, to tackle each task independently. These methods are trained using freely available data only, namely reviews, their corresponding user ratings, and keywords that describe product aspects. We combine the outputs of our sentiment and aspect detection methods to identify the most important opinions discussed in a set of reviews about an entity, and construct opinion summaries. Thorough evaluation of our methodology on human annotated review data indicates that our approach produces better sentiment and aspect predictions, and more informative opinion summaries than competitive systems.

Acknowledgements

First and foremost, I want to thank my principal supervisor, Mirella Lapata, for her unlimited support throughout my PhD journey, and beyond. Her unmatched passion for research and deep knowledge of her craft is a constant motivation in my work. I've been very lucky to have her as my mentor and I thank her for trusting and guiding me, through the good and the bad.

I would also like to thank my former supervisor, Victor Lavrenko, who helped me make my first small steps as a researcher. His approach to problem solving has forever influenced the way I think about hard problems, and taught me that having a bad idea is always better than having none.

I am also grateful to Charles Sutton and Timothy Hospedales for their invaluable contributions during the early stages of my PhD. Large parts of this thesis can be traced back to discussions I had with them.

I also thank my examiners, Bing Liu and Adam Lopez, for their feedback and for making my viva an exceedingly positive experience.

Thanks to all the people I have interacted with during my years in ILCC, and especially to the members of the EdinburghNLP group, whose feedback helped me raise the standards of my work. Special thanks to the best office mates one could have: Janie, Philip, Akash, Uche, David, Jane, Victor, Bharat, Siva.

None of this would be possible without my family and friends. Most of all, thanks to my parents for their unconditional love and support.

And lastly, Evita, thank you for standing beside me every step of the way. Thank you for being patient, supporting and loving.

Declaration

I declare that this thesis was composed by myself, that the work contained herein is my own except where explicitly stated otherwise in the text, and that this work has not been submitted for any other degree or professional qualification except as specified.

(*Stefanos Angelidis*)

To Evita, for always being there.

Table of Contents

Chapter 1

Introduction

Everyone is entitled to their own opinion, and this has never felt more true than in our Digital Age. The proliferation of user-generated content on the Web has transformed the way people express attitudes towards aspects of their everyday experience. Social media platforms, like Twitter[1] and Facebook[2], have established themselves as the foremost channels of real-time communication; yet nowhere are personal opinions more densely concentrated than in the domain of *online user reviews*.

Online review interfaces, often found in electronic commerce websites, like Amazon[3], or purpose-built review websites, like Rotten Tomatoes[4] and Yelp[5], allow customers to express their satisfaction about entities of interest. This is a significant shift from the time when reviews were authored exclusively by a small number of professional critics, and has had a huge impact on the entertainment (Duan et al., 2008), hotel (Ye et al., 2009) and commerce (TurnTo Report, 2017) industries. At the same time, the number of reviews written every day grows constantly; more than 27 million Yelp reviews were written in 2017, a figure that has tripled in less than 5 years (Yelp Report, 2017).

The growing availability of such opinion-rich resources has also sparked the interest of Natural Language Processing (NLP) researchers. As users need to navigate through an increasingly large pool of reviews to inform their decisions, *opinion mining* (Pang and Lee, 2008), i.e., the set of problems relating to the automatic analysis and summarization of opinions in text, has received significant attention. The literature

[1] `http://www.twitter.com`

[2] `http://www.facebook.com`

[3] `http://www.amazon.com`

[4] `http://www.rottentomatoes.com`

[5] `http://www.yelp.com`

has predominantly dealt with two core questions when analyzing reviews: *"What are people talking about?"*, and *"How are they talking about it?"*

The methods relating to the former question tackle variants of the *aspect detection* problem (Popescu and Etzioni, 2005; Titov and McDonald, 2008b; Li et al., 2010; He et al., 2017). Aspects are identifiable characteristics of a product or service which are likely to influence user satisfaction. For example, *image quality* and *connectivity* are two characteristics of televisions, whereas *food* and *ambience* are important aspects of restaurants. Approaches that relate to the second question fall under the umbrella of *sentiment analysis* (Turney, 2002; Pang et al., 2002; Taboada et al., 2011; Socher et al., 2013). In its most popular incarnation, sentiment analysis deals with classifying the *polarity* of a given text, at the phrase (Socher et al., 2013), sentence (Kim, 2014), or document level (Pang et al., 2002; Le and Mikolov, 2014; Yang et al., 2016). Sentiment polarity measures the attitude of a text and takes discrete or continuous values ranging from very negative to very positive.

Aspect and sentiment detection have been studied extensively as standalone problems but have also become particularly useful in the context of opinion summarization, i.e., the aggregation of salient opinions on an entity of interest. The majority of related work aims to create summaries from collections of reviews and decomposes the problem into the two aforementioned subtasks (Hu and Liu, 2004): (a) the detection of aspect-specific expressions, and (b) the prediction of their sentiment polarity. Positive or negative, aspect-specific comments are more likely to express salient opinions. These are used either to produce structured summaries, which indicate the distribution of opinions for each product aspect (Hu and Liu, 2004; Liu et al., 2005), or textual summaries, which may be extractive (Beineke et al., 2004; Carenini et al., 2006) or abstractive (Ganesan et al., 2010; Gerani et al., 2014). An example of an opinion summarization pipeline is shown in Figure 1.1, where opinions about the image quality, sound quality, connectivity, and price of a television are extracted from multiple reviews and grouped based on their polarity, while neutral or redundant comments are discarded. This thesis will present an end-to-end system for the task of extractive opinion summarization, combining novel, neural-based methods which *learn* to detect fine-grained aspect and sentiment in reviews.

The abundance of review data has increased the applicability of machine learning and, more recently, neural networks to opinion analysis (Socher et al., 2013; Tang et al., 2015a; Wang et al., 2016; He et al., 2017). The ability of neural networks to directly learn dense feature representations from labeled or unlabeled text has fueled recent ad-

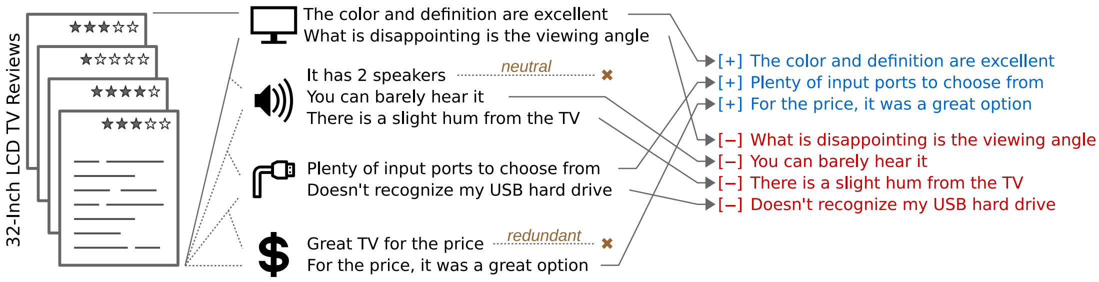

32-Inch LCD TV Reviews
The color and definition are excellent
What is disappointing is the viewing angle
It has 2 speakers
neutral
You can barely hear it
There is a slight hum from the TV
Plenty of input ports to choose from
Doesn't recognize my USB hard drive
Great TV for the price
redundant
For the price, it was a great option
[+] The color and definition are excellent
[+] Plenty of input ports to choose from
[+] For the price, it was a great option
[−] What is disappointing is the viewing angle
[−] You can barely hear it
[−] There is a slight hum from the TV
[−] Doesn't recognize my USB hard drive

Figure 1.1: An extractive opinion summarization pipeline. Opinions on image quality, sound quality, connectivity, and price of an LCD television are extracted from a set of reviews. Their polarities are then used to group them into positive and negative, while neutral or redundant comments are discarded.

vances in the field, and throughout NLP (Mikolov et al., 2013; Pennington et al., 2014; Bahdanau et al., 2015; Cao et al., 2015; Cheng and Lapata, 2016; Dong and Lapata, 2018). In the case of coarse-grained tasks, like document-level sentiment classification, reviews paired with freely available user ratings have been used extensively to train supervised neural models (Tang et al., 2015a; Yang et al., 2016).

However, when the goal is to detect polarity or aspects on a finer granularity, namely in sentences or phrases, supervision does not come for free. Neural approaches have relied either on unsupervised architectures (Yin et al., 2016; He et al., 2017), which are hard to train and often require post-hoc human involvement; or on large-scale data annotation efforts (Socher et al., 2013; Wang and Ling, 2016), which enable fully supervised training but are costly and may not translate across domains or languages.

The work presented in this thesis circumvents these obstacles using weakly supervised learning. Our proposed neural methods rely only on freely available information, in the form of user ratings or product domain labels, and require minimal human intervention, namely a few aspect-denoting words. Specific types of weakly supervised formulations, like Multiple Instance Learning (Pappas and Popescu-Belis, 2014, 2017; Kotzias et al., 2015) or seed-based model initialization (Mukherjee and Liu, 2012) have been previously used for opinion mining. However, to the best of our knowledge, our work presents the first thorough exploration of these ideas in a neural context, and as part of a unified framework for opinion summarization.

1.1 Thesis Statement

In this thesis, we investigate a series of hypotheses relating to the extraction of opinions from reviews, which we test through extensive evaluation and analysis of our methods.

> HYPOTHESIS I: Weakly supervised learning, using signals from freely available information and minimal domain knowledge, is sufficient to train neural networks that can detect fine-grained sentiment and aspects in reviews.

The ability to train otherwise data-hungry neural models, without the requirement of human-annotated data or hard-crafted rules, is a significant step towards establishing deep learning as the method of choice in real-world opinion mining applications.

> HYPOTHESIS II: Sentiment and aspect predictions obtained from these neural networks are good indicators for the extraction of salient comments from collections of reviews, and the generation of useful opinion summaries.

While sentiment and aspect detectors have been previously combined to produce structured opinion summaries (Hu and Liu, 2004), our work is the first to explore their utility for *text-based* opinion summarization, without recourse to gold-standard data.

> HYPOTHESIS III: The granularity of extracted segments is important. Subsentence clauses provide a better basis for opinion summarization, as they communicate more succinct and targeted sentiment.

Previous research has hinted at the benefits of subsentence segmentation for document-level sentiment analysis (Bhatia et al., 2015) and generic summarization (Li et al., 2016). Here, we perform a large-scale judgement elicitation study to investigate if clause-based summaries are preferred by human judges.

1.2 Contributions

This thesis proposes a number of novel methods for the analysis of sentiment polarity, aspect information, and opinion salience in user reviews. Below, we summarize the main contributions of the thesis:

The OpoSum corpus We introduce and make publicly available OpoSum, an Opinion Summarization corpus for the training and evaluation of weakly supervised methods like ours. OpoSum's training set contains over one million reviews from eight diverse domains: *movies*, *local businesses*, *laptop bags*, *bluetooth headsets*, *boots*, *keyboards*, *televisions* and *vacuums*. All reviews are paired with user ratings, which we use to train our weakly supervised sentiment models. OpoSum also contains development and test sets for four complementary subtasks: *fine-grained sentiment detection*, *aspect extraction*, *salient opinion ranking* and *opinion summarization*. For sentiment detection, we annotated 350 reviews across all eight domains with sentence- and clause-level sentiment labels (positive, neutral, and negative). For aspect extraction, salient opinion ranking, and opinion summarization, we annotated 600 reviews across 6 domains on the clause level. Details on the construction of various parts of the corpus are provided in Chapters 3, 4, 5 and 6.

Analysis of Polarity in Reviews We analyze the different ways in which expressions of varying polarity are distributed within reviews. In particular, we propose a categorization of reviews and their constituent sentences based on the distribution of sentiment within them. Then, we use the polarity-annotated portion of OpoSum as a testbed to investigate the proportion of reviews (and sentences) that convey uniform or mixed sentiment, a crucial distinction for our sentiment detection models.

Sentiment Detection Neural Networks We present two neural architectures for the detection of sentiment-heavy expressions in reviews. Firstly, our HierNet model is a hierarchical neural network which composes word and segment representations to predict the sentiment of whole reviews. Its attention mechanism allows it to differentiate between review segments of varying importance, and thus, it is able to identify opinions that correlate with the overall polarity of the review. Secondly, our MilNet model, which is based on the principles of Multiple Instance Learning (Keeler and Rumelhart, 1992; Dietterich et al., 1997), predicts the sentiment of individual segments and then combines those predictions to classify the review as a whole. Contrary to HierNet, MilNet can detect expressions of opposing polarity within a review. We propose a polarity scoring function that facilitates opinion ranking and extraction from single reviews, and evaluate both models against rule-based and fully supervised methods.

Multi-Seed Aspect Extractor We formulate the task of aspect extraction, i.e., the grouping of review segments based on the aspects they discuss, under a weakly supervised setting, which only requires minimal human intervention. We present MATE, a Multi-Seed Aspect Extractor, that is initialized with sets of aspect-denoting keywords and is trained without recourse to human-annotated reviews. Additionally, we propose a multi-tasking variant, which uses a secondary objective to help our model focus on aspect-signaling words. Experiments indicate substantial improvements over a state-of-the-art aspect discovery model (He et al., 2017).

Opinion Extraction Framework We bring all modeling contributions together and propose a unified opinion extraction framework that combines the polarity and aspect predictions of our models. Our multi-review opinion summarizer requires no further training and is, therefore, only reliant on the weak supervision signals used for our sentiment and aspect detectors. We describe three separate opinion salience approaches: one that only uses polarity; one based on aspect predictions; and one that combines both sources of information. We test our approaches on salient opinion ranking, where segments across multiple reviews are ranked according to their significance, and observe that the combination of polarity and aspect predictions yields the best results. The top ranked opinions are then filtered to avoid redundancy, generating 100-word opinion summaries. Evaluation against the reference summaries of our OpoSum corpus, indicates that our extraction method significantly outperforms summarization baselines.

Human Evaluation We also perform two large-scale user studies on the quality of our methods for single- and multi-review summarization. To the best of our knowledge, this is the first published study where opinion summaries from different systems are judged according to multiple criteria.

1.3 Thesis Outline

Chapter 2 acts as an introduction to neural modeling from the perspective of sentiment analysis. We first provide a simplistic definition of fully supervised sentiment classification, where the text to be classified is viewed as a sequence of words. We then present a series of neural models of increasing complexity: a Convolutional Neural Network (Kim, 2014), and three variants of a Recurrent Neural Network (Hochreiter and Schmidhuber, 1997; Cho et al., 2014). The purpose of this chapter is to familiarize the reader with basic neural concepts, which we use as building blocks for our weakly supervised models, presented in following chapters.

Chapter 3 focuses on review data, as we describe the different ways in which sentiment is expressed. We define a categorization of reviews and review sentences based on the variation of sentiment polarity within them, and explain how certain methods may struggle when opinions of opposing polarity are intermixed. We then present the first part of our opinion summarization corpus, OPOSUM. It includes large-scale training collections of reviews from multiple domains, and development and test sets annotated with fine-grained sentiment labels on two levels of granularity: sentences and clauses. We use our human-annotated data to investigate the sentiment uniformity, or lack thereof, in reviews.

Chapter 4 presents our weakly supervised neural models for the detection of fine-grained sentiment in reviews. Both models are based on architectures that gradually compose sentiment information up the review hierarchy, and are trained using freely-available document-level labels only. Our first model, HIERNET, uses vector-based composition, and lacks the ability to identify opinions of opposing polarity. In contrast, our MILNET model, which is based on Multiple Instance Learning (Keeler and Rumelhart, 1992; Dietterich et al., 1997), overcomes this restriction by first predicting the sentiment of individual sentences or clauses, and then combining these predictions. After training, both models use an attention-based polarity scoring technique to identify salient opinions and produce single-review opinion summaries. MILNET outperforms HIERNET and other baselines in three sentiment detection tasks and produces summaries that are preferred by human judges.

Chapter 5 presents our aspect extraction methodology. We first describe the different approaches one might take towards identifying aspect-specific expressions in reviews. We discuss a previously proposed neural topic model for the task (He et al., 2017) and point out its shortcomings, namely the requirement for post-hoc interpretation of induced topics. We then move on to our Multi-Seed Aspect Extractor (MATE), which is able to target specific product aspects through the use of aspect seed words. We evaluate MATE, and a multi-tasking variant, on the aspect portion of OPOSUM, revealing significant improvements over previous work.

Chapter 6 arrives at the ultimate goal of this thesis, the extraction of opinion summaries from multiple reviews. We present three approaches for the identification of salient opinions, which are based on the predictions of our neural models: a polarity-based score, an aspect-based score, and one that combines both polarities and aspects. Automatic and human evaluation shows that our combined extraction method produces opinion summaries of higher quality when compared against multiple summarization baselines.

Chapter 7 summarizes our main findings, and concludes this thesis by discussing limitations of our work and possible directions for future research.

1.4 Published Work

Chapter 4 is largely based on the work presented in Angelidis and Lapata (2018a), but has been extended to include further experiments. Parts of Chapters 3 and 4 have been published in Angelidis and Lapata (2018b).

Chapter 2

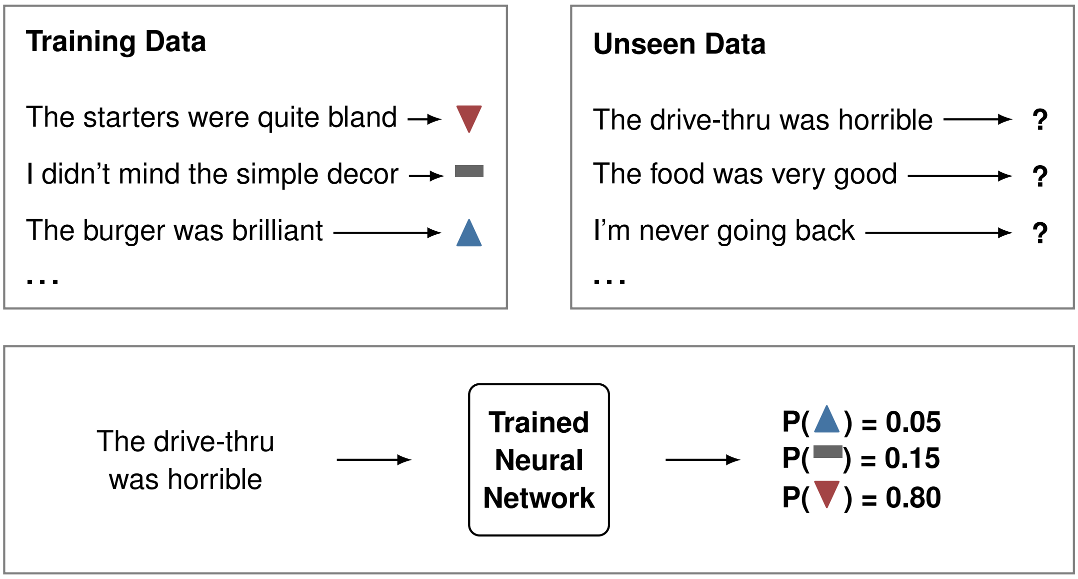
Background: Neural Sentiment Analysis

The revival of neural networks in the past decade has had unprecedented impact in NLP research, setting new standards in a wide range of language understanding and generation applications. Multiple factors have fuelled this, including the refinement of neural modeling methods, advances in computational power, and the introduction of large-scale training corpora.

Neural networks' ability to learn rich representations directly from text has freed NLP from the burden of hand-crafted features and specialized preprocessing tools. Instead, information about the semantic content (Kim, 2014; Yang et al., 2016; Cheng et al., 2016), syntax (Chen and Manning, 2014; Dyer et al., 2015), and discourse structure (Kalchbrenner and Blunsom, 2013; Zhang et al., 2015) of written language can be encoded in continuous vectors which are passed on to higher levels of neural computation, specifically designed for the task at hand.

In this chapter, we present a brief introduction to the task of sentiment analysis from the perspective of neural modeling. In particular, we provide a simplistic description of the task, and present a series of neural architectures of increasing complexity that attempt to encode the sentiment of texts and predict their polarity.

The main purpose of this chapter is to familiarize the reader with basic neural modeling concepts in the context of supervised sentiment analysis. We present core ideas, like convolution, sequence modeling, and attention, that will form the basis of the weakly supervised models presented in Chapters 4 and 5.



Training Data

The starters were quite bland → ▼

I didn't mind the simple decor → ▬

The burger was brilliant → ▲

...

Unseen Data

The drive-thru was horrible → ?

The food was very good → ?

I'm never going back → ?

...

The drive-thru was horrible → **Trained Neural Network** →

P(▲) = 0.05
P(▬) = 0.15
P(▼) = 0.80

Figure 2.1: A toy example of supervised sentiment analysis. A training set (top left) of positive (▲), neutral (▬), and negative (▼) sentences is used to train a neural classifier. The trained model (bottom) can predict the sentiment of unseen instances (top right).

2.1 Supervised Sentiment Analysis

Sentiment analysis, i.e., the detection of sentiment orientation in natural language text, is most commonly modeled as a fully supervised classification task. As such, it requires a dataset C of training instance-label pairs $\{(s_i, y_i)\}_{i=1}^{|C|}$, where s_i is a text unit (phrase, sentence or document) and y_i is its true sentiment label. Instance labels y_i take values from an ordered label-set $[1, L]$, where 1 and L denote maximally negative and positive sentiment, respectively. Furthermore, instance s_i is viewed as a *sequence* of words $(w_1, w_2, \dots, w_n)$, where n is the sequence length.

Figure 2.1 shows a toy example of a sentiment analysis training set (top left), which consists of sentences, paired with one of three sentiment labels; *positive*, *neutral*, or *negative*. The goal is to train a classifier (bottom) and use it to predict the sentiment of unseen test instances (top right).

A sentiment classifier, parameterized by θ, will produce a probability distribution $\mathbf{p}_i$ over sentiment labels, and classify s_i by selecting the most probably one:

$$\mathbf{p}_i = \langle p_i^{(1)}, \dots, p_i^{(L)} \rangle \,, \tag{2.1}$$

$$p_i^{(c)} = P_\theta(y_i = c | w_1, \dots w_n) \,. \tag{2.2}$$

In the case of neural modeling, parameters θ are defined by the network's architecture and learned via backpropagation. The rest of the chapter describes such architectures.

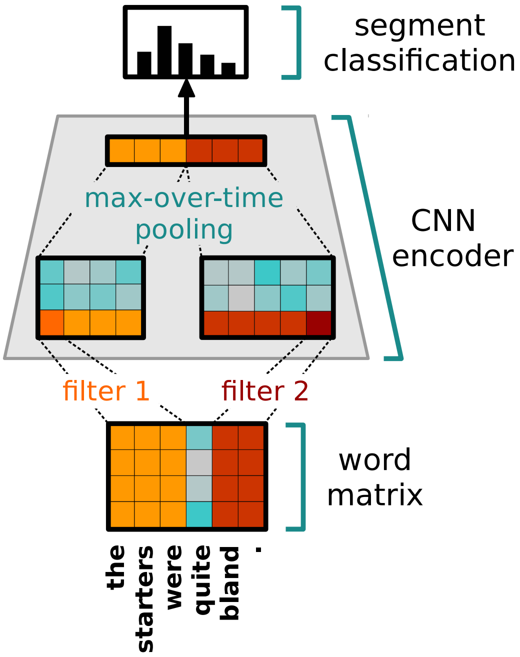


Figure 2.2: A *Convolutional Neural Network* (CNN) for sentiment classification.

2.2 Convolutional Neural Network

Convolutional Neural Networks (CNNs) have significantly advanced the field of Computer Vision and have been adapted with great success to various NLP tasks. Here, we present a typical CNN model for sentiment classification, which is largely based on the work of Kim (2014) for modeling sentences.

Let $\mathbf{x}_j$ denote a k-dimensional word embedding (Mikolov et al., 2013; Pennington et al., 2014) of the j-th word in text sequence s_i of length n. Word embeddings are dense, low-dimensional, and commonly pre-trained word representations that have been shown to boost performance across NLP tasks. The segment's input representation is the concatenation of word embeddings $\mathbf{x}_1, \dots, \mathbf{x}_n$, resulting in word matrix $\mathbf{X}$, as shown at the bottom of Figure 2.2. Let $\mathbf{X}_{j:j+l}$ refer to the concatenation of embeddings $\mathbf{x}_j, \dots, \mathbf{x}_{j+l}$. A convolution filter $\mathbf{W} \in \mathbb{R}^{l \times k}$, applied to a window of l words, produces a new feature:

$$c_j = \text{ReLU}(\mathbf{W} \circ \mathbf{X}_{j:j+l} + b)\,, \tag{2.3}$$

where ReLU is the *Rectified Linear Unit* non-linearity (Nair and Hinton, 2010), '$\circ$' denotes the entrywise product followed by a sum over all elements, and $b \in \mathbb{R}$ is a bias term. Applying the same filter to every possible window of words in the segment, produces a feature map $\mathbf{c} = [c_1, c_2, \dots, c_{n-l+1}]$, like the ones shown in horizontal orange and red rows in Figure 2.2. Multiple feature maps for varied window sizes are applied, resulting in a fixed-size segment representation $\mathbf{v}$, via max-over-time pooling. We will refer to the application of convolution to an input word matrix $\mathbf{X}$, as $\mathbf{v} = \mathrm{CNN}(\mathbf{X})$. Vector $\mathbf{v}$ is fed into a softmax classifier to produce a sentiment prediction, shown as a probability bar chart in Figure 2.2:

$$\mathbf{p}_i = \mathrm{softmax}(\mathbf{W}_c\mathbf{v} + \mathbf{b}_c)\,. \tag{2.4}$$

The CNN is trained using the *Negative Log-Likelihood* (NLL) of the predictions:

$$L = -\sum_i \log p_i^{(y_i)}\,. \tag{2.5}$$

2.3 Recurrent Neural Networks

CNNs are capable of encoding a variable-length sequence of words into a fixed-length representation without significant computational requirements, but lack the ability to model long-range temporal dependencies. Recurrent Neural Networks (RNNs), on the other hand, utilize an internal hidden state and combine inputs at each time-step with previously stored information. RNN models, namely those based on *Long Short-Term Memory* (LSTM; Hochreiter and Schmidhuber 1997) and *Gated Recurrent Units* (GRU; Cho et al. 2014), have been used successfully for modeling text sequences in a plethora of tasks (Bahdanau et al., 2015; Tang et al., 2015a; Filippova et al., 2015; Kim et al., 2016; Yang et al., 2016; Cheng and Lapata, 2016). GRUs are computationally more efficient, have been shown to perform on par with LSTMs (Chung et al., 2014), and are used throughout this thesis. The following models utilize them to encode and classify sequences of words. We present three variants; a basic, unidirectional RNN; a bidirectional RNN with average or max pooling; and a bidirectional RNN with attention.

2.3.1 Simple GRU

In the simplest case, an RNN will encode the semantic content of a word sequence by feeding word embeddings one-by-one in a single direction (left-to-right) and combining them with information already stored in the GRU from previous time-steps. The process is illustrated in Figure 2.3 and described in detail below.

At time-step j, where w_j's vector is fed into the model, the GRU computes a new hidden state as:

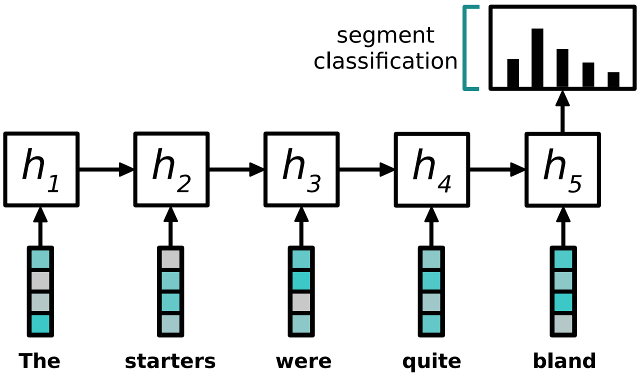


Figure 2.3: A unidirectional GRU-based *Recurrent Neural Network* (RNN) for sentiment classification.

$$\mathbf{h}_j = (1 - \mathbf{z}_j) \odot \mathbf{h}_{j-1} + \mathbf{z}_j \odot \tilde{\mathbf{h}}_j\,, \tag{2.6}$$

where $\odot$ is the element-wise multiplication operator. This is a linear interpolation between the previous hidden state vector $\mathbf{h}_{j-1}$ and the newly computed $\tilde{\mathbf{h}}_j$. Gate vector $\mathbf{z}_j$ controls the flow of temporal information by deciding how much of the past hidden state will be kept, and is computed as:

$$\mathbf{z}_j = \sigma(\mathbf{W}_z \mathbf{x}_j + \mathbf{U}_z \mathbf{h}_{j-1} + \mathbf{b}_z)\,, \tag{2.7}$$

where $\mathbf{x}_j$ is the word vector of input word w_j. Candidate hidden state $\tilde{\mathbf{h}}_j$ is:

$$\tilde{\mathbf{h}}_j = \tanh(\mathbf{W}_h \mathbf{x}_j + \mathbf{r}_j \odot (\mathbf{U}_h \mathbf{h}_{j-1}) + \mathbf{b}_h)\,, \tag{2.8}$$

where $\mathbf{r}_j$ is the reset gate vector:

$$\mathbf{r}_j = \sigma(\mathbf{W}_r \mathbf{x}_j + \mathbf{U}_r \mathbf{h}_{j-1} + \mathbf{b}_r)\,. \tag{2.9}$$

We refer to the application of the GRU on the i-th input word as $\mathbf{h}_j = \mathrm{GRU}(\mathbf{x}_j)$. Once the whole sequence of length n has been processed, we obtain n hidden vectors $(\mathbf{h}_1, \dots, \mathbf{h}_n)$. In this case, we use the final hidden vector, $\mathbf{h}_n$, to represent the whole sequence and a softmax classifier to predict the segment's sentiment, as shown in Figure 2.3. The model is trained using NLL, similarly to the CNN.

2.3.2 Bidirectional GRU

We can take advantage of richer contextual information, by extending our GRU model to encode sequences in both directions. Such a *bidirectional* GRU will produce hidden

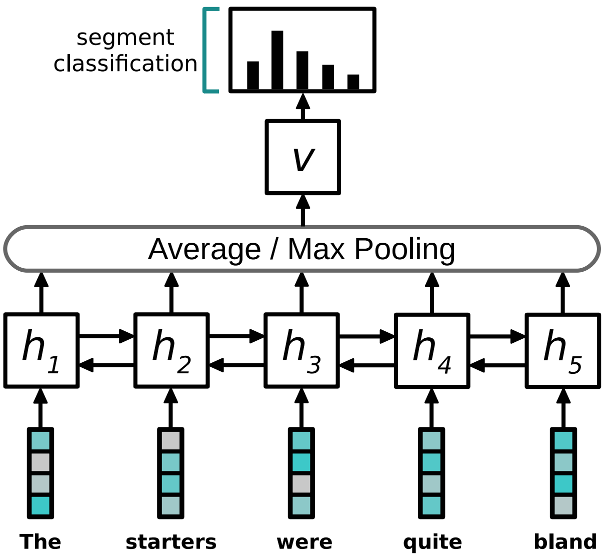


Figure 2.4: A bidirectional GRU-based *Recurrent Neural Network* (RNN) for sentiment classification.

vectors for either direction independently, and concatenate them:

$$\overrightarrow{\mathbf{h}}_j = \overrightarrow{\mathrm{GRU}}(\mathbf{x}_j)\,, \tag{2.10}$$

$$\overleftarrow{\mathbf{h}}_j = \overleftarrow{\mathrm{GRU}}(\mathbf{x}_j)\,, \tag{2.11}$$

$$\mathbf{h}_j = [\overrightarrow{\mathbf{h}}_j, \overleftarrow{\mathbf{h}}_j]\,. \tag{2.12}$$

Then, given the sequence of hidden states $\mathbf{h} = (\mathbf{h}_1, \mathbf{h}_2, \ldots, \mathbf{h}_n)$, we get a final representation $\mathbf{v}$ by averaging across time-steps (*Average Pooling*) or by taking the max along each dimension (*Max Pooling*), as illustrated in Figure 2.4. Again, the final document vector is fed into a *softmax* classifier and the model is trained end-to-end using NLL.

2.3.3 Bidirectional GRU with Attention

A segment representation can be produced by average or max pooling across hidden vectors with minimal computational complexity. This is, however, a crude way of composing feature vectors, as not all words in a text convey important sentiment clues. Instead, we can use an attention mechanism (Bahdanau et al., 2015), which rewards words that are more likely to be good sentiment predictors. Many attention methods have been proposed in literature, depending on the task at hand (Bahdanau et al., 2015; Xu et al., 2015; Gregor et al., 2015; Luong et al., 2015). Here, we describe a technique proposed by Yang et al. (2016), and designed for neural networks that attempt to produce a single representation for a unit of text.

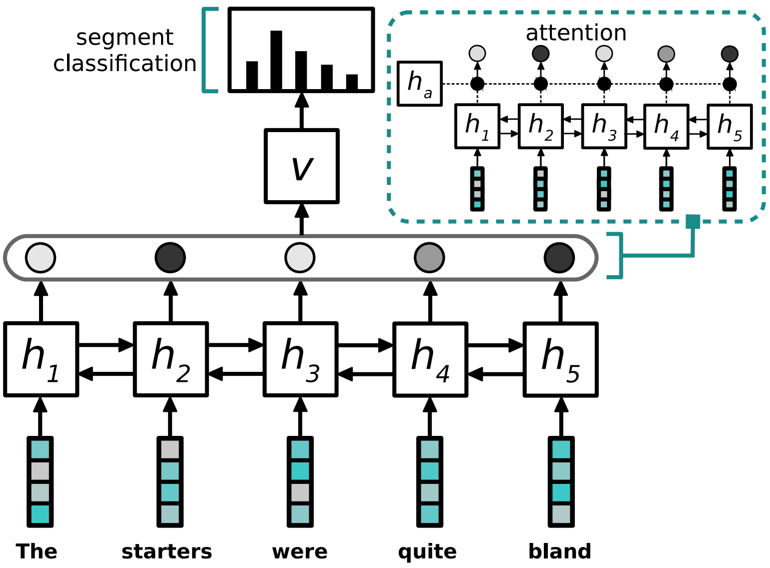


Figure 2.5: A bidirectional GRU-based *Recurrent Neural Network* (RNN) with attention for sentiment classification.

The importance of each word is measured with the aid of a vector $\mathbf{h}_a$, as follows:

$$\mathbf{h}'_j = \tanh(\mathbf{W}_a\mathbf{h}_j + \mathbf{b}_a)\,, \tag{2.13}$$

$$a_j = \frac{exp(\mathbf{h}'^{\mathsf{T}}_j \mathbf{h}_a)}{\sum_j exp(\mathbf{h}'^{\mathsf{T}}_j \mathbf{h}_a)}\,, \tag{2.14}$$

where Equation (2.13) defines a one-layer MLP that produces an attention vector for the i-th word. Attention weights a_j are computed as the normalized similarity of each $\mathbf{h}'_j$ with $\mathbf{h}_a$. Vector $\mathbf{h}_a$ is a global attention vector, which is randomly initialized and learned during training. It can be thought of as a trained *key*, able to recognize sentiment-heavy terms. The attention mechanism is depicted in the dashed box of Figure 2.5, with attention weights shown as shaded circles (darker shade indicates higher attention). The segment is then represented as the attention-weighted average of the hidden vectors:

$$\mathbf{v} = \sum_j a_j \mathbf{h}_j \tag{2.15}$$

Similar neural architectures can be used to hierarchically model whole documents, e.g., a review viewed as sequences of sentences. Each constituent sentence is encoded into a vector using a CNN or RNN, and those sentence vectors are then passed on to a document-level RNN. We explore such neural networks in Chapter 4.

2.4 Summary

In this chapter, we presented an introduction to the task of neural sentiment analysis. We described a fully supervised view of the problem, where word sequences (phrases, sentences or documents) paired with sentiment labels are used to train a sentiment classifier. We then presented a series of neural architectures of increasing complexity for the task. First, a Convolutional Neural Network, which has been applied to sentence classification with great success. CNNs are highly efficient architectures, but are unable to take advantage of long-range temporal dependencies. For this reason, we also presented three variants of a GRU-based Recurrent Neural Network: a simple, unidirectional network; a bidirectional network with an average- or max-pooling layer; and an attention-based bidirectional network.

Both the CNN and RNN architectures will become relevant in later parts of this thesis, as they act as core components of our hierarchical neural networks for the detection of sentiment in reviews, presented in detail in Chapter 4.

In the next chapter, we briefly move away from machine learning, and focus on the epicenter of this thesis; the reviews themselves. We explore the different ways in which sentiment is conveyed within reviews and present the first part of OPOSUM, our opinion summarization dataset.

Chapter 3

Analyzing Polarity in Review Data

The continuous increase in the number of online user reviews produced daily has had a significant influence on sentiment analysis research (Pang and Lee, 2008; Liu, 2012). Neural networks, like the ones presented in Chapter 2, have been successfully applied to several NLP tasks, but require data of high quality and quantity for training.

Before we move on to describe the problems, methodology, and experiments of our work, this chapter focuses on the reviews themselves, as we take a first step towards understanding how sentiment is conveyed in them. We begin by presenting a conceptual overview of how objective and subjective expressions may be combined to communicate opinions (Section 3.1). Using intuitive rules and examples of real reviews, we propose a categorization of reviews and their constituent sentences, based on the distribution of sentiment within them. We hypothesize that the polarity of opinions is non-uniform in all but the most extreme cases, and claim that this has a significant effect on methods that detect fine-grained sentiment.

Section 3.2 introduces OpoSum, a large-scale dataset of reviews, originating from eight diverse domains, which we use throughout this work to train and evaluate our models. OpoSum's training set contains more than one million reviews, paired with their user ratings, but no fine-grained annotations. We also describe our efforts to annotate a held-out set of reviews with polarity labels on the sentence and sub-sentence level, to be used for evaluating our sentiment models. Here, we use these segment polarity annotations to verify our initial hypotheses about sentiment expression and analyze trends across the different domains of the dataset.

3.1 Sentiment in Reviews

Reviews differ from most types of user-generated text on the web, in that they are inherently opinionated. Other means of writing, like newswire, are expected to be objective. Free-form user content in social media posts and message board conversations may express information that ranges from purely factual to highly subjective, depending on the topic of discussion (Barbosa and Feng, 2010). In contrast, user reviews are guaranteed to include the author's subjective views on a product or service.

The subjective nature of reviews stems from their very nature, namely they convey the author's opinions. Opinions are based, to some extent, on objective shortcomings or advantages of the reviewed entity but are largely influenced by the author's own experience and predisposition, which will unavoidably vary for different people. As we shall see in detail in Chapters 4 and 5, a single review can express multiple opinions about different aspects of an entity, and the overall attitude of the reviewer can be thought of as their aggregate.

The sentiment of individual opinions within a review is not necessarily uniform. For example, a very positive review may still contain a number of neutral (i.e., objective) statements. By the same token, one can easily imagine a somewhat negative review containing a positive comment about a particular aspect of the reviewed entity. Ambivalent reviews are likely to include a combination of positive and negative opinions. Furthermore, non-uniform sentiment can manifest itself in shorter text chunks, such as sentences, clauses, even phrases. Understanding how opinions of varied polarity are expressed in reviews is an important first step towards designing successful sentiment analysis and opinion mining systems.

In this section, we focus on how sentiment is distributed within whole reviews (Section 3.1.1) and within single sentences (Section 3.1.2). In particular, we propose a simple categorization of reviews and sentences based on their sentiment uniformity, or lack thereof.

3.1.1 Review Types

We start off by grouping reviews into different types based on the sentiment polarity of the segments that comprise them. For the purpose of this discussion, we define segments to be sentences or sub-sentence clauses that coherently communicate a positive, negative, or neutral statement (see Section 3.2.1.1 for details on review segmentation).

We distinguish between three types of reviews: reviews of *strictly* uniform senti-

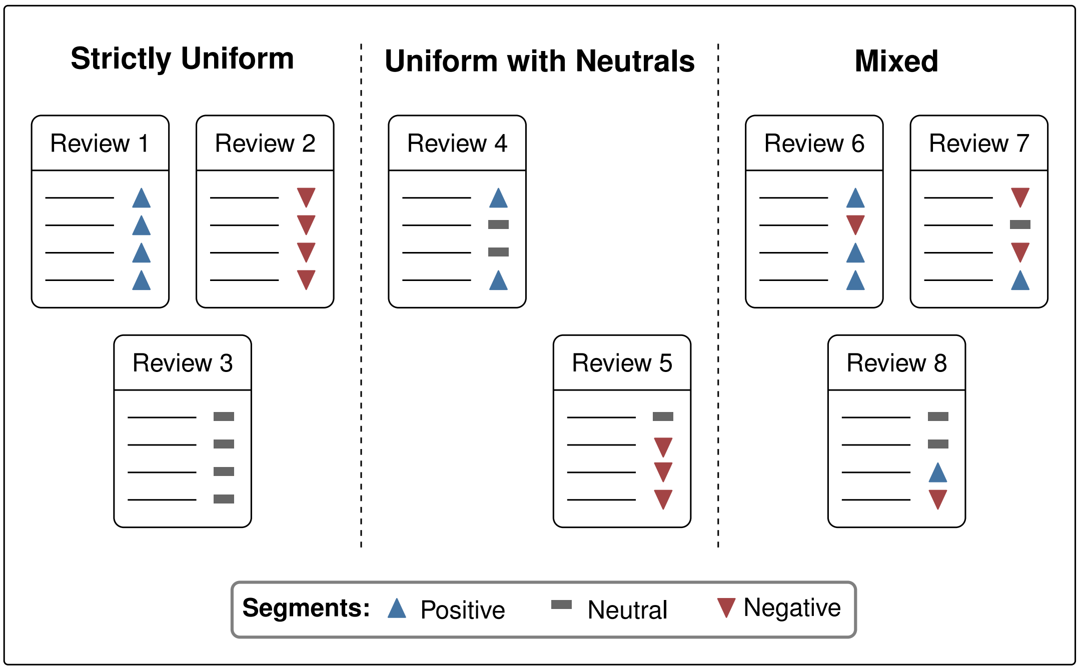


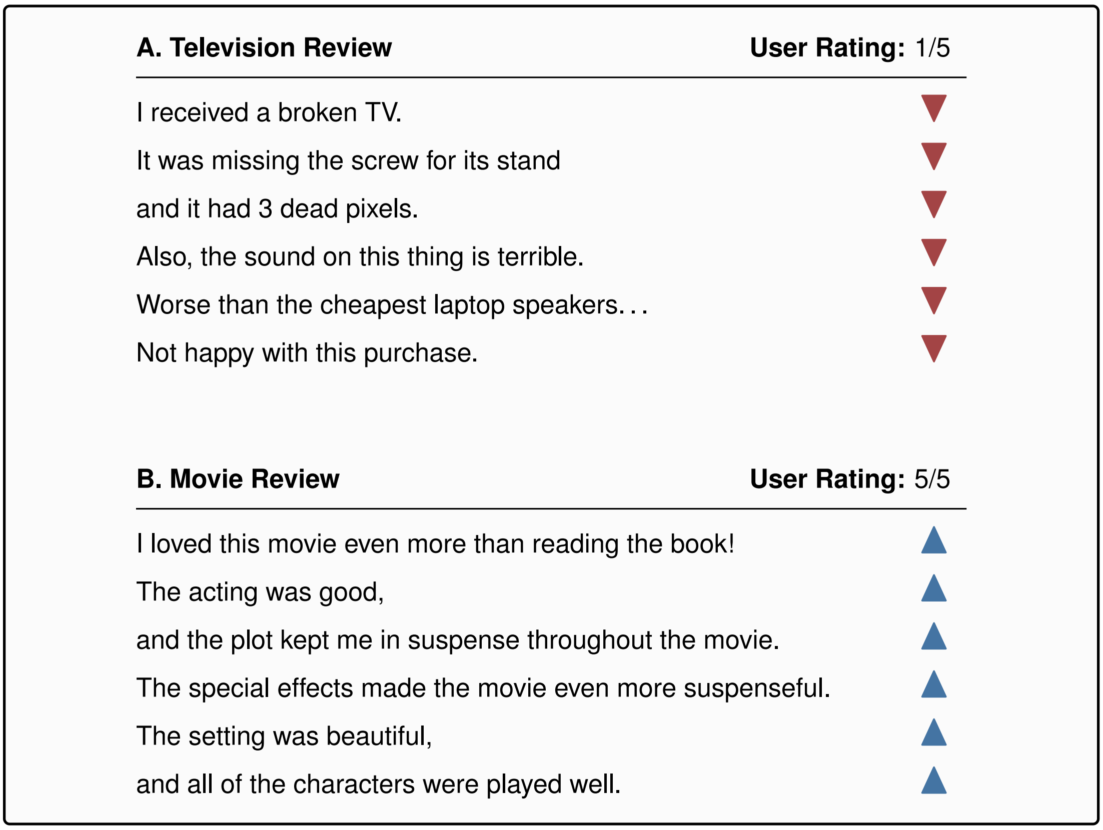
Figure 3.1: Categorization of reviews based on variation of sentiment in their constituent segments: strictly uniform (Type I), uniform with neutral segments (Type II) and mixed (Type III).

ment (Type I), reviews of uniform sentiment that include neutral segments (Type II), and reviews of mixed sentiment (Type III). We provide an overview in Figure 3.1, and a detailed description of each type that includes real examples in the forthcoming paragraphs.

Review Type I: Strictly Uniform Sentiment

This category refers to reviews where every segment conveys the same sentiment polarity. This means that a Type I review would contain all-positive, all-negative or all-neutral segments, as shown in the review sketches 1, 2 and 3 of Figure 3.1 respectively.

Our expectation is that such cases will be relatively rare, and mostly found in either extremely positive or extremely negative reviews. We provide real examples of Type I reviews from Amazon (review A) and IMDb (review B) in Figure 3.2. Review A is very negative (1 out of 5 stars) and all its segments express negative opinions. Analogously, review B (5 our of 5 stars) contains only positive segments. If one intended to infer the sentiment of every segment in a review given its sentiment as a whole, Type I reviews would present little challenge; the task boils down to propagating the review's sentiment to every constituent segment.



A. Television Review	**User Rating:** 1/5
I received a broken TV.	▼
It was missing the screw for its stand	▼
and it had 3 dead pixels.	▼
Also, the sound on this thing is terrible.	▼
Worse than the cheapest laptop speakers...	▼
Not happy with this purchase.	▼

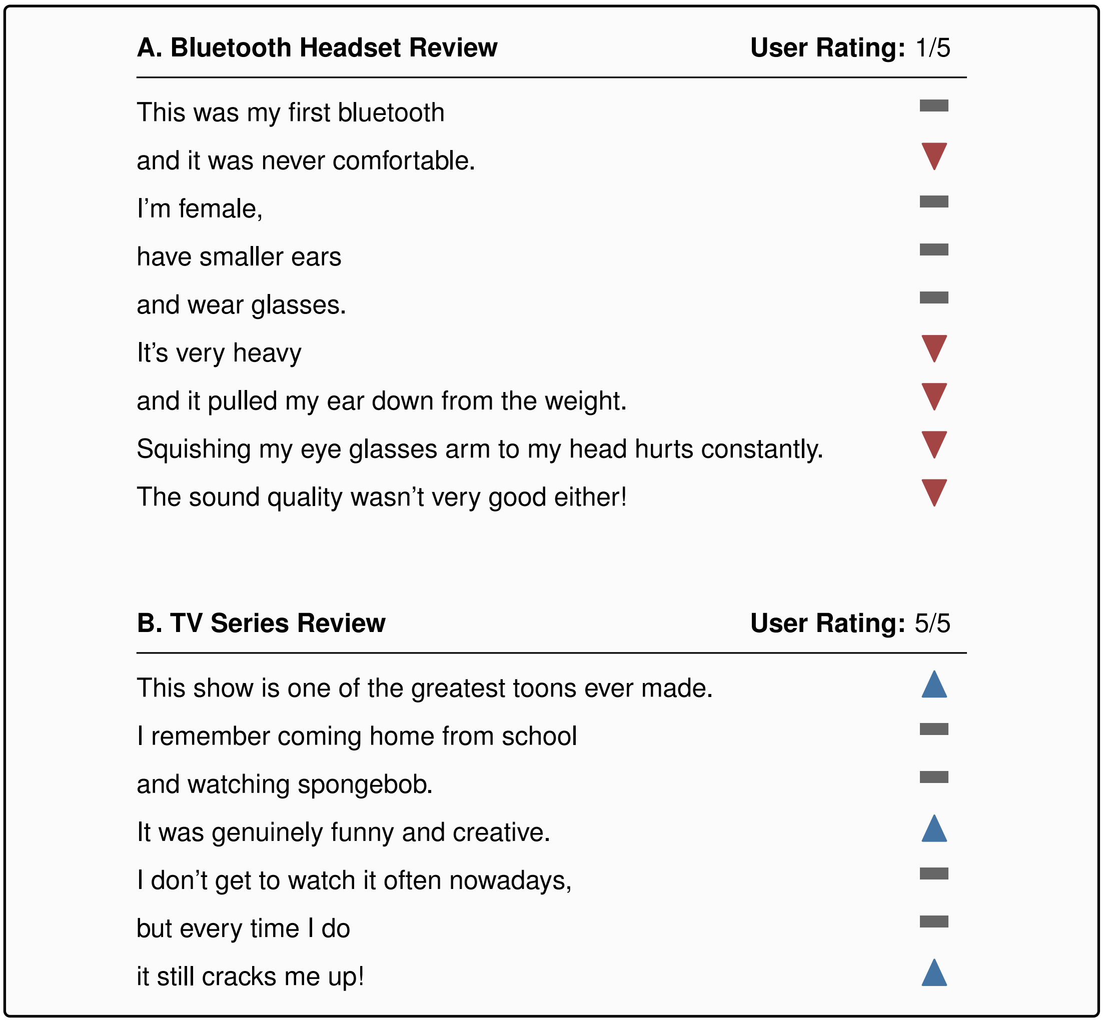
B. Movie Review	**User Rating:** 5/5
I loved this movie even more than reading the book!	▲
The acting was good,	▲
and the plot kept me in suspense throughout the movie.	▲
The special effects made the movie even more suspenseful.	▲
The setting was beautiful,	▲
and all of the characters were played well.	▲

Figure 3.2: Examples of reviews with strictly uniform sentiment (Type I). The television review is from Amazon and the movie review from IMDb. User ratings were provided by the reviewers. In Type I reviews, user ratings match perfectly the sentiment polarity of all individual segments (▼: Negative, ▲: Positive).

Review Type II: Uniform Sentiment with Neutral Segments

We define Type II reviews as those that contain positive- or negative-only opinions mixed with neutral statements, as shown in review sketches 4 and 5 of Figure 3.1.

Again, we expect these reviews to be concentrated on the positive and negative ends of the user rating spectrum. These should be more common than Type I instances, as reviewers tend to use objective statements in conjunction with their opinions. Figure 3.3 contains examples, where neutral segments like *"This was my first bluetooth"* and *"I remember coming home from school"* are combined with negative (Review A) or positive opinions (Review B). For Type II reviews, correctly predicting the sentiment of individual segments based on the review's overall polarity requires a mechanism to distinguish between segments that convey sentiment and those that do not.



A. Bluetooth Headset Review	**User Rating:** 1/5
This was my first bluetooth	▬
and it was never comfortable.	▼
I'm female,	▬
have smaller ears	▬
and wear glasses.	▬
It's very heavy	▼
and it pulled my ear down from the weight.	▼
Squishing my eye glasses arm to my head hurts constantly.	▼
The sound quality wasn't very good either!	▼

B. TV Series Review	**User Rating:** 5/5
This show is one of the greatest toons ever made.	▲
I remember coming home from school	▬
and watching spongebob.	▬
It was genuinely funny and creative.	▲
I don't get to watch it often nowadays,	▬
but every time I do	▬
it still cracks me up!	▲

Figure 3.3: Examples of reviews with uniform sentiment and neutral segments (Type II), for a bluetooth headset (Amazon) and a TV series (IMDb). User ratings were provided by the reviewers. In Type II reviews, there is a slight mismatch between the user rating and the sentiment of segments (▼: Negative, ▬: Neutral, ▲: Positive).

Review Type III: Mixed Sentiment

Finally, we group the remaining reviews, i.e., those that contain at least one positive and one negative segment, under Type III (see review sketches 6, 7 and 8 of Figure 3.1). Type III reviews, like the ones shown in Figure 3.4, are a mixture of positive, negative and neutral statements and are expected to be the majority. They also present a significant challenge for methods that try to predict sentiment on the segment-level. With opposing opinions expressed in a single review, the prediction of their polarity can no longer rely on the overall sentiment of the review, and therefore requires judging the polarity of each segment individually.

A. Bluetooth Headset Review	**User Rating:** 1/5
The appearance of this headset looks fine,	▲
but the microphone is basically crap.	▼
You will get enhanced NOISE rather than enhanced voice.	▼
Everybody complains	▼
that he/she cannot hear me.	▼

B. Vacuum Review	**User Rating:** 2/5
I just bought this	▬
and used it two times.	▬
Like others said,	▬
it’s very powerful,	▲
but the power works against it!	▼
The noise is deafening... Seriously.	▼
My ears are actually ringing right now.	▼

C. Restaurant Review	**User Rating:** 3/5
This place is in a scary neighborhood,	▼
but the inside looks nice and well kept up.	▲
The cashier I got tried to charge me more than he should!	▼
He said the deal was $1 off.	▬
I told him to ask his manager	▬
and she would correct him,	▬
which she did.	▬
Fries were still good though.	▲

Figure 3.4: Examples of reviews with mixed sentiment (Type III). The first two reviews are from Amazon and the third from Yelp. User ratings were provided by the reviewers. In Type III reviews, we expect the highest disagreement between user rating and segment sentiment (▼: Negative, ▬: Neutral, ▲: Positive).

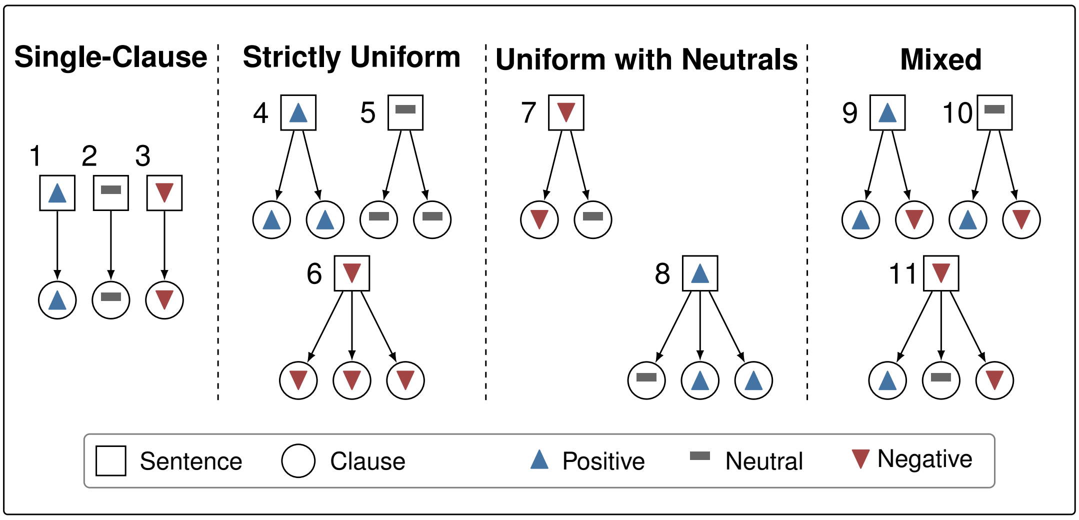


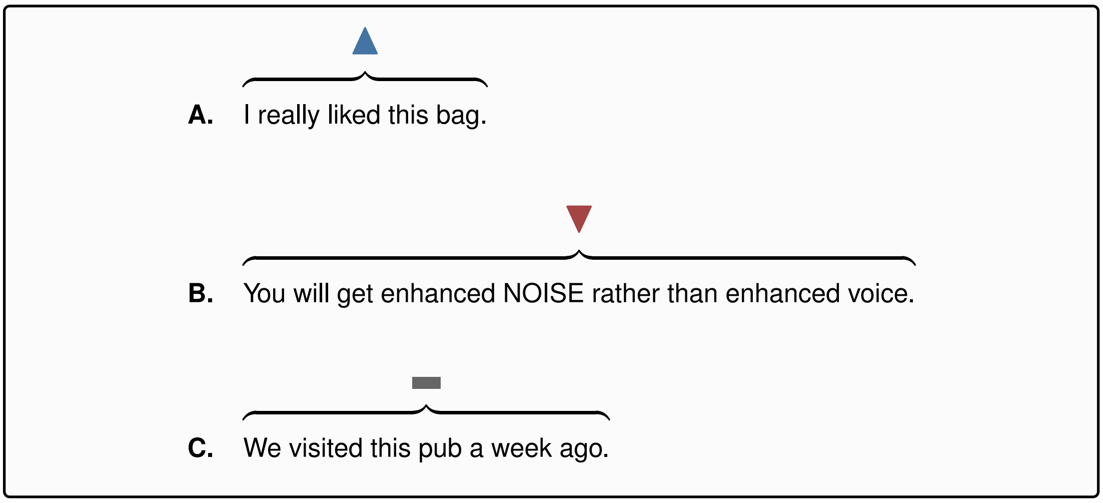
Figure 3.5: Categorization of review sentences based on variation of sentiment in their constituent clauses: single-clause (Type 0), strictly uniform (Type I), uniform with neutral clauses (Type II) and mixed (Type III).

3.1.2 Sentence Types

Having established the ways in which sentiment is expressed within reviews, we now turn our attention to the main building block of written text, the sentence. Similarly to reviews, sentences may contain one or more clauses.[1] The preferred level of granularity matters for tasks like sentiment prediction or opinion summarization and, in this thesis, we argue that using subsentential text units as the basis for sentiment analysis is advantageous. A single review sentence may target more than one aspects of the entity under review and express opinions of different sentiment for each of them. In such cases, detecting the polarity of a multi-clause sentence can be challenging for humans and automated methods alike.

Analogously to the previous section, we categorize the different means of composing sentiment within a sentence. We identify four sentence types which we describe in detail in the following paragraphs: single-clause sentences (Type 0), *strictly* uniform sentences (Type I), uniform sentences that include neutral clauses (Type II) and sentences with clauses of opposing polarities (Type III). Figure 3.5 provides an overview.

[1] We provide details on how we define and a obtain sub-sentence segmentation of sentences in Section 3.2.1.1. For now, the reader may assume the typical definition of a clause, i.e., the smallest grammatical unit that can express a complete proposition.



A. I really liked this bag. ▲

B. You will get enhanced NOISE rather than enhanced voice. ▼

C. We visited this pub a week ago. ▬

Figure 3.6: Examples of single-clause sentences (Type 0), from a bag, a bluetooth headset, and a restaurant review (▼: Negative, ▬: Neutral, ▲: Positive).

Sentence Type 0: Single-Clause

In the simplest case, a sentence is made up of a single clause and there is no sentiment composition to be considered, as illustrated by examples 1, 2 and 3 in Figure 3.5. Therefore, judging the sentiment of single-clause sentences, like the ones shown in Figure 3.6, boils down to analyzing a single proposition.

Sentence Type I: Strictly Uniform Sentiment

Sentences of Type I comprise more than one clauses with every clause conveying similar sentiment. The sentiment of the whole sentence is positive, negative or neutral, depending on the polarity of its constituents, as illustrated in sentence sketches 4, 5 and 6 of Figure 3.5.

Sentence examples are shown in Figure 3.7 and exemplify the lack of ambiguity in sentiment composition. Humans and automated systems should have little trouble judging the sentiment of either the whole sentences or their individual clauses. Still, in applications where the target of each opinion is important, considering sub-sentence segmentation can help identify aspect-specific expressions. For example, in sentence C, the first clause expresses an opinion about *acting*, whereas the second comments on the *plot*.

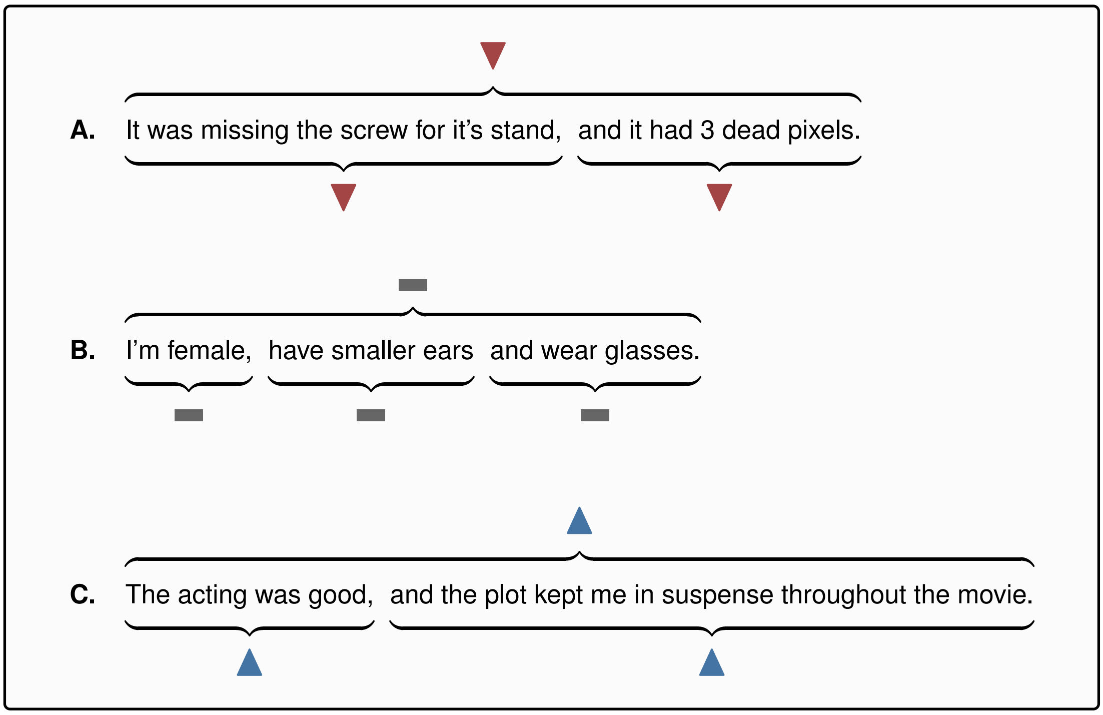

A.
It was missing the screw for it's stand,
and it had 3 dead pixels.
B.
I'm female,
have smaller ears
and wear glasses.
C.
The acting was good,
and the plot kept me in suspense throughout the movie.

Figure 3.7: Examples of sentences with strictly uniform sentiment (Type I), from a television, a bluetooth headset, and a movie review (▼: Negative, ▬: Neutral, ▲: Positive).

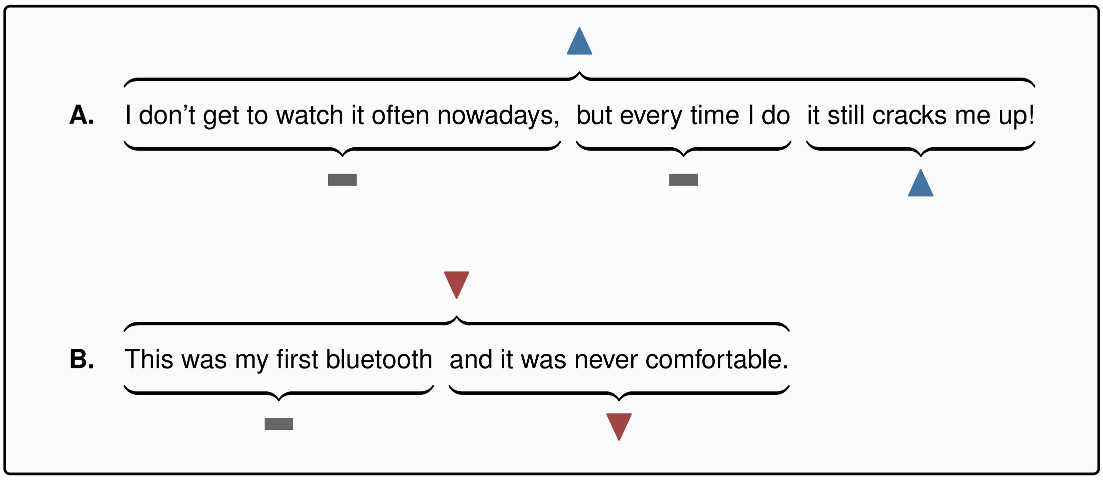

A.
I don't get to watch it often nowadays,
but every time I do
it still cracks me up!
B.
This was my first bluetooth
and it was never comfortable.

Figure 3.8: Examples of sentences with uniform sentiment and neutral clauses, from a TV series and a bluetooth headset review. (▼: Negative, ▬: Neutral, ▲: Positive)

Sentence Type II: Uniform Sentiment with Neutral Clauses

Type II sentences contain more than one clauses of non-negative or non-positive sentiment, i.e., combine clauses of similar polarity with objective statements. The sentiment of the whole sentence is determined by the sentiment of its non-neutral constituents, as illustrated by examples 7 and 8 in Figure 3.5.

Examples of positive (A) and negative (B) sentences with neutral propositions are provided in Figure 3.8. In the first sentence, neutral clauses *"I don't watch it often nowadays,"* and *"but every time I do"* provide context but do not alter the sentiment of the sentence, which is established by the positive clause *"it still cracks me up!"*. The same is true for the neutral *"This was my first bluetooth"*, which doesn't influence the negative sentiment set by *"and it was never comfortable"* in sentence B. Identifying concise opinions in sentences like these would require a method that can figure out the specific clauses that communicate sentiment.

Sentence Type III: Mixed Sentiment

Finally, Type III sentences are composed of more than one clauses, where at least one is positive and one is negative as illustrated by examples 9, 10 and 11 in Figure 3.5. In cases like these, the sentiment of the sentence as a whole depends on the relative importance of each opinion and is potentially ambiguous.

We provide a number of examples of varying ambiguity in Figure 3.9. In sentence A, the statement *"Great buy"* establishes the overall satisfaction of the reviewer, despite the cautionary elaboration *"just be careful with the straps"*. Sentence B first describes the positive attribute *"it's very powerful"*, amplified by the objective statement *"Like others said"* which indicates consensus among users. However, the sentence's overall sentiment is dominated by the argument that *"the power works against it!"*. While examples A and B may seem trivial for humans, an automated sentiment analysis system could still struggle to predict sentence-level sentiment.

There are cases, however, where the sentiment of sentences can be ambiguous even for human judges. Sentence C presents two diverging opinions (*"The appearance of this headset looks fine"* and *"but the microphone is basically crap"*) for different aspects of bluetooth headset (*looks* and *sound quality*, respectively). The overall sentiment of the sentence would be negative for most people, but it highly depends on whether someone values sound quality more than appearance. The same can be said about sentence D, where the extent to which the restaurant's *location* (*"This place is in a scary neighborhood"*) is affecting the reviewer's positive opinion about its *ambience* (*"but the inside looks nice and well kept up"*) is unclear.

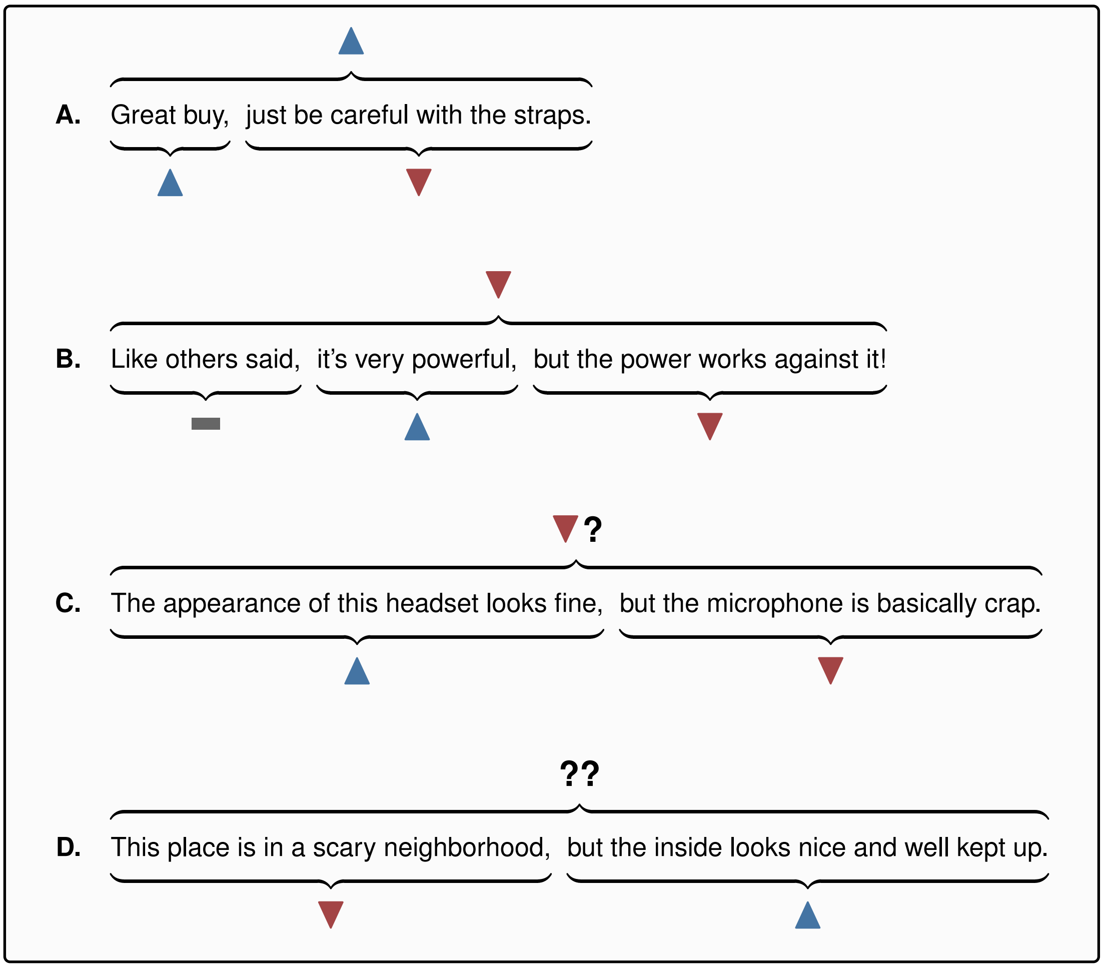

A. Great buy, just be careful with the straps.
B. Like others said, it's very powerful, but the power works against it!
C. The appearance of this headset looks fine, but the microphone is basically crap.
?
D. This place is in a scary neighborhood, but the inside looks nice and well kept up.
??

Figure 3.9: Examples of sentences with mixed sentiment (Type II), from a bag, a vacuum cleaner, a bluetooth headset, and a restaurant review (▼: Negative, ▬: Neutral, ▲: Positive).

So far, we described how expressions of varying sentiment may be distributed within reviews and sentences. We used review examples to show the different ways in which such expressions can be put together to articulate the opinions of reviewers, and discussed the degree to which humans and automated systems may struggle to deal with sentiment composition.

In the following section, we introduce OpoSum (shorthand for Opinion Summarization), the dataset used for training and evaluating our methods throughout this thesis. We give details about OpoSum's training set, as well as the first part of its human-annotated evaluation set, which is devoted to fine-grained sentiment. We also present empirical analysis on the distribution of sentiment in our data.

3.2 Review Data: The OpoSum Corpus

With neural networks playing an increasingly prominent role in the modeling of documents across domains (Cheng and Lapata, 2016; Liu and Lapata, 2018), large-scale review corpora have inevitably fuelled the advancements in state-of-the-art sentiment analysis research (Tang et al., 2015b; Yang et al., 2016). In contrast to other NLP tasks, where labeled data are particularly hard and expensive to obtain, reviews are almost always paired with a user rating that indicates the reviewer's overall satisfaction with his or her experience.

When the task at hand is document-level sentiment analysis, i.e., the prediction of the user's rating given the review text, this means that gold-standard labels are readily available for model training and testing. Unfortunately, this is not the case for machine learning efforts that aim to perform finer-grained analysis of sentiment, namely on the sentence, sub-sentence, or token level. In such cases, most previous work has opted to painstakingly label instances using human annotators, and design fully supervised learning methods (Socher et al., 2013; Kim, 2014).

Given the overarching goal of this thesis, which is to avoid human-annotated data for training, we heavily rely on those large-scale collection of reviews, which will provide a weak but necessary supervision signal for our models (see Chapters 4 and 5). However, as with any NLP problem, evaluation of our methods requires gold-standard data. With this in mind, we obtained readily available review data for training and annotated a smaller held-out set for testing. We use reviews from 3 sources, which cover 8 product or service domains in total and come with user ratings. To test the effectiveness of our segment-level sentiment predictors, we annotated a small but representative sample from each of these domains on two levels of granularity: sentences and clauses. The training and sentiment-annotated evaluation data are part of our opinion summarization corpus, OpoSum.

In the following sections we present details on the collection and annotation of these reviews.[2] In particular, Section 3.2.1 describes the large-scale training corpus we use throughout this thesis, including the preprocessing steps required to segment reviews into sentences and clauses. Then, Section 3.2.2 details the annotation procedure and characteristics of our segment polarity evaluation corpus. Finally, we look to explore the data and verify some of the claims of Section 3.1 in Section 3.2.3.

[2] Both training and testing data are available at: `https://github.com/stangelid/oposum`.

The OPOSUM Corpus								
TRAINING DATA								
	Local Business	Movies & TV	Laptop Bags	B/T Headsets	Boots	Keyb/s	TVs	Vac/s
Source	Yelp	IMDb	Amazon	Amazon	Amazon	Amazon	Amazon	Amazon
Products	–	–	2040	1471	4723	983	1894	1184
Reviews	335K	348K	43K	80K	78K	34K	57K	68K
Sent./Rev	8.9	14.0	5.9	7.5	5.5	7.5	10.7	9.0
EDUs/Rev	19.1	37.4	14.1	18.3	12.7	18.5	26.0	22.0
Words/Rev	128.3	279.2	98.1	122.5	82.6	127.0	180.4	146.6
Classes	5	10	5	5	5	5	5	5

Table 3.1: The OPOSUM reviews used for training our models, showing statistics about the number of reviews, average review sizes and number of classes.

3.2.1 Training Data

With reviews becoming an increasingly important factor in customer decisions, large collections of rated reviews can now be sourced from a plethora of web-based services: e-commerce portals like Amazon[3], crowd-sourced review websites like Yelp[4] and TripAdvisor[5], online databases like IMDb[6], and so on. In this work, we used data from three sources:

Yelp: Yelp is an online, crowd-sourced review forum which publishes user-generated reviews of local businesses in more than 20 countries. As of the end of 2017, 148 million reviews have been published on the platform. Since 2013, Yelp has been regularly releasing review collections for research purposes. In our work, we use the 2013 iteration of their dataset, as described by Tang et al. (2015a).

IMDb: IMDb, or the *Internet Movie Database*, is an online database of information about films and television programs, which also allows for user ratings and reviews. Diao et al. (2014) constructed a large dataset of reviews originating from IMDb, which we use here.

Amazon: An e-commerce giant, Amazon sells over 450 million products just in the US, along with millions of rated user reviews. He and McAuley (2016) collected

[3] https://www.amazon.com/

[4] https://www.yelp.com

[5] https://www.tripadvisor.com

[6] https://www.imdb.com/

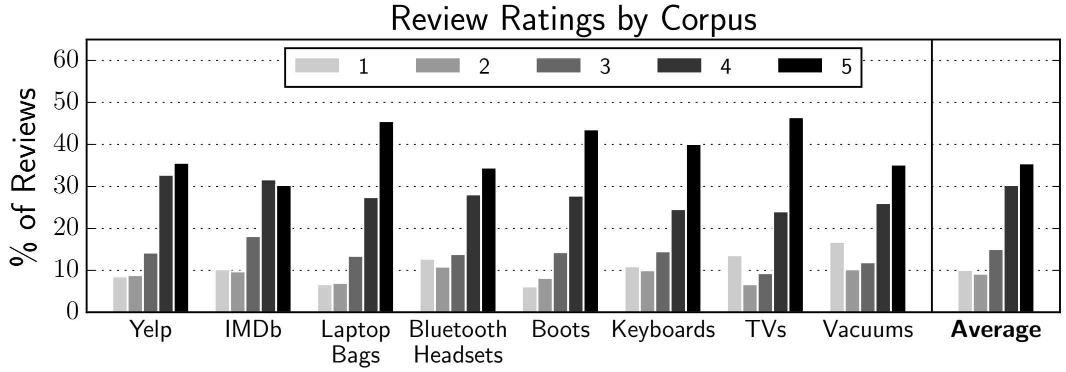


Figure 3.10: Distribution of user ratings in OpoSum's training set (we have merged IMDb's for presentation purposes).

and made publicly available 142 million Amazon reviews spanning an 18-year period (1996-2014). The reviews are organized in coarse-grained product categories, and accompanied by rich meta-data information including sub-category labels, product titles and reviewer identifiers. In order to make the data more manageable for training purposes, we subsampled reviews from 6 diverse, fine-grained product domains: *Laptop Bags*, *Bluetooth Headsets*, *Boots*, *Keyboards*, *Televisions* and *Vacuums*.

Table 3.1 shows the dataset sizes and other statistics for the collections used in this thesis. Figure 3.10 presents the distribution of user ratings for each domain and on average.[7] Throughout this thesis, and in accordance with literature, we use "Yelp" and "IMDb" to refer to the *Local Business* and *Movies & TV* domains, respectively. We use the domain name (*Laptop Bags*, *Bluetooth Headsets*, and so on) to refer to the Amazon collections.

3.2.1.1 Review Segmentation Policies

As mentioned earlier, one of the hypotheses investigated in this thesis regards the potential advantages of using sub-sentence clauses as the fundamental unit for analyzing opinions in reviews. While the vast majority of previous research on sentiment analysis and extractive summarization has viewed documents as sequences of sentences (Tang et al., 2015b; Yang et al., 2016; Cheng and Lapata, 2016; Nallapati et al., 2017), recent work has indicated that clauses, as instantiated by Rhetorical Structure Theory's (Mann and Thompson, 1988) *Elementary Discourse Units* (EDUs) can benefit both tasks (Bhatia et al., 2015; Li et al., 2016).

[7] For presentation purposes, this chapter's figures reduce IMDb's 10 classes to five, through merging.

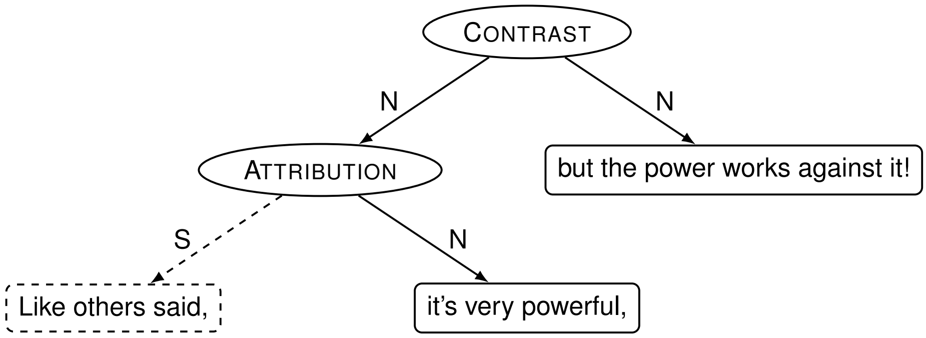


Figure 3.11: Rhetorical Structure Theory tree of a Vacuum review sentence.

Rhetorical Structure Theory (RST) represents documents as trees. The leaf nodes of RST trees are non-overlapping EDUs, which are a segmentation of sentences approximately corresponding to independent clauses. EDUs are hierarchically connected to form longer EDU spans. Specifically, an RST parse of a document will produce the following:

- **EDU segmentation:** Every sentence in the document is split into non-overlapping EDUs.
- **Nuclearity:** Adjacent EDUs may be connected in a Nucleus-Satellite relation, where the Nucleus EDU communicates a more salient piece of information to the reader than the Satellite. Multi-nucleus sibling relations are also possible. Connected siblings now form an EDU span which is similarly connected with other EDUs or EDU spans. The tree is not limited to a single sentence, as inter-sentence connections gradually build a discourse tree for the whole document.
- **Relations:** The type of association between connected EDUs or EDU spans is characterized by discourse relations, which include *Elaboration*, *Attribution*, *Contrast* and *Enablement* among others.

An example of a RST-parsed sentence taken from a Vacuum review is shown in Figure 3.11. The sentence *"Like others said, it's very powerful, but the power works against it!"* is split into three EDUs, and their nuclearity and relations is shown. The segment *"Like others said,"* is a satellite EDU that *attributes* the opinion *"it's very powerful"* (the Nucleus of the pair) to other reviewers. This EDU span as a whole is *contrasted* against the author's belief that *"[but] the power works against it!"*, forming multi-nucleus relation.

We constructed two versions for each review domain; one segmented into sentences, following previous work (Tang et al., 2015b; Yang et al., 2016), and one segmented into EDUs obtained from a state-of-the-art discourse parser (Feng and Hirst, 2012). In this thesis, we only utilize RST's segmentation and leave the potential use of the tree structure, as well as additional information pertaining to rhetorical relations and nuclearity, to future work. Hence, in the EDU-split versions of our corpora reviews are sequence of segments, albeit of finer granularity than sentences.

3.2.2 Segment Polarity Evaluation Data

Robust evaluation of any machine learning problem requires gold-standard evaluation data. For document-level sentiment analysis, where the task is to predict the overall polarity of a review, test data are freely available through the reviewers' ratings. However, this is not the case for our tasks of interest, like the prediction of sentiment at the segment level. Evaluating the performance of competing models will, therefore, require human annotations of sentiment for review sentences and EDUs.

Previous work on fine-grained sentiment analysis (Socher et al., 2013; Kim, 2014) mostly relied on the *rating scales* annotation method, where an annotator is presented with a piece of text and is asked to indicate its polarity on an ordinal scale between 1 and L, where 1 signifies highly negative and L highly positive sentiment. Averaging multiple annotations per item results in real-valued polarity scores, which may or may not be discretized to obtain distinct sentiment labels (e.g., positive, neutral, negative).

This approach is intuitive but presents a number of challenges, including *inconsistencies between annotators* (different people rate differently), *inconsistencies in individual annotators* (people rate differently over time), *scale region bias* (certain regions of the scale tend to be preferred). Additionally, there is an unavoidable trade-off between coarse-grained (too restrictive) and fine-grained (too overwhelming) scales (Kiritchenko and Mohammad, 2017).

One common alternative is the *paired comparison* method (David, 1963), where annotators see pairs of instances and select the one that is stronger in terms of the property of interest (e.g., "*which segment is more positive?*"). However, this requires order of N^2 annotations, where N is the number of instances to be annotated. *Ranking* methods, where more than two items are presented simultaneously and annotators provide their relative ranking, require fewer itemsets but are more cognitively demanding.

Best-Worst Scaling We opt for a third alternative, *Best-Worst Scaling* (BWS), also referred to as Maximum Difference Scaling (Louviere and Woodworth, 1991; Louviere

The OpoSum Corpus								
Segment Polarity Evaluation Data								
	Yelp	IMDb	Laptop Bags	B/T Headsets	Boots	Keyb/s	TVs	Vac/s
Reviews	100	100	25	25	25	25	25	25
Sentences	1065	1029	162	159	138	161	173	163
EDUs	2100	2398	365	317	301	344	334	357

Table 3.2: Statistics for our segment polarity evaluation corpus.

et al., 2015). A variant of comparative methods, BWS is an annotation scheme that has been gaining popularity in recent years and was used, among others, for measuring the sentiment polarity of words or phrases (Kiritchenko and Mohammad, 2016), emotion intensity in tweets (Mohammad and Bravo-Marquez, 2017) and relational similarity between word pairs (Jurgens et al., 2012).

In BWS, N instances are organized in sets of n items ($n > 1$ and typically $n = 4$), such that no item appears more than once in a single n-tuple and each item appears approximately in the same number of n-tuples in total. The combinations are not exhaustive, meaning that only a small random sample of n-tuples out of all possible ones are created. Multiple annotators are asked to select the *best* (highest in terms of property of interest) and *worst* (lowest in terms of property of interest) items. The final score for an instance is computed as the percentage of times it was selected as best minus the percentage of times it was selected as worst (Orme, 2009). The real-valued scores range from -1 to 1, and can be used to rank all items by the property of interest.

Recent studies have shown the advantages of BWS over other schemes (Kiritchenko and Mohammad, 2017), as only a very small number of tuples ($2N$ or $3N$) and annotations per tuple (between 2 and 4) are sufficient to obtain significantly more reliable results compared to a rating scale scheme with the same number of total judgements.

Constructing the Corpus To create our segment polarity evaluation corpus, we sampled reviews from every review domain such that all document-level classes are represented uniformly, and the review lengths are representative of the domains. The selected reviews were removed from the training data to make sure they are truly *unseen* during test time. As with the original collections, we segment reviews both on the sentence and EDU level. Review, sentence and EDU counts are shown in Table 3.2.

Television Corpus: EDU polarities

[+1.000]	Wonderful television picture, size and quality are all excellent.
[+0.982]	Nice and bright with good colors, I really like it.
⋮	⋮
[+0.444]	Picture quality is OK but not fantastic.
[+0.444]	The TV quality is solid for the price.
⋮	⋮
[+0.056]	and the TV box clearly says so.
[0.000]	According to some research I have done,
⋮	⋮
[-0.222]	However, this Samsung TV does not have integrated WI-FI.
[-0.278]	but its loudest volume has us straining to hear.
⋮	⋮
[-0.944]	Worse than the cheapest laptop speakers...
[-1.000]	This is the worst attempt of HDTV

Figure 3.12: Snippets from the ranked list of EDU polarities (shown in brackets) in our Television domain.

We used the Figure Eight platform[8] to annotate the data. Separate BWS annotations were performed for each domain and each segmentation, resulting in 16 distinct annotated corpora (eight domains × two segmentation policies). We used $1.5N$ 4-tuples per corpus, were N is the number of segments. Every tuple of segments was shown to three annotators, who were asked to select the most positive and most negative of the four. We provide the full instructions and interface for the annotation in Appendix A.1.

Using the previously mentioned BWS scoring method, we obtain a ranked list of segments spanning the whole range of sentiment polarities. A snippet from one of these lists is provided in Figure 3.12, where EDU polarities from the *televisions* domain are shown. Parts of the analysis in Section 3.2.3 and the evaluation presented in Chapter 3 rely on discrete sentiment labels for segments. Therefore, we also discretize the obtained scores into 3 classes: *positive* for segments with a polarity above $\frac{1}{3}$, *negative* for segments with a polarity below $-\frac{1}{3}$ and *neutral* for any other segment.

[8] `https://www.figure-eight.com/`

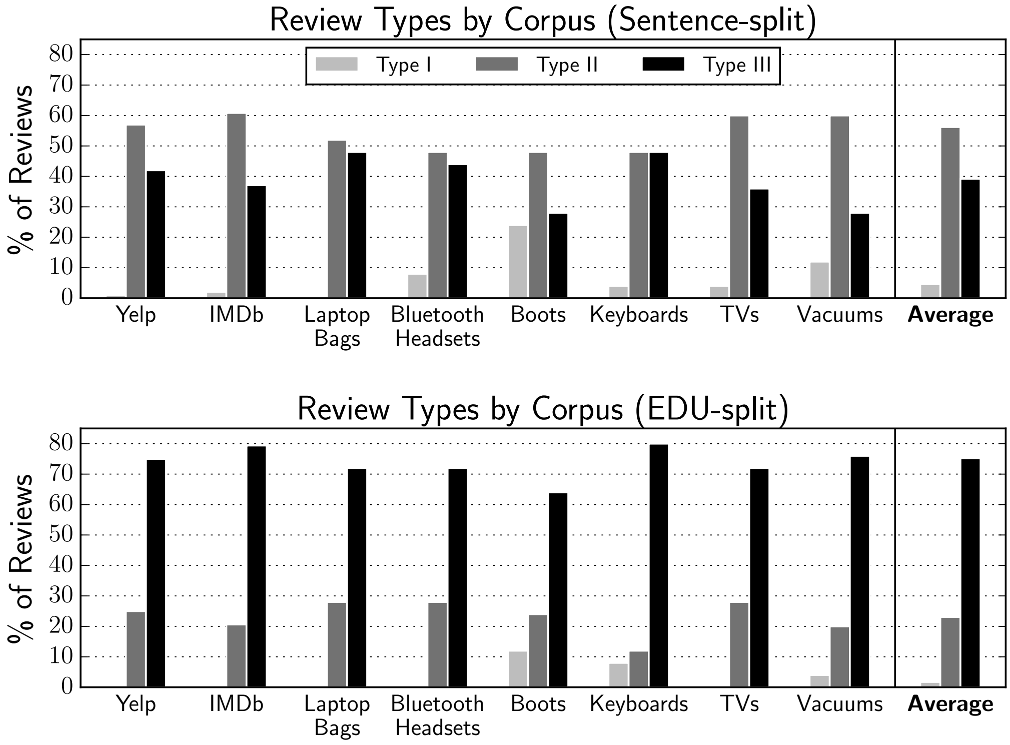


Figure 3.13: Distributions of Type I (strictly uniform), Type II (uniform with neutral segments), and Type III (mixed) reviews for every domain (and on average) on the sentence- and EDU-split versions of our annotated data.

3.2.3 Analysis

We finally move on to discuss some of the interesting findings that emerged from exploring OpoSum's segment polarity evaluation data. These include analysis of the distributions of review and sentence types across domains and document-level user ratings, and examination of fine-grained polarity scores and discrete labels.

Review Types Having obtained sentiment labels (positive, neutral and negative) for every segment in our evaluation corpus allows us to examine the number of strictly uniform (Type I), uniform with neutral segments (Type II), and mixed (Type III) reviews in the data. Additionally, we can compare how the granularity of segmentation relates to the distribution of sentiment within reviews, since we have annotated the sentiment of both sentences and EDUs.

Figure 3.13 shows the proportion of reviews that fall into each type for every domain separately, and on average. We observe that reviews with *strictly* uniform sentiment account for a very small fraction of data. In particular, only 10% of reviews

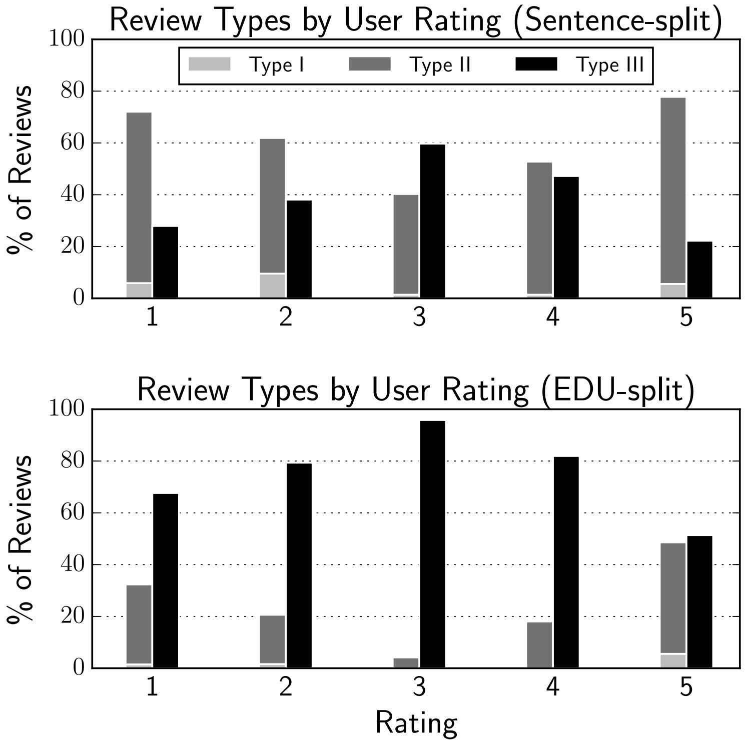


Figure 3.14: Distributions of Type I (strictly uniform), Type II (uniform with neutral segments), and Type III (mixed) reviews for every document-level class (i.e., user rating) on the sentence- and EDU-split versions of our annotated data.

contain sentences of identical sentiment, and an even smaller proportion of reviews contain strictly uniform EDUs. The majority of reviews fall under Type II (uniform sentiment with neutral segments) and Type III (mixed sentiment). In sentence-split reviews, Type II accounts for approximately 55% of cases, and Type III for 40%. The picture differs significantly when considering EDU segmentation, as more than 70% of reviews express at least one positive and one negative opinion (Type III).

We arrive at a number of conclusions from these observations. Firstly, reviews of perfectly uniform sentiment are very rare, as most reviews contain some combination of positive, negative and neutral statements. This confirms our hypothesis and reinforces our belief that successful methods for detecting fine-grained sentiment must have the ability to explicitly model segment subjectivity and polarity. Additionally, the abundance of Type III reviews in the EDU-split dataset indicates that further sentiment variation emerges when one investigates text units of finer granularity.

We are also interested in how review types are distributed among reviews of a particular user rating. Our hypothesis is that uniform and strictly uniform reviews would

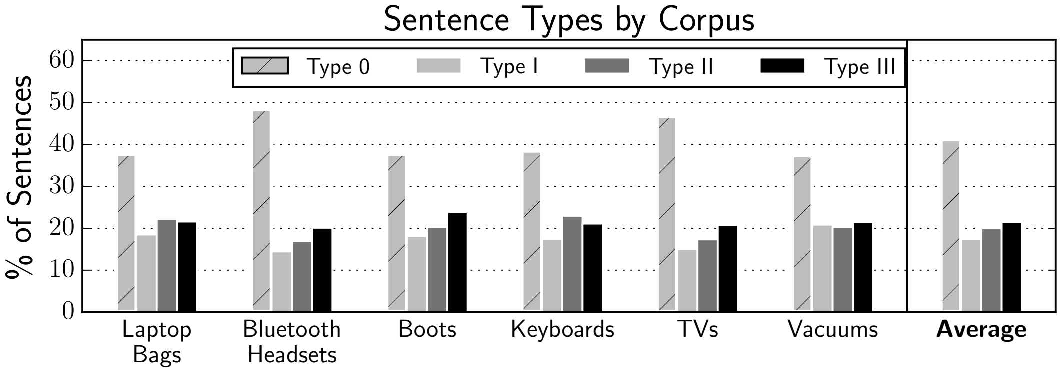


Figure 3.15: Distributions of Type 0 (single-clause), Type I (strictly uniform), Type II (uniform with neutral clauses) and Type III (mixed) sentences for every domain (and on average) on out annotated data.

be more frequent for very positive or very negative reviews, whereas ambivalent reviews will tend to contain segments of mixed polarity. Figure 3.14 shows how review types are distributed within each document-level rating and confirms our expectation. Type I and II reviews are mostly concentrated at both ends of the rating scale, whereas the Type III reviews have a single mode that appears in instances accompanied by a 3 out of 5 rating. The trend is similar for both segmentation policies, but is much more pronounced for EDU-split reviews.

Sentence Types We performed a similar analysis on the distribution of sentence types in the human-annotated data, by investigating the discrete sentiment labels in sentences and their constituent EDUs. In particular, we computed the proportion of sentences containing only a single EDU (Type 0) and those with multiple constituent EDUs of strictly uniform (Type I), uniform (Type II) and mixed (Type III) sentiment.

Figure 3.15 presents the distributions of sentence types for all Amazon domains, as well as their average. We observe that approximately 40% of sentences contain only a single EDU, and that multi-clause sentences are almost uniformly distributed across Types I, II and III. 20% of sentences express neutral statements along with opinions of uniform sentiment (Type II) and a further 20% contain EDUs of opposing polarity (Type III), which further underscores the potential advantage of EDU segmentation.

Distribution of Polarities Scores An illustration of the polarity scores obtained by BWS is shown in Figure 3.16. For each of the domains (and for their combination; top

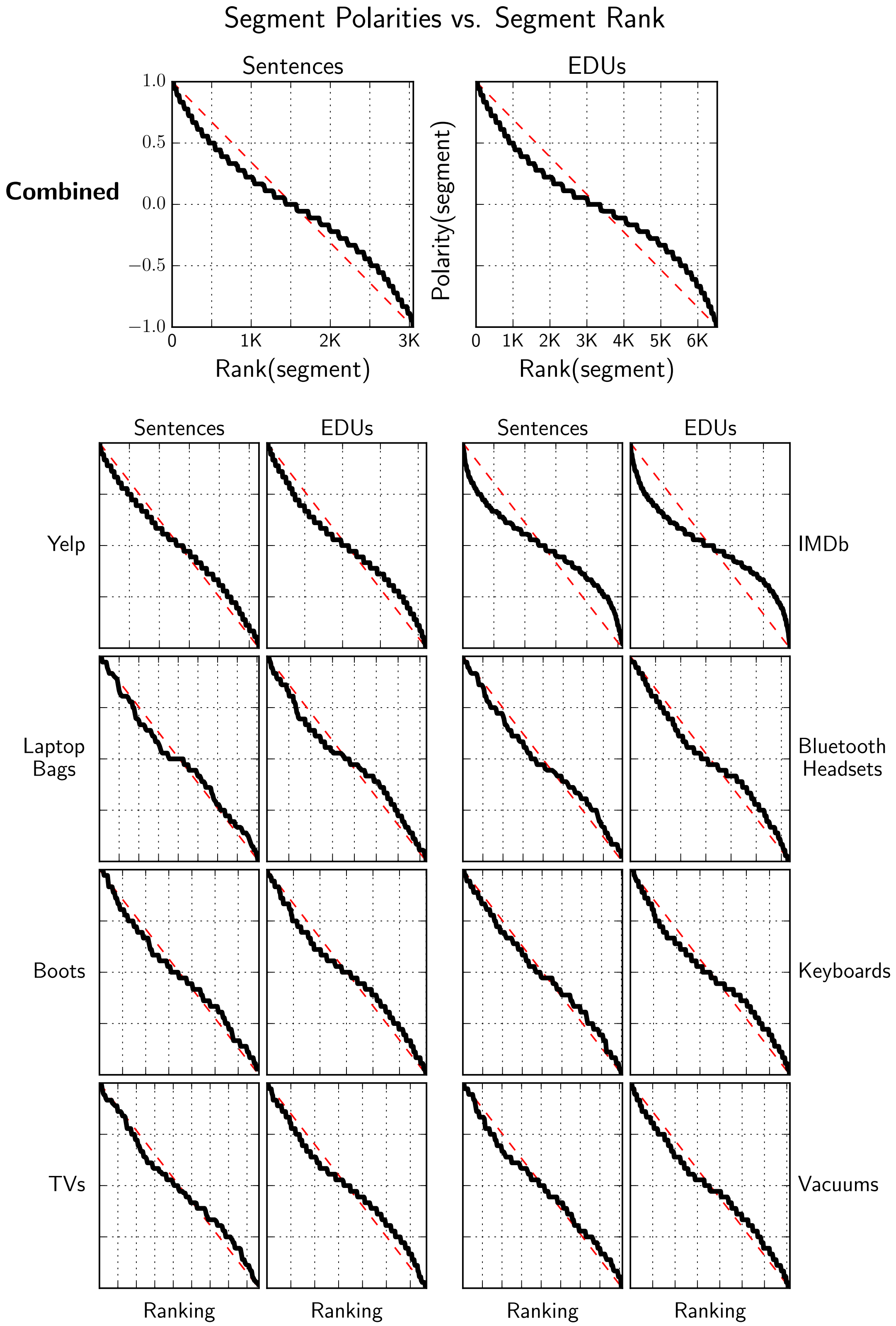


Figure 3.16: Segment polarity scores obtained from Best-Worst Scaling annotation plotted against the segment's rank in the polarity scale. We provide plots for all segments in our human-annotated data (larger figures on top of page) and for each domain individually.

of the figure), polarity scores (y-axis) for sentences and EDUs are plotted against their rank (x-axis) in the polarity scale. We observe that the ranked lists for each corpus have no significant gaps, i.e., they contain entries across the $[-1,1]$ range. The red dashed lines represent how a perfectly uniform distribution of scores would look like. The s-shaped form of the lines indicates that the positive and negative ends of the scale are populated with slightly less segments than its center. The trend is most pronounced in IMDb reviews, indicating a larger proportion of neutral segments. Amazon reviews, on the other hand, have very uniform polarity distributions.

Segment Sentiment vs. Review Sentiment Finally, we investigate how the numbers of positive, negative and neutral segments vary based on the overall sentiment of reviews. Figure 3.17 illustrates the relative proportion of segment labels for each document-level class (i.e., user rating). Again, we provide plots for sentence- and EDU-split reviews, averaged across domains (top of figure), and for each domain. A large fraction of segments are neutral across all classes. For sentences, we find more neutral segments in the middle of the user review scale, whereas the number of neutrals is more consistent for EDUs. As expected, negative reviews are dominated by negative segments and positive reviews contain many more positive ones.

3.3 Summary

This chapter introduced an in-depth overview and analysis of review data. In particular, we first presented an abstract categorization of the various ways that sentiment is expressed and combined within reviews and their constituent sentences. We hypothesized that reviews of non-uniform sentiment will be more frequent and that a considerable fraction of sentences will also combine opinions of different polarity. We argued that this should pose certain constraints on the type of methods used for fine-grained sentiment analysis.

We then presented the first two parts of OpoSum: a large-scale review corpus to be used for the training of learning methods, comprising multiple collections of reviews from diverse domains; and a held-out segment polarity evaluation dataset, which contains fine-grained sentiment labels obtained via best-worst scaling.

Finally, we used the human-annotated polarity data as a testbed to verify our expectations regarding the distribution of sentiment in reviews. Our analysis showed that (a) the majority of reviews contain at least 2 opinions of opposing polarity, (b) reviews

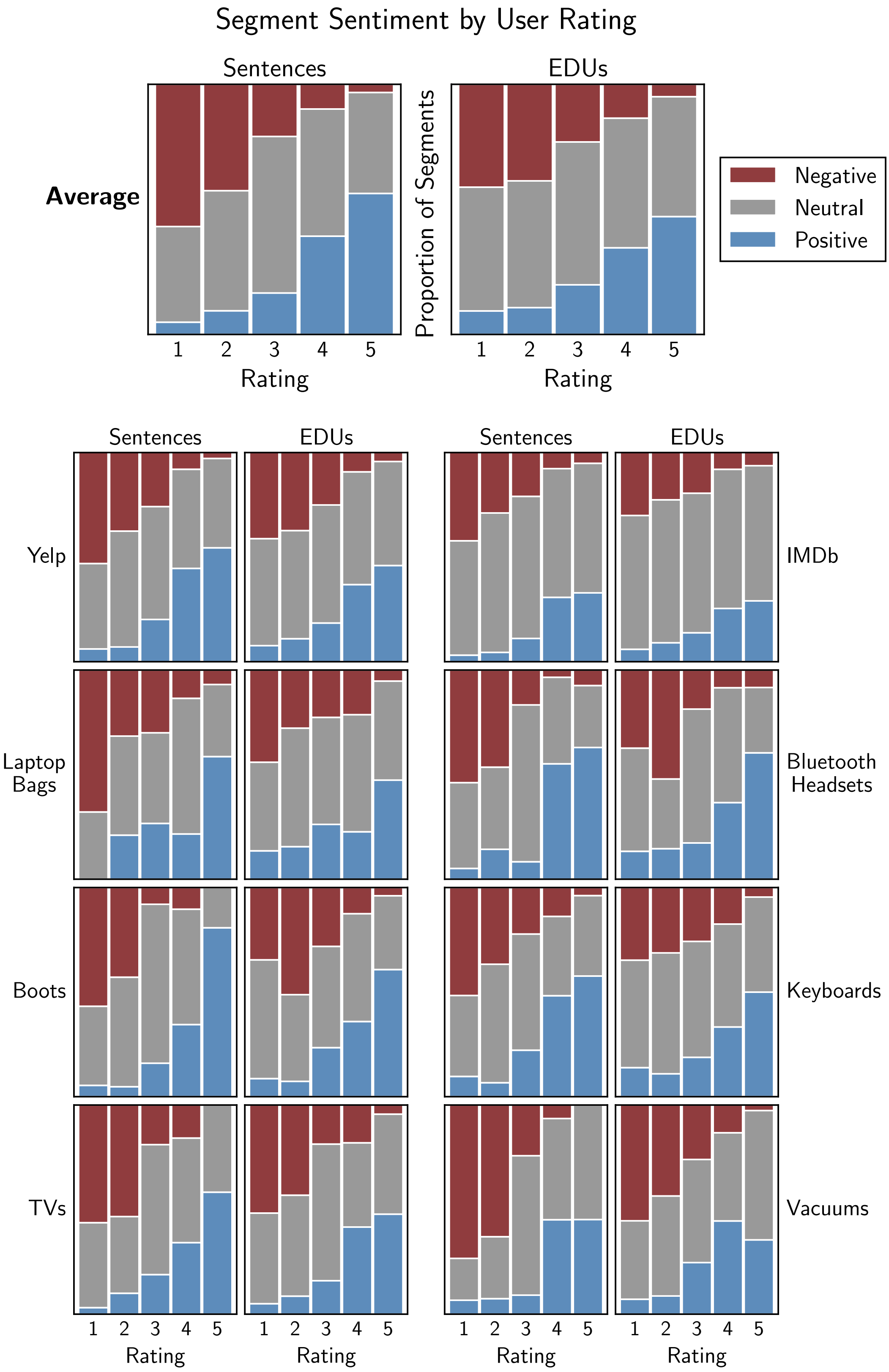


Figure 3.17: The proportion of positive, neutral and negative segments for each document-level sentiment class in each domain of OpoSum and on average.

of uniform sentiment are rare and found almost exclusively in extremely positive or negative reviews, (c) a third of the sentences that are made of more than one clauses will contain opinions of opposing sentiment, and (d) neutral statements appear in reviews consistently regardless of the overall user rating. It follows that models aiming to detect fine-grained sentiment must have the flexibility to detect segments of opposing polarity, examine reviews at the clause level, and successfully deal with neutral statements.

In the next chapter, we present two neural networks that tackle the task: a hierarchical model, HIERNET, which uses attention weights to identify sentiment-heavy opinions, and MILNET, which is based on Multiple Instance Learning (Keeler and Rumelhart, 1992; Dietterich et al., 1997), and can detect sentiment across the polarity spectrum, regardless of the review's overall rating.

Chapter 4

Detecting Fine-Grained Sentiment in Reviews

In the simplest case, identifying opinions boils down to detecting fine-grained sentiment within reviews. While other factors are clearly in play, like the target of a statement (i.e., aspect of reviewed item; see Chapter 5), sentiment is the clearest indicator of subjectivity in text (Pang and Lee, 2008).

In this chapter, we present our weakly supervised approach for the detection of segment-level polarity in reviews. Our goal is to design neural models that learn to detect sentiment-heavy statements, without recourse to fine-grained human annotations. Instead, we only use supervision on the document level, which we obtain from freely available user ratings that often accompany review data.

To that end, we propose two hierarchical neural networks which take different approaches towards composing sentiment information. One follows the standard route of hierarchically combining the *representations* of constituent segments into a single vector, used to predict the review's overall sentiment. The review-level prediction is then propagated down to individual segments, and fine-tuned using attention to uncover neutral statements. The second is based on Multiple Instance Learning (Keeler and Rumelhart, 1992; Dietterich et al., 1997), a special case of weakly supervised learning, and makes *individual* predictions on the segment level which are then combined into a review-level prediction. Again, we use an attention-based mechanism to further differentiate between segments of varying importance.

We evaluate these methods in three ways. Firstly, we formalize the task of sentiment detection as a segment-level classification problem, similarly to most previous work. Secondly, we view the task from a ranking perspective, which we argue is closer to a practical opinion detection system. Finally, we use the outputs of our model to perform single-review opinion extraction and use human judges to measure the quality of the obtained extractive summaries.

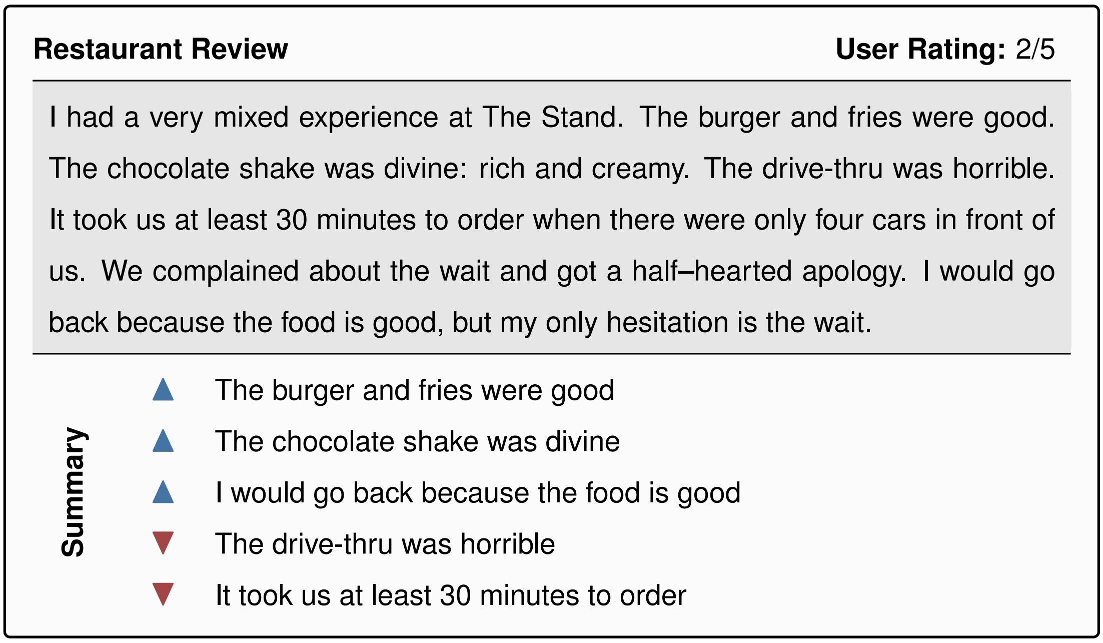
Our experimental results suggest that: (a) our multiple instance learning model produces more accurate sentiment distinctions for review segments than a traditional model based on hierarchical vector composition; (b) it performs comparably to a state-of-the-art lexicon-based sentiment detection model and a fully supervised neural network; and (c) EDU-based extractive summaries are more informative than those based on sentences.

4.1 Introduction

Sentiment analysis has become a fundamental area of research in Natural Language Processing thanks to the proliferation of user-generated content in the form of online reviews, blogs, internet forums, and social media. A plethora of methods have been proposed in the literature that attempt to distill sentiment information from text, allowing users and service providers to make opinion-driven decisions.

The success of neural networks in a variety of applications (Bahdanau et al., 2015; Le and Mikolov, 2014; Socher et al., 2013) and the availability of large amounts of labeled review data have led to an increased focus on sentiment classification. Supervised models are typically trained on documents (Johnson and Zhang, 2015a,b; Tang et al., 2015a; Yang et al., 2016), sentences (Kim, 2014), or phrases (Socher et al., 2011, 2013) annotated with sentiment labels and used to predict sentiment in unseen texts. Coarse-grained document-level annotations are relatively easy to obtain due to the widespread use of opinion grading interfaces (i.e., star ratings accompanying reviews). In contrast, the acquisition of sentence- or phrase-level sentiment labels remains a laborious and expensive endeavor despite its relevance to various opinion mining applications, e.g., detecting or summarizing consumer opinions in online product reviews.

The usefulness of finer-grained sentiment analysis is illustrated in Figure 4.1, where snippets of opposing polarities are extracted from a 2-star restaurant review. Although, as a whole, the review conveys negative sentiment, aspects of the reviewer's experience were clearly positive, as indicated by statements like *"The burger and fries were good"* and *"The chocolate shake was divine"*. This goes largely unnoticed when focusing solely on the review's overall rating.



Restaurant Review **User Rating:** 2/5

I had a very mixed experience at The Stand. The burger and fries were good. The chocolate shake was divine: rich and creamy. The drive-thru was horrible. It took us at least 30 minutes to order when there were only four cars in front of us. We complained about the wait and got a half–hearted apology. I would go back because the food is good, but my only hesitation is the wait.

Summary

▲ The burger and fries were good

▲ The chocolate shake was divine

▲ I would go back because the food is good

▼ The drive-thru was horrible

▼ It took us at least 30 minutes to order

Figure 4.1: An EDU-based summary of a 2-out-of-5 stars review with positive and negative snippets.

In this chapter, we consider the problem of segment-level sentiment analysis as a weakly supervised task. Instead of judging the sentiment of segments using fine-grained supervision (i.e., via expensive human annotations), we present two methods that only require document-level labels and learn to introspectively judge the sentiment of constituent segments, within the context of whole reviews.

Beyond showing how to utilize document collections of rated reviews to train fine-grained sentiment predictors, we also investigate the granularity of the extracted segments. Previous research (Tang et al., 2015a; Yang et al., 2016; Cheng and Lapata, 2016; Nallapati et al., 2017) has predominantly viewed documents as sequences of sentences. Inspired by recent work in summarization (Li et al., 2016) and sentiment classification (Bhatia et al., 2015), and driven by our analysis presented in Chapter 3, we also represent documents as sequences of *Elementary Discourse Units* (EDUs) that approximate sub-sentence clauses, under the framework of Rhetorical Structure Theory (Mann and Thompson, 1988).

Our contributions in this chapter are three-fold: we propose two hierarchical neural networks, including a novel architecture based on *Multiple Instance Learning* (MIL; Keeler and Rumelhart 1992), which utilize document-level sentiment supervision to judge the polarity of constituent segments; we apply these methods on our newly created OpoSum corpus, a publicly available dataset which includes an evaluation set of segment-level polarity annotations for sentences and EDUs (see Chapter 3); and we present empirical findings (through automatic and human-based evaluation) that neural multiple instance learning is superior to more conventional neural architectures and a lexicon-based method, and on par with a fully supervised CNN model, like the one described in Section 2.2.

4.2 Related Work

Our work lies at the intersection of multiple research areas, including sentiment classification, opinion mining and multiple instance learning. We review related work in these areas below.

4.2.1 Sentiment Classification

Sentiment classification is one of the most popular tasks in opinion analysis. Early work focused on unsupervised methods and the creation of sentiment lexicons (Turney, 2002; Hu and Liu, 2004; Wiebe et al., 2005; Baccianella et al., 2010) based on which the overall polarity of a text can be computed (e,g., by aggregating the sentiment scores of constituent words). More recently, Taboada et al. (2011) introduced SO-CAL, a state-of-the-art method that combines a rich sentiment lexicon with carefully defined rules over syntax trees to predict sentence sentiment.

Supervised learning techniques have dominated the literature (Pang et al., 2002; Pang and Lee, 2005; Qu et al., 2010; Xia and Zong, 2010; Wang and Manning, 2012; Le and Mikolov, 2014) thanks to user-generated sentiment labels or large-scale crowdsourcing efforts (Socher et al., 2013). Neural network models in particular have achieved state-of-the-art performance on various sentiment classification tasks due to their ability to alleviate feature engineering. Kim (2014) introduced a very successful Convolutional Neural Network (CNN) architecture for sentence-level classification, whereas other work (Socher et al., 2011, 2013) uses recursive neural networks to learn sentiment for segments of varying granularity (i.e., words, phrases, and sentences). We describe Kim's (2014) CNN encoder in detail in Section 4.4, as it is a component of our models.

The availability of large-scale datasets (Diao et al., 2014; Tang et al., 2015a) has also led to the development of document-level sentiment classifiers which exploit hierarchical neural representations. These are obtained by first building representations of sentences and aggregating those into a document feature vector (Tang et al., 2015a). Yang et al. (2016) further acknowledge that words and sentences are deferentially important in different contexts. They present a model which learns to attend (Bahdanau et al., 2015) to individual text parts when constructing document representations. We describe a similar architecture in Section 4.4 and detail how it can be used to produce segment-level sentiment distinctions.

Our work draws inspiration from representation learning, especially the idea that not all parts of a document convey sentiment-worthy clues (Yang et al., 2016). While previous work on segment-level sentiment classification has proposed fully supervised methods that view segments in isolation (Socher et al., 2013; Kim, 2014), our methods provide a natural way of predicting the polarity of individual text segments *within context* (i.e the reviews they appear in), without requiring segment-level annotations. Moreover, our attention mechanism directly facilitates opinion detection rather than simply aggregating sentence representations into a single document vector.

4.2.2 Opinion Mining

A standard setting for opinion mining and summarization (Lerman et al., 2009; Carenini et al., 2006; Ganesan et al., 2010; Di Fabbrizio et al., 2014; Gerani et al., 2014) assumes a set of documents that contain opinions about some entity of interest (e.g., camera). The goal of the system is to generate a summary that is representative of the average opinion and speaks to its important aspects (e.g., picture quality, battery life, value). Output summaries can be extractive (Lerman et al., 2009) or abstractive (Ganesan et al., 2010; Gerani et al., 2014; Di Fabbrizio et al., 2014) and the underlying systems exhibit varying degrees of linguistic sophistication from identifying aspects (Lerman et al., 2009) to using RST-style discourse analysis, and manually defined templates (Gerani et al., 2014; Di Fabbrizio et al., 2014).

Our proposed method departs from previous work in that it focuses on detecting opinions in individual documents. Given a review, we predict the polarity of every segment, allowing for the extraction of sentiment-heavy opinions. We explore the usefulness of EDU segmentation inspired by Li et al. (2016), who show that EDU-based summaries align with near-extractive summaries constructed by news editors. Importantly, our model is trained in a weakly supervised fashion on large scale review collections, without recourse to fine-grained labels or gold-standard opinion summaries.

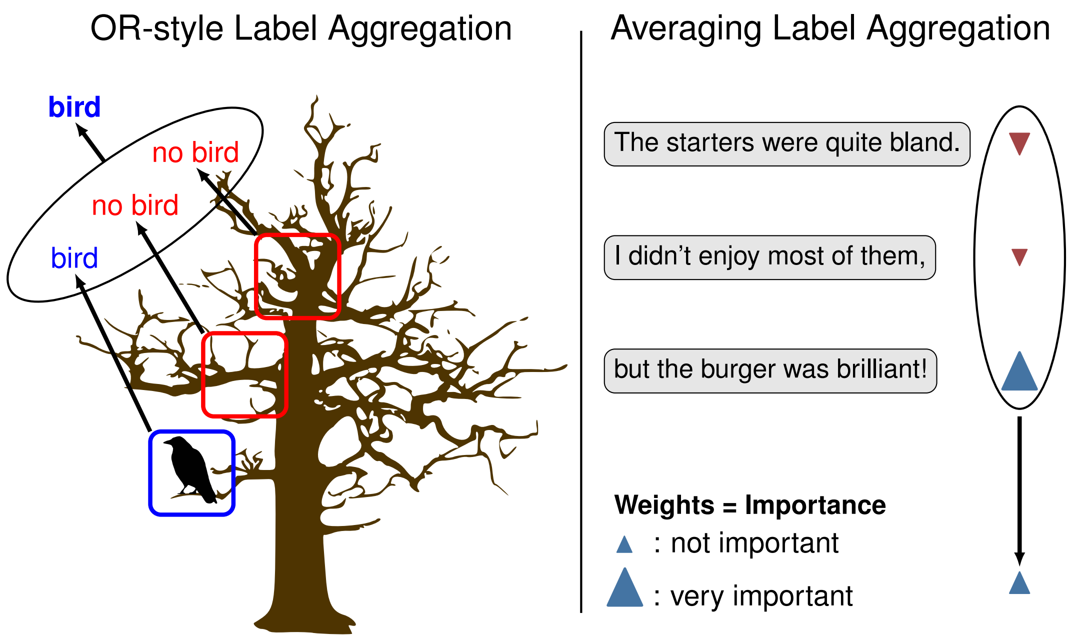

OR-style Label Aggregation
bird
no bird
no bird
bird
Averaging Label Aggregation
The starters were quite bland.
I didn't enjoy most of them,
but the burger was brilliant!
Weights = Importance
▲ : not important
▲ : very important

Figure 4.2: Two tasks framed under Multiple Instance Learning: (left) Object recognition, where a bird classifier is applied on multiple image patches and a single positive patch deems the whole image as positive; and (b) sentiment analysis, where the sentiment predictions of a series of segments is combined using a weighted average.

4.2.3 Multiple Instance Learning

Our novel model, presented in Section 4.5, adopts a *Multiple Instance Learning* (MIL) framework. MIL is a special case of weakly supervised learning where labels are associated with groups of instances or *bags* (reviews in our case), while instance labels (segment-level sentiment) are unobserved. An aggregation function is used to combine instance predictions and assign labels on the bag level. The goal is either to label bags (Keeler and Rumelhart, 1992; Dietterich et al., 1997; Maron and Ratan, 1998) or to simultaneously infer bag and instance labels (Zhou et al., 2009; Wei et al., 2014; Kotzias et al., 2015). We view segment-level sentiment analysis as an instantiation of the latter variant.

Applications of MIL are many and varied. MIL was first explored by Keeler and Rumelhart (1992) for recognizing handwritten post codes, where the position and value of individual digits was unknown. MIL techniques have since been applied to drug activity prediction (Dietterich et al., 1997), image retrieval (Maron and Ratan, 1998; Zhang et al., 2002), object detection (Zhang et al., 2006; Carbonetto et al., 2008; Cour et al., 2011), text classification (Andrews and Hofmann, 2004), image captioning (Wu et al., 2015), paraphrase detection (Xu et al., 2014), and information extraction (Hoffmann et al., 2011).

Initial MIL efforts for binary classification made the strong assumption that a bag is negative only if all of its instances are negative, and positive otherwise (Dietterich et al., 1997; Maron and Ratan, 1998; Zhang et al., 2002; Andrews and Hofmann, 2004; Carbonetto et al., 2008). A common application of an OR-style label aggregation function like this is object recognition, where an object in question is predicted to appear in an image (bag), if at least a single image patch (instance) contains it. Figure 4.2 (left) illustrates such an example, where an image patch including a bird is sufficient to recognize that the image contains a bird.

Subsequent work relaxed this assumption, allowing for prediction combinations better suited to the tasks at hand. Weidmann et al. (2003) introduced a generalized MIL framework, where a combination of instance types is required to assign a bag label. Zhou et al. (2009) used graph kernels to aggregate predictions, exploiting relations between instances in object and text categorization. Xu and Frank (2004) proposed a multiple-instance logistic regression classifier, where instance predictions were averaged, assuming equal and independent contribution toward bag classification. More recently, Kotzias et al. (2015) used sentence vectors obtained by a pre-trained hierarchical CNN (Denil et al., 2014) as features under an unweighted average MIL objective. Prediction averaging was further explored by Pappas and Popescu-Belis (2014; 2017), who used a weighted summation of predictions, an idea we adopt as well.

When applied to sentiment analysis, MIL takes advantage of supervision signals on the document level in order to train segment-level sentiment predictors. Label aggregation by averaging is better suited for this task, based on the assumption that a review's overall sentiment is the (weighted) average of its constituent opinions' polarity. The idea is illustrated in Figure 4.2 (right), where three segments of varying sentiment and importance are combined to communicate the author's attitude.

Although their work is not couched in the framework of MIL, Täckström and McDonald (2011) show how sentence sentiment labels can be learned as latent variables from document-level annotations using hidden conditional random fields. Pappas and Popescu-Belis (2014) use a multiple instance regression model to assign sentiment scores to specific aspects of products. The Group-Instance Cost Function (GICF), proposed by Kotzias et al. (2015), averages sentence sentiment predictions during training, while ensuring that similar sentences receive similar polarity labels. Their model is not trainable end-to-end, in contrast with our proposed network. Additionally, none of the aforementioned efforts explicitly evaluate opinion extraction quality.

4.3 Preliminaries

In this section, we provide formal definitions of the concepts used throughout the chapter, as well as an overview of the three complementary tasks that will form the basis of the evaluation of our system: *segment-level sentiment classification*, *polarity ranking*, and *single-review opinion extraction*.

4.3.1 Definitions

Let $\mathcal{C}$ denote a corpus of reviews from a domain $d_{\mathcal{C}}$, e.g., televisions or restaurants. The dataset contains a set of reviews $\{r_i\}_{i=1}^{|\mathcal{C}|}$ expressing customers' opinions. Each review r is accompanied by the author's overall rating $y_r \in [1, L]$, where the labelset is ordered and classes 1 and L correspond to maximally negative and maximally positive sentiment. Each review is split into segments $(s_1, \ldots, s_m)$, and a segment s_j is in turn viewed as a sequence of words $(w_{j1}, \ldots, w_{jn})$. In the context of this chapter, a segment may be a sentence or an EDU. Additionally, each segment s is associated with an *unobserved* polarity.

A segment's polarity, pol_s, is either expressed using discrete classes in $[1, L]$, similarly to review labels, or takes real values in $[-1, +1]$, where -1 indicates maximally negative and $+1$ maximally positive sentiment. Both discrete and real-valued definitions are relevant to this work. Our neural models will produce a probability distribution over the discrete classes, which we then transform into real-valued scores to facilitate polarity-based segment ranking and opinion extraction, as described in Section 4.6.

Probabilistic sentiment classifiers will produce document-level predictions $\hat{y}_r$ by selecting the most probable class according to class distribution $\mathbf{p}_r = \langle p_r^{(1)}, \ldots, p_r^{(L)} \rangle$. In a non-MIL framework a classifier would learn to predict the review's sentiment by directly conditioning on its segments' feature representations $(\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_m)$, e.g., their word embeddings (Mikolov et al., 2013; Pennington et al., 2014), or their composition:

$$\mathbf{p}_r = \hat{f}_\theta(\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_m) \tag{4.1}$$

Alternatively, we can formalize the task under multiple instance learning (MIL), by viewing corpus $\mathcal{C}$ as a collection of labeled *bags* (reviews), each of which is a group of unlabeled *instances* (segments). MIL dictates that the overall sentiment of a review,

y_r, is an unknown function of the unobserved segment-level labels:

$$y_r = f(y_1, y_2, \ldots, y_m) \tag{4.2}$$

A MIL classifier will produce a class distribution $\mathbf{p}_i$ for each segment and additionally learn to combine these into a document-level prediction:

$$\mathbf{p}_i = \hat{g}_{\theta_s}(\mathbf{v}_i)\,, \tag{4.3}$$

$$\mathbf{p}_r = \hat{f}_{\theta_d}(\mathbf{p}_1, \mathbf{p}_2, \ldots, \mathbf{p}_m)\,. \tag{4.4}$$

In sections 4.4 and 4.5, we present two neural architectures. A non-MIL *Hierarchical Network* (HIERNET), which adheres to the the definition of Equation (4.1), and a *Multiple Instance Learning Network* (MILNET) that jointly learns the functions $\hat{g}$ and $\hat{f}$ of Equations (4.3) and (4.4).

4.3.2 Sentiment Detection Tasks

Instead of focusing solely on a single type of fine-grained sentiment analysis, like classification, we evaluate the effectiveness of our methods on the following interrelated tasks, illustrated in Figure 4.3.

Sentiment Classification In the simplest case, a segment-level sentiment predictor is asked to classify each instance according to a set of pre-defined sentiment labels. The classification may be binary (positive vs. negative) or more fine-grained. In this chapter, we deal with a 3-class classification problem (positive, negative and neutral). Sentiment classification is the most commonly addressed task (Kim, 2014; Socher et al., 2013), but leaves little room for investigating the more intricate differences in sentiment between different sentences or clauses.

Polarity Ranking A more useful alternative to classification arises if we relax the assumption that sentiment needs to fall into a set of discrete classes. Instead, we can produce a relative ranking of segments within a review based on their predicted real-valued polarity. The ranked list provides a better understanding of the range of expressed opinions and is better suited for downstream applications, like opinions summarization.

Opinion Extraction A direct application of ranking segments according to their polarity, opinion extraction is the task of selecting the subset of review segments that

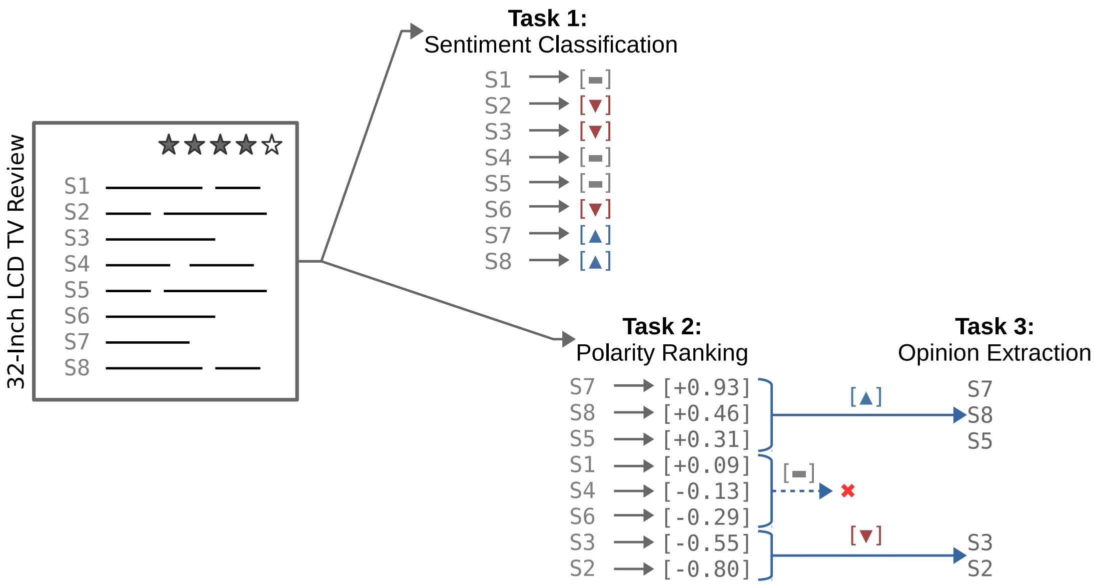


Figure 4.3: An overview of the three tasks relating to sentiment detection that we explore in this chapter: *sentiment classification*, *polarity ranking*, and *single-review opinion extraction*.

convey the most useful comments, thus producing an opinion summary. In this chapter, we consider single-review extractive summaries using only polarity information. In Chapter 6, we explore multi-review opinion summarization.

4.4 Hierarchical Network

The first model we present is a variant of the Hierarchical Attention Network proposed by Yang et al. (2016) and follows a line of research (Denil et al., 2014; Tang et al., 2015a) that tries to hierarchically compose representations of documents from their constituents (i.e. words and sentences). Additionally, it explores the idea that not all parts of a review are important with regards to the overall opinion of the author.

The Hierarchical Network, or HIERNET, consists of three main components as shown in Figure 4.4: a segment encoder that combines a word-level CNN component and a segment-level Gated Recurrent Unit (GRU); an attention-based mechanism for composing segment representations into a single vector; and a document-level softmax classifier.

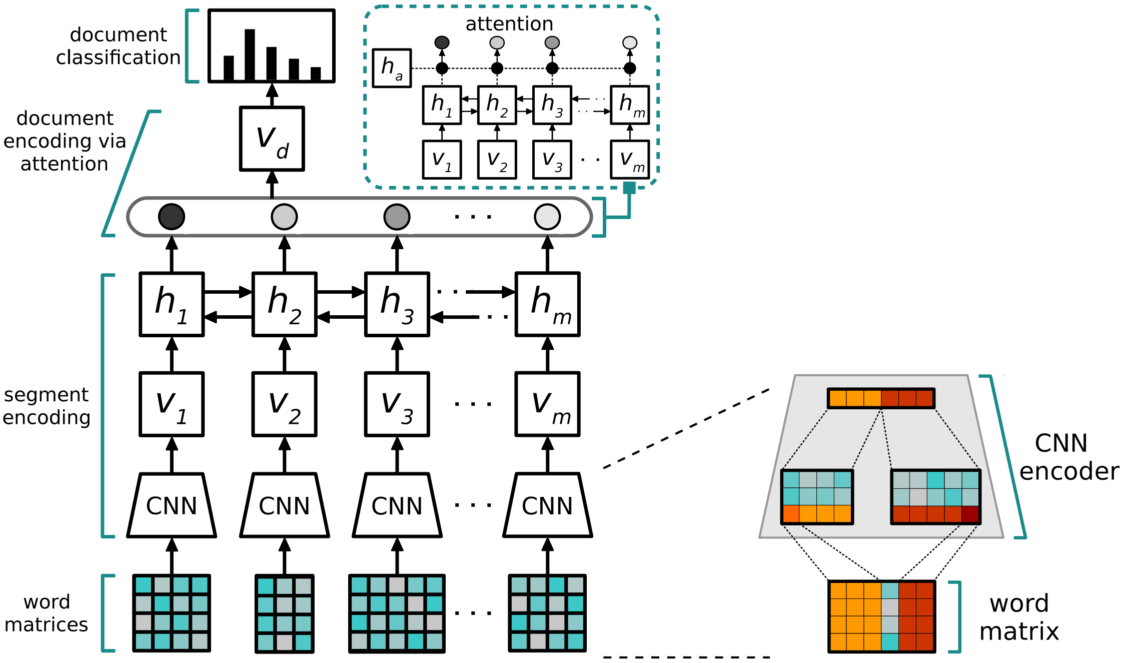


Figure 4.4: An attention-based *Hierarchical Network* (HIERNET) for sentiment analysis. The model predicts the overall sentiment of a review. We describe a method to obtain individual segment predictions via attention in Section 4.6.

4.4.1 Segment Encoding

Word CNN: We use the encoding mechanism of the CNN classifier (Kim, 2014), described in detail in Section 2.2. Let $\mathbf{x}_i$ denote a k-dimensional word embedding of the i-th word in text segment s of length n. The segment's input representation is the concatenation of word embeddings $\mathbf{x}_1,\dots,\mathbf{x}_n$, resulting in word matrix $\mathbf{X}$. Let $\mathbf{X}_{i:i+l}$ refer to the concatenation of embeddings $\mathbf{x}_i,\dots,\mathbf{x}_{i+l}$. A convolution filter $\mathbf{W} \in \mathbb{R}^{l\times k}$, applied to a window of l words, produces a new feature $c_i = \mathrm{ReLU}(\mathbf{W} \circ \mathbf{X}_{i:i+l} + b)$, where ReLU is the *Rectified Linear Unit* non-linearity, '$\circ$' denotes the entrywise product followed by a sum over all elements and $b \in \mathbb{R}$ is a bias term. Applying the same filter to every possible window of word vectors in the segment, produces a feature map $\mathbf{c} = [c_1, c_2, \dots, c_{n-l+1}]$. Multiple feature maps for varied window sizes are applied, resulting in a fixed-size segment representation $\mathbf{v}$, via max-over-time pooling. We refer to the application of convolution to an input word matrix $\mathbf{X}$, as $\mathrm{CNN}(\mathbf{X})$.

Segment GRU: Segment vectors $(\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_m)$ are then passed on to a GRU-based bidirectional recurrent component, inspired by the architectures described in Section 2.3. In the following equations, we use $\mathrm{GRU}(\mathbf{v}_i)$ to refer to the application of the GRU unit at time-step i.

We use separate GRU modules to produce forward and backward hidden vectors, which are then concatenated:

$$\overrightarrow{\mathbf{h}}_i = \overrightarrow{\mathrm{GRU}}(\mathbf{v}_i), \tag{4.5}$$

$$\overleftarrow{\mathbf{h}}_i = \overleftarrow{\mathrm{GRU}}(\mathbf{v}_i), \tag{4.6}$$

$$\mathbf{h}_i = [\overrightarrow{\mathbf{h}}_i, \overleftarrow{\mathbf{h}}_i],\ i \in [1, m]. \tag{4.7}$$

4.4.2 Document Encoding via Attention

A document-level representation can be produced by taking the average of segment hidden vectors: $\mathbf{v}_r = {}^1/_m \sum_i \mathbf{h}_i$. This is, however, a crude way of composing feature vectors, as not all parts of a review convey important sentiment clues. We opt for a segment attention mechanism which rewards text units that are more likely to be good sentiment predictors. The importance of each segment is measured with the aid of a vector $\mathbf{h}_a$, as follows:

$$\mathbf{h}'_i = \tanh(\mathbf{W}_a \mathbf{h}_i + \mathbf{b}_a), \tag{4.8}$$

$$a_i = \frac{exp(\mathbf{h}'^{\top}_i \mathbf{h}_a)}{\sum_i exp(\mathbf{h}'^{\top}_i \mathbf{h}_a)}, \tag{4.9}$$

where Equation (4.8) defines a one-layer MLP that produces an attention vector for the i-th segment. Attention weights a_i are computed as the normalized similarity of each $\mathbf{h}'_i$ with $\mathbf{h}_a$. Vector $\mathbf{h}_a$, which is randomly initialized and learned during training, can be thought of as a trained *key*, able to recognize sentiment-heavy segments. The attention mechanism is depicted in the dashed box of Figure 4.4, with attention weights shown as shaded circles. The whole review is then represented as the weighted average of the segments' hidden vectors:

$$\mathbf{v}_r = \sum_i a_i \mathbf{h}_i. \tag{4.10}$$

4.4.3 Document-level Prediction

A final sentiment prediction is obtained using a softmax classifier:

$$\mathbf{p}_r = \mathrm{softmax}(\mathbf{W}_c \mathbf{v}_r + \mathbf{b}_c), \tag{4.11}$$

where $\mathbf{W}_c$ and $\mathbf{b}_c$ are the classifier's parameters. The model is trained end-to-end on documents with user-generated sentiment labels. We use the negative log likelihood of the document-level prediction as an objective function:

$$L = -\sum_r \log p_r^{(y_r)} \tag{4.12}$$

4.4.4 Segment-level Prediction

HIERNET lacks the ability to naturally produce segment-level predictions. We naively apply its document-level probability distribution to all constituent segments, which can only be effective for reviews of strictly uniform sentiment (Type I; see Figure 3.2) and cannot distinguish between opinions of varying significance. In Section 4.6, we propose a polarity scoring method based the model's attention that overcomes this limitation, thus making HIERNET better suited to handle reviews of uniform sentiment with neutral segments (Type II; see Figure 3.3).

4.5 Multiple Instance Learning Network

Hierarchical neural models like HIERNET have been used (Tang et al., 2015b; Yang et al., 2016) to predict document-level polarity by first encoding sentences and then combining these representations into a document vector. Hierarchical vector composition produces powerful sentiment predictors, but falls short of introspectively judging the polarity of segments with opposing polarity.

Our *Multiple Instance Learning Network* (henceforth MILNET) is based on the following intuitive assumptions about opinionated text. Each segment conveys a degree of sentiment polarity, ranging from very negative to very positive. Additionally, segments have varying degrees of importance, in relation to the overall opinion of the author. The overarching polarity of a text is an aggregation of segment polarities, weighted by their importance. Thus, our model attempts to predict the polarity of segments and decides which parts of the document are good indicators of its overall sentiment, allowing for the detection of sentiment-heavy opinions. An illustration of MILNET is shown in Figure 4.5; the model consists of three components: a CNN segment encoder, a softmax segment classifier and an attention-based prediction weighting module.

4.5.1 Segment Encoding

An encoding $\mathbf{v}_i = \text{CNN}(\mathbf{X}_i)$ is produced for each segment, using the Word CNN architecture described in Section 4.4.

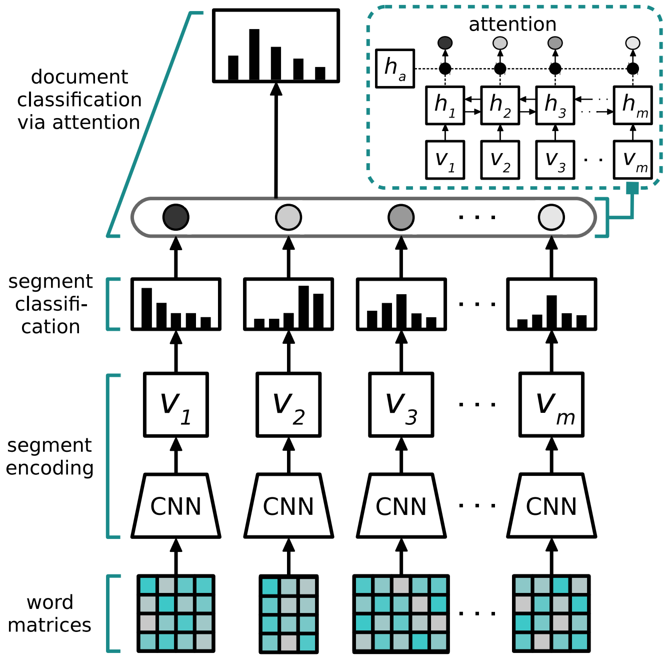


Figure 4.5: Our novel *Multiple Instance Learning Network* (MILNET). The model naturally produces segment-level predictions of potentially opposing polarity, which are then combined using attention.

4.5.2 Segment-level Prediction

Obtaining a separate representation $\mathbf{v}_i$ for every segment in a review allows us to produce individual segment sentiment predictions $\mathbf{p}_i = \langle p_i^{(1)}, \ldots, p_i^{(L)} \rangle$. This is achieved using a softmax classifier:

$$\mathbf{p}_i = \mathrm{softmax}(\mathbf{W}_c \mathbf{v}_i + \mathbf{b}_c)\,, \tag{4.13}$$

where $\mathbf{W}_c$ and $\mathbf{b}_c$ are the classifier's parameters, shared across all segments. Individual distributions $\mathbf{p}_i$ are shown in Figure 4.5 as small bar-charts.

4.5.3 Document-level Prediction

A document-level prediction can be produced by taking the average of segment class distributions: $\mathbf{p}_r^{(c)} = {}^1\!/_m \sum_i p_i^{(c)}\,, c \in [1, L]$. As with the attention-based vector composition of HIERNET, we opt for an attention mechanism which rewards text units that are more likely to be good sentiment predictors.

Again, we produce segment attention weights through the mechanism described in Equations (4.5) – (4.9), but use them to combine segment *predictions* instead of segment vectors. In particular, we obtain a document-level distribution over sentiment labels as the weighted sum of segment distributions (see top of Figure 4.5):

$$p_r^{(c)} = \sum_i a_i p_i^{(c)}\,,\; c \in [1, L]\,. \tag{4.14}$$

The model is trained end-to-end on reviews with document-level sentiment labels in a process identical to HIERNET. We use the negative log likelihood of the document-level prediction as an objective function:

$$L = -\sum_r \log p_r^{(y_r)}\,. \tag{4.15}$$

Our hypothesis is that, in order to make accurate predictions on the document-level, MILNET needs to accurately predict the sentiment of segments. The model has no restriction with regards to the polarity of individual predictors, and is able to handle cases where statements of opposing polarity are intertwined within a single review.

4.6 Polarity Scoring

After training, our models produce segment-level sentiment predictions for unseen texts in the form of class probability distributions. This is achieved explicitly (MILNET) or implicitly, by assigning document-level predictions to every segment (HIERNET). Given our interests in finer-grained sentiment distinctions, and opinion extraction, we transform discrete predictions into real-valued polarity scores as follows.

We introduce a method that takes our model's confidence in the prediction into account, by reducing each segment's class probability distribution $\mathbf{p}_i$ to a single real-valued polarity score. To achieve this, we first define a real-valued *class weight* vector $\mathbf{w} = \langle w^{(1)}, \ldots, w^{(L)} \,|\, w^{(c)} \in [-1, 1] \rangle$ that assigns uniformly-spaced weights to the ordered labelset, such that $w^{(c+1)} - w^{(c)} = \frac{2}{L-1}$. For example, in a 5-class scenario, the class weight vector would be $\mathbf{w} = \langle -1, -0.5, 0, 0.5, 1 \rangle$. We compute the polarity score of a segment s_i as the dot-product of the probability distribution $\mathbf{p}_i$ with vector $\mathbf{w}$:

$$pol_{s_i} = \sum_c p_i^{(c)} w^{(c)} \quad \in [-1, 1]\,. \tag{4.16}$$

As a way of increasing the effectiveness of our methods, we introduce a *gated* extension that uses the attention mechanism of our model to further differentiate between segments that carry significant sentiment cues and those that do not:

$$gtd\text{-}pol_{s_i} = a_i \cdot pol_{s_i}\,, \tag{4.17}$$

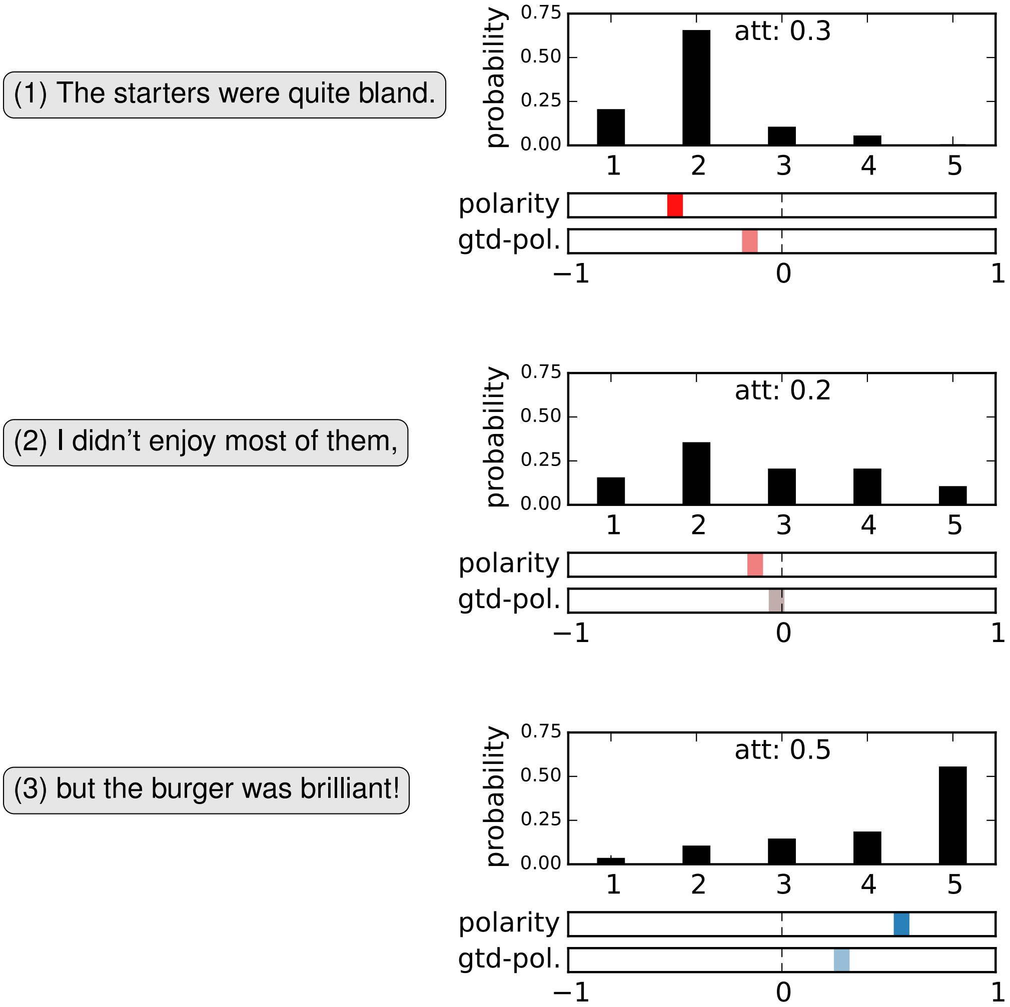


Figure 4.6: Polarity scores (below bar charts) obtained from class probability distributions for three EDUs (left) extracted from a restaurant review. Attention weights (top of bar charts) are used to fine-tune the obtained polarities.

where a_i is the attention weight assigned to the i-th segment. This forces the polarity scores of segments the model does not attend to closer to 0.

An illustration of our polarity scoring functions is provided in Figure 4.6, where the class predictions of three restaurant review segments are mapped to their corresponding polarity scores. We observe that our method produces the desired result; segments 1 and 2 convey negative sentiment and receive negative scores, whereas the third segment is mapped to a positive score. Although the same discrete class label is assigned to the first two, the second segment's score is closer to 0 (neutral) as its class probability mass is more evenly distributed.

The example also illustrates why EDU-based segmentation might be beneficial for opinion extraction. The second and third EDUs correspond to the sentence: *I didn't enjoy most of them, but the burger was brilliant.* Taken as a whole, the sentence conveys mixed sentiment, whereas the EDUs clearly convey opposing sentiment.

Gated polarity will be our method of choice for (a) classifying the sentiment of segments as positive, neutral or negative (via thresholding); (b) ranking segments based on their relative sentiment; (c) using these rankings to extract significant opinions. The system architecture for both HIERNET and MILNET is shown in Figure 4.7.

4.7 Experiments

In this section, we present our experimental evaluation and findings across three main tasks relating to segment-level sentiment detection: classification, polarity ranking and single-review opinion extraction. We first discuss various models used for comparison with our approach in Section 4.7.1 and then present details on implementation and training in Section 4.7.2. Our results are discussed in Section 4.7.3.

4.7.1 Model Comparison

Throughout our experiments, we compare HIERNET and MILNET against the following methods:

Majority: Majority class applied to all instances (classification only).

Seg-CNN: The fully supervised CNN segment classifier presented on Section 2.2. Seg-CNN is trained on OPOSUM's segment-level labels (classification only).

GICF: The Group-Instance Cost Function model introduced by Kotzias et al. (2015). This is an unweighted average prediction aggregation MIL method that uses sentence features from a pre-trained convolutional neural model (classification only).

SO-CAL: State-of-the-art lexicon- and syntax-based system that produces real-valued polarity scores (Taboada et al., 2011).

4.7.2 Model Training and Evaluation

We trained MILNET and HIERNET on OPOSUM's large-scale review collections. OPOSUM contains eight separate training sets of reviews from different domains, namely Yelp (local business), IMDb (movies & TV), laptop bags, bluetooth headsets, boots,

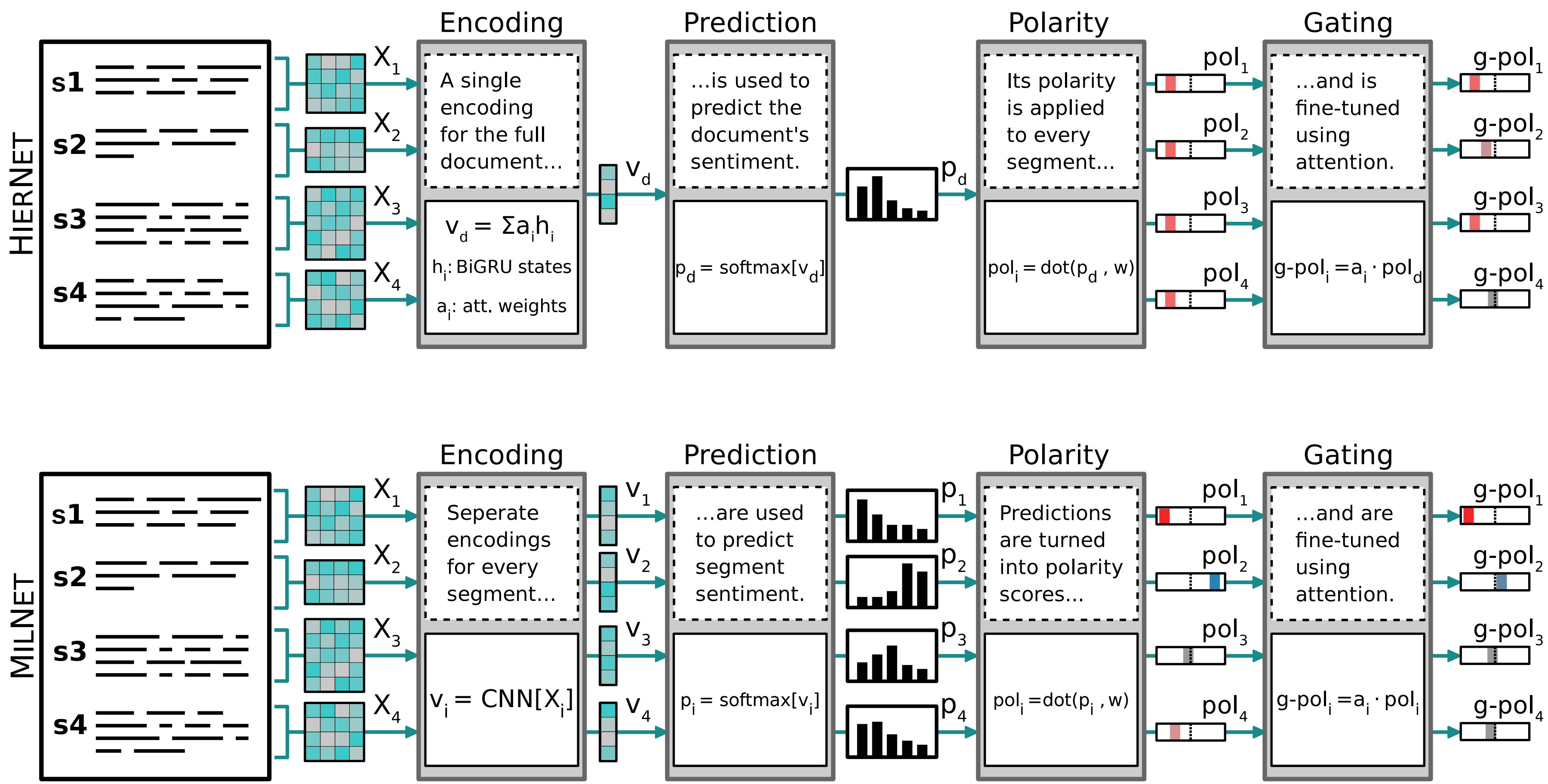


Figure 4.7: System pipelines for HIERNET and MILNET showing four distinct phases for fine-grained sentiment detection: *encoding*, *sentiment prediction*, *polarity scoring*, and *gating*. In HIERNET, segments are encoded into a single vector representation, which is used to predict the overall sentiment of the review. The review's polarity is applied to every constituent segment, and is fine-tuned using gating to uncover neutral statements. MILNET uses individual input vectors to predict the sentiment of each segment independently. Predictions are transformed into polarity scores, which are fine-tuned using gating.

keyboards, televisions and vacuums (see Section 3.2.1 for details). In testing our models, we used the Yelp and IMDb evaluation sets (100 annotated reviews each) for the sentiment classification and opinion extraction tasks, whereas all eight domains were used for polarity ranking. We summarize the statistics of our training and evaluation corpus, OPOSUM, as first described in Chapter 3, and indicate the domains used per task for this chapter in Table 4.1.

We used the Adadelta optimizer (Zeiler, 2012) to train the models for 25 epochs. Mini-batches of 200 documents were organized based on the reviews' segment and document lengths so the amount of padding was minimized. We used 300-dimensional pre-trained word2vec embeddings (Mikolov et al., 2013). We tuned hyper-parameters to maximize document-level classification accuracy on the held-out validation sets of review collections, resulting in the following configuration (unless otherwise noted). For the CNN encoder, we used window sizes of 3, 4 and 5 words with 100 feature maps per window size, resulting in 300-dimensional segment vectors. The GRU hidden vector dimensions for each direction were set to 50 and the attention vector dimensionality to 100. We used L2-normalization and dropout to regularize the softmax classifiers and additional dropout on internal GRU connections.

The fully supervised convolutional segment classifier (Seg-CNN) uses the same window size and feature map configuration as our segment encoder. Seg-CNN was trained and evaluated on different folds of OPOSUM's segment-level labels. Seg-CNN is not directly comparable to MILNET (or HIERNET) due to differences in supervision type (segment vs. document labels) and training size (1K-2K segment labels vs. ∼250K document labels). However, the comparison is indicative of the utility of fine-grained sentiment predictors that do not rely on expensive segment-level annotations. We explore the effect of training size in our models and in relation to Seg-CNN, as part of our classification experiments.

4.7.3 Results

We test competing models on different parts of OPOSUM's sentiment evaluation corpus. For classification, we evaluated the accuracy of our models on the Yelp and IMDb domains and, additionally, on the sentence sentiment corpus of Kotzias et al. (2015). For sentiment ranking, we used all eight OPOSUM domains. The human evaluation of opinion summaries was performed again on Yelp and IMDb.

The OpoSum Corpus

Training Data

	Yelp	IMDb	L. Bags	Bluet/th	Boots	Keyb/s	TVs	Vac/s
Products	–	–	2040	1471	4723	983	1894	1184
Reviews	335K	348K	43K	80K	78K	34K	57K	68K
$^{\text{Sent.}}/_{\text{Rev}}$	8.9	14.0	5.9	7.5	5.5	7.5	10.7	9.0
$^{\text{EDUs}}/_{\text{Rev}}$	19.1	37.4	14.1	18.3	12.7	18.5	26.0	22.0
$^{\text{Words}}/_{\text{Rev}}$	128.3	279.2	98.1	122.5	82.6	127.0	180.4	146.6
Classes	5	10	5	5	5	5	5	5

Segment Polarity Evaluation Data

	Yelp	IMDb	L. Bags	Bluet/th	Boots	Keyb/s	TVs	Vac/s
Reviews	100	100	25	25	25	25	25	25
Sentences	1065	1029	162	159	138	161	173	163
EDUs	2100	2398	365	317	301	344	334	357

Domains Per Task

	Yelp	IMDb	L. Bags	Bluet/th	Boots	Keyb/s	TVs	Vac/s
Sentim. Classification	✓	✓						
Polarity Ranking	✓	✓	✓	✓	✓	✓	✓	✓
1-Doc Opinion Extr.	✓	✓						

Table 4.1: The OpoSum corpus, showing training and evaluation set statistics, as well as the domains used for the three sentiment detection tasks of this chapter. Yelp and IMDb are used on all three tasks (segment-level sentiment classification, polarity ranking, and opinion extraction). For polarity ranking, we also use the six Amazon domains.

4.7.3.1 Sentiment Classification

Real-valued polarity scores produced by HierNet, MilNet and SO-Cal are mapped to discrete labels using two appropriate thresholds $t_1, t_2 \in [-1, 1]$, so that a segment s is classified as negative if $\text{polarity}(s) < t_1$, positive if $\text{polarity}(s) > t_2$ or neutral otherwise.[1] To evaluate performance, we use macro-averaged F1 which is unaffected by class imbalance. We select optimal thresholds using 10-fold cross-validation and report mean scores across folds. The fully supervised Seg-CNN classifiers naturally produces class predictions, and it is trained on the discrete classes of the human-annotated data.

[1] The discretization of polarities is only used for classification purposes and is not necessary for the remaining two tasks, where we only need a relative ranking of segments.

	Sentence-Split		EDU-Split	
Method	Yelp	IMDb	Yelp	IMDb
Majority	19.02†	18.32†	17.03†	21.52†
SO-CAL	56.53†	53.21†	58.16†	60.40
Seg-CNN	56.18†	58.32†	**59.96**	**62.95**†
HierNet$_{\text{NoGT}}$	55.33†	48.47†	51.43†	49.70†
HierNet	56.64†	62.12	58.75†	57.38
MilNet$_{\text{NoGT}}$	61.41	59.99†	59.58	57.71†
MilNet	**63.35**	**63.97**	59.85	59.87

Table 4.2: Segment classification results on the Yelp and IMDb evaluation sets of OpoSum (10-fold cross validation; macro-averaged F1). † indicates that the system in question is significantly different from MilNet (approximate randomization test, $p < 0.05$).

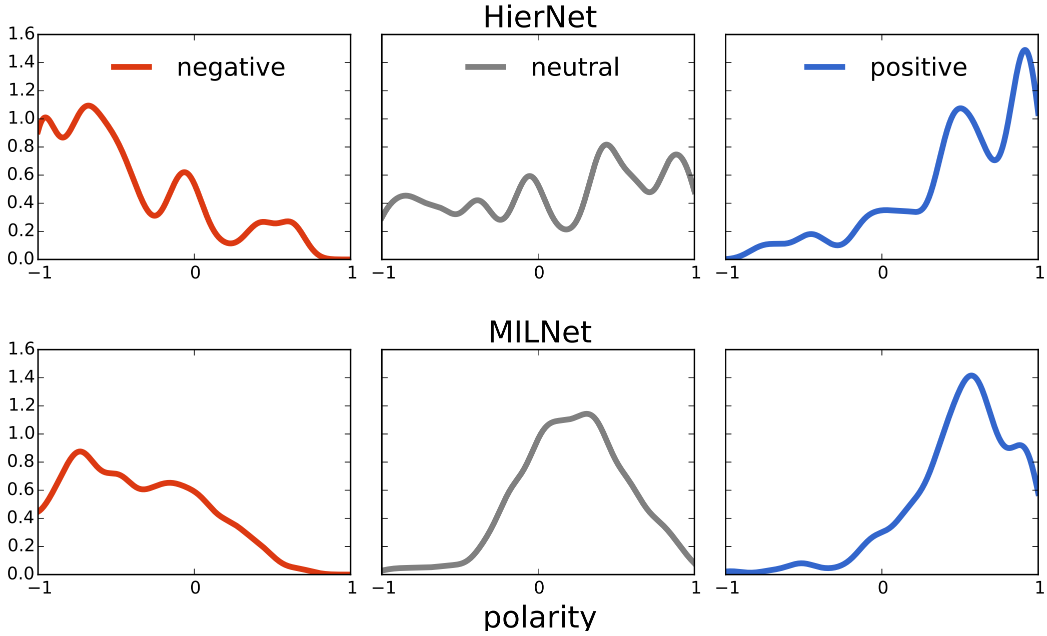


Figure 4.8: Distribution of predicted polarity scores per class (Yelp Sentence test data).

Again, we use 10-fold cross-validation for training and evaluating Seg-CNN (same folds with our methods), and report mean scores across folds.

Table 4.2 summarizes our results. The first block in the table reports the performance of the majority class, SO-Cal and Seg-CNN models. The second block shows our HierNet and MilNet models without (NoGT subscript) and with gated polarities. When considering models that use document-level supervision, MilNet with

	Neutral Sentences		Neutral EDUs	
Method	Non-Gtd	Gated	Non-Gtd	Gated
HIERNET	4.67	36.60	2.39	55.38
MILNET	39.61	44.60	52.10	56.60

Table 4.3: F1 scores for neutral segments on OPOSUM's Yelp test set. The table compares gated and non-gated model variants, showing substantial improvement in neutral segment detection via gating, especially for HIERNET.

Method	Yelp	IMDB
GICF	86.3	86.0
GICF$_{\text{HN}}$	92.9	86.5
GICF$_{\text{MN}}$	93.2	91.0
MILNET	94.0	91.9

Table 4.4: Accuracy scores on the sentence classification test sets introduced in Kotzias et al. (2015).

gated polarities obtains the best classification performance across all four datasets. Interestingly, it performs comparably to Seg-CNN, the fully supervised segment classifier, which provides additional evidence that MILNET can effectively identify segment polarity without the need for segment-level annotations. Our model also outperforms the strong SO-CAL baseline in all but one datasets, which is remarkable given the expert knowledge and linguistic information used to develop the latter. Polarity predictions obtained from HIERNET result in lower classification performance across the board. The use of gated polarities benefits all model configurations, indicating the method's ability to selectively focus on segments with significant sentiment cues.

We further analyzed the polarities assigned by MILNET and HIERNET to positive, negative, and neutral segments. Figure 4.8 illustrates the distribution of polarity scores produced by the two models on the Yelp dataset (sentence segmentation). In the case of negative and positive sentences, both models demonstrate appropriately skewed distributions. However, the neutral class appears to be particularly problematic for HIERNET, where polarity scores are scattered across a wide range of values. In contrast, MILNET is more successful at identifying neutral sentences, as its corresponding distribution has a single mode near zero. Attention gating addresses this issue by moving the polarity scores of sentiment-neutral segments towards zero. This is illustrated in Table 4.3 where we observe that gated variants of both models do a better job at identifying neutral segments. The effect is very significant for HIERNET, while MILNET benefits slightly and remains more effective overall.

In order to examine the effect of training size, we trained multiple models using subsets of the original document collections. We trained on five random subsets for each training size, ranging from 100 documents to the full training set, and tested

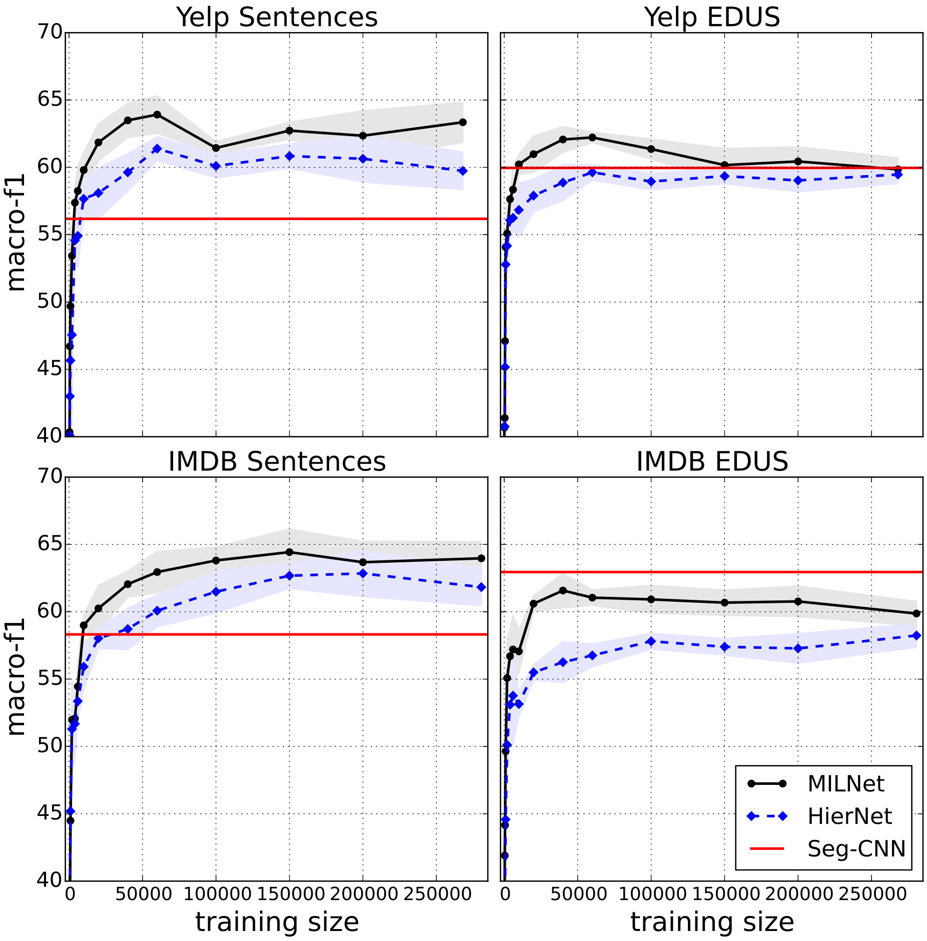


Figure 4.9: Performance of HIERNET and MILNET for varying training sizes on sentence- and EDU-split variants of Yelp and IMDb (evaluation corpus).

segment classification performance on the evaluation data. The results, averaged across trials, are presented in Figure 4.9. With the exception of the IMDB EDU-segmented dataset, MILNET only requires a few thousand training documents to outperform the supervised Seg-CNN. HIERNET follows a similar curve, but is inferior to MILNET. A reason for MILNET's inferior performance on the IMDB corpus (EDU-split) can be low-quality EDUs, due to the noisy and informal style of language used in IMDB reviews.

Finally, we compared MILNET against the GICF model (Kotzias et al., 2015) on their Yelp and IMDB sentence sentiment datasets.[2] Their model requires sentence embeddings from a pre-trained neural model. We used the hierarchical CNN from their work (Denil et al., 2014) and, additionally, pre-trained HIERNET and MILNET

[2] GICF only handles binary labels, which makes it unsuitable for the full-scale comparisons in Table 4.2. Here, we binarize our training datasets and use same-sized sentence embeddings for all four models ($\mathbb{R}^{150}$ for Yelp, $\mathbb{R}^{72}$ for IMDB).

sentence embeddings. The results in Table 4.4 show that MILNET outperforms all variants of GIFC. Our models also seem to learn better sentence embeddings, as they improve GICF's performance on both collections.

4.7.3.2 Polarity Ranking

We now focus on evaluating the ranking of segments within reviews based on their polarity. This experimental setup requires no thresholding mechanism and is closer to real-life applications of opinion mining where the most positive and/or most negative opinions are presented to the user.

Here, we evaluate and compare three models that naturally produce real-valued polarity scores for every review segment: SO-CAL, HIERNET and MILNET. The problem is similar to a retrieval task; depending on whether we are interested in detecting only positive, only negative or salient opinions of either polarity, we set the corresponding segments' true labels to 1 and use *Average Precision* to measure the predicted ranking's quality. We use the original gated polarities (pol_s) to detect positive segments, their inverse ($-1 \times pol_s$) for negative ones, and their absolute values ($|pol_s|$) to retrieve any salient (positive or negative) segment.

Tables 4.5 and 4.6 present the results for our sentence- and EDU-split corpora respectively. Blue-, red- and non-shaded parts correspond to the retrieval of positive, negative and mixed salient segments. For positive sentences and EDUs, the results are mixed. On average, SO-CAL outperforms both HIERNET and MILNET, with the latter performing slightly better overall. The picture is different for negative and mixed segments, where MILNET is clearly superior.

We are also interested in the effect of gating on the opinion rankings. Tables 4.7 and 4.8 show Average Precision scores across domains for non-gated (gray-shaded) and gated variants of HIERNET and MILNET. Gating improves performance almost universally for both models, with differences being very significant for HIERNET.

Finally, we present Average Precision scores separately for reviews of strictly uniform sentiment (Type I), uniform sentiment with neutral comments (Type II) and mixed sentiment (Type III) in Table 4.9. For sentences, HIERNET has the advantage when examining Type I reviews only, which is not surprising, given the perfect alignment between review and segment sentiment in such cases. We observe lower scores across methods for Type II and III reviews, where MILNET performs best. A similar trend is found for EDU-split reviews too, although all three methods perform perfectly for the very few EDU-split Type I reviews.

Sentence-Split

Avg.Pr. (Pos)	Yelp	IMDb	L.Bags	B/tooth	Boots	Keyb/s	TVs	Vac/s	**AVG**
SO-Cal	44.4	31.4	39.3	**50.0**	41.5	36.9	**58.1**	**47.7**	**43.7**
HierNet	43.0	**35.2**	30.8	45.2	**52.8**	**46.1**	43.7	39.1	42.0
MilNet	**50.7**	31.9	**45.2**	45.0	48.5	40.2	48.5	31.5	42.7

Avg.Pr. (Neg)	Yelp	IMDb	L.Bags	B/tooth	Boots	Keyb/s	TVs	Vac/s	**AVG**
SO-Cal	39.2	21.7	42.3	43.5	**40.8**	41.3	45.0	35.5	38.7
HierNet	41.0	31.0	43.6	44.9	16.2	31.2	30.4	53.9	36.5
MilNet	**52.0**	**41.1**	**59.3**	**50.5**	32.9	**64.2**	**50.2**	**58.3**	**51.1**

Avg.Pr. (All)	Yelp	IMDb	L.Bags	B/tooth	Boots	Keyb/s	TVs	Vac/s	**AVG**
SO-Cal	47.3	24.6	43.9	39.0	**50.2**	42.6	**55.5**	40.6	43.0
HierNet	44.2	**35.1**	43.1	**49.3**	39.4	53.2	40.9	48.2	44.2
MilNet	**51.7**	34.7	**49.3**	46.1	44.2	**55.0**	49.1	**48.5**	**47.3**

Table 4.5: Average Precision scores for the sentence-split version of every domain in OpoSum (and on average).

EDU-Split

Avg.Pr. (Pos)	Yelp	IMDb	L.Bags	B/tooth	Boots	Keyb/s	TVs	Vac/s	**AVG**
SO-Cal	47.6	32.1	**33.8**	45.3	**46.5**	**40.1**	**50.2**	35.4	**41.3**
HierNet	44.0	**33.6**	31.9	**47.6**	40.8	39.9	49.6	**38.7**	40.8
MilNet	**48.7**	32.2	32.0	45.7	44.0	37.7	48.4	35.9	40.6

Avg.Pr. (Neg)	Yelp	IMDb	L.Bags	B/tooth	Boots	Keyb/s	TVs	Vac/s	**AVG**
SO-Cal	36.9	29.6	26.7	41.3	37.0	34.5	40.9	44.5	36.4
HierNet	34.3	22.3	24.6	39.9	24.3	36.2	38.7	41.6	32.7
MilNet	**50.0**	**36.3**	**36.6**	**52.4**	**48.9**	**48.5**	**50.7**	**54.9**	**47.3**

Avg.Pr. (All)	Yelp	IMDb	L.Bags	B/tooth	Boots	Keyb/s	TVs	Vac/s	**AVG**
SO-Cal	48.7	30.3	46.9	37.5	49.2	42.7	53.9	45.4	44.3
HierNet	45.9	32.4	37.8	**54.6**	42.5	50.4	53.5	**51.3**	46.0
MilNet	**51.3**	**34.4**	**51.0**	52.0	**53.7**	**57.4**	**58.8**	50.9	**51.2**

Table 4.6: Average Precision scores for the EDU-split version of every domain in OpoSum (and on average).

Sentence-Split									
Avg.Pr	Yelp	IMDb	L.Bags	B/tooth	Boots	Keyb/s	TVs	Vac/s	**AVG**
HierNet	34.1	18.7	38.6	**54.8**	37.8	44.8	35.5	45.7	38.6
	44.2	**35.1**	**43.1**	49.3	**39.4**	**53.2**	**40.9**	**48.2**	**44.2**
MilNet	50.5	30.2	44.1	**57.5**	41.5	51.5	**50.9**	**49.1**	46.9
	51.7	**34.7**	**49.3**	46.1	**44.2**	**55.0**	49.1	48.5	**47.3**

Table 4.7: Average Precision scores without (gray) and with gating (white) for the sentence-split version of every domain in OpoSum (and on average).

EDU-Split									
Avg.Pr.	Yelp	IMDb	L.Bags	B/tooth	Boots	Keyb/s	TVs	Vac/s	**AVG**
HierNet	24.7	14.3	24.4	47.3	36.0	32.3	31.7	29.6	30.0
	45.9	**32.4**	**37.8**	**54.6**	**42.5**	**50.4**	**53.5**	**51.3**	**46.0**
MilNet	42.5	27.2	44.2	**52.3**	49.4	43.5	50.1	50.9	45.0
	51.3	**34.4**	**51.0**	52.0	**53.7**	**57.4**	**58.8**	50.9	**51.2**

Table 4.8: Average Precision scores without (gray) and with gating (white) for the EDU-split version of every domain in OpoSum (and on average).

	Sentence-Split			EDU-Split		
Avg.Pr.	Type I	Type II	Type III	Type I	Type II	Type III
SO-Cal	40.4	30.9	32.4	100.0	35.7	31.2
HierNet	**85.8**	42.1	37.7	100.0	52.4	39.1
MilNet	56.0	**47.1**	**40.6**	100.0	**55.8**	**42.6**

Table 4.9: Average Precision scores across review types for the sentence- and EDU-split versions of OpoSum.

4.7.3.3 Opinion Extraction

For our opinion extraction experiments, we obtained summaries from HierNet and MilNet by selecting the most salient segments (positive and negative) from individual reviews, given a summarization budget. Workers of crowd-sourcing platform Figure Eight (all native English speakers) were shown an original review and a set of extractive, bullet-style summaries, produced by competing systems using a 30% com-

Method	Informativeness	Polarity	Coherence
HierNet^{sent}	43.7	33.6	43.5
MilNet^{sent}	**45.7**	**36.7**	**44.6**
Unsure	10.7	29.6	11.8
HierNet^{edu}	34.2†	28.0†	**48.4**
MilNet^{edu}	**53.3**	**61.1**	45.0
Unsure	12.5	11.0	6.6
MilNet^{sent}	35.7†	33.4†	**70.4**†
MilNet^{edu}	**55.0**	**51.5**	23.7
Unsure	9.3	15.2	5.9
Lead	34.0	19.0†	**40.3**
Random	22.9†	19.6†	17.8†
MilNet^{edu}	**37.4**	**46.9**	33.3
Unsure	5.7	14.6	8.6

Table 4.10: Human evaluation results for the opinion summaries produced in Yelp and IMDb (in preference percentages). † indicates that the system in question is significantly different from MilNet (sign-test, $p < 0.01$).

pression rate. Participants were asked to decide which summary was best according to three criteria: *Informativeness* (Which summary best captures the salient points of the review?), *Polarity* (Which summary best highlights positive and negative comments?) and *Coherence* (Which summary is more coherent and easier to read?). Subjects were allowed to answer “Unsure” in cases where they could not discriminate between summaries. We used all reviews from the Yelp and IMDb evaluation datasets and collected three responses per comparison. For the full instructions of the study, refer to Appendix B.1. We ran four judgment elicitation studies: one comparing HierNet and MilNet when summarizing reviews segmented as sentences, a second one comparing the two models with EDU segmentation, a third which compares EDU- and sentence-based summaries produced by MilNet, and a fourth where EDU-based summaries from MilNet were compared to a Lead (the first *N* words from each document) and a Random (random EDUs) baseline.

Table 4.10 summarizes our results, showing the proportion of participants that preferred each system. The first block in the table shows a slight preference for MilNet across criteria. The second block shows significant preference for MILNET against HIERNET on informativeness and polarity, whereas HIERNET was more often preferred in terms of coherence, although the difference is not statistically significant. The third block compares sentence and EDU summaries produced by MILNET. EDU summaries were perceived as significantly better in terms of informativeness and polarity, but not coherence. This is somewhat expected as EDUs tend to produce more terse and telegraphic text and may seem unnatural due to segmentation errors. In the fourth block we observe that participants find MILNET more informative and better at distilling polarity compared to the LEAD and RANDOM (EDUS) baselines. We should point out that the LEAD system is not a strawman; it has proved hard to outperform by more sophisticated methods (Nenkova, 2005), particularly on the newswire domain.

Example EDU- and sentence-based summaries of a Yelp review, produced by gated variants of HIERNET and MILNET, are shown in Figure 4.10, with attention weights and polarity scores shown for each extracted segment. For both granularities, HIERNET's positive document-level prediction results in a single polarity score assigned to every segment, and further adjusted using the corresponding attention weights. The extracted segments are informative, but fail to capture the negative sentiment of some segments. In contrast, MILNET is able to detect positive and negative snippets via individual segment polarities. Here, EDU segmentation produced a more concise summary with a clearer grouping of positive and negative snippets.

4.8 Summary

In this chapter, we presented two hierarchical neural network models for fine-grained sentiment analysis, including MILNET which is based on multiple instance learning. Our models can be trained on large scale sentiment classification datasets, without the need for segment-level sentiment labels. As a departure from the commonly used vector-based composition, MILNET first predicts sentiment at the sentence- or EDU-level and subsequently combines predictions up the document hierarchy. The attention-based gated polarity scores provide a natural way to detect and extract salient opinions. Experimental results demonstrated the superior performance of MILNET against a more conventional hierarchical network (HIERNET) and other baselines across 3 tasks: segment classification, polarity ranking and single-review opinion extraction. Interestingly, it performed on par with a fully supervised CNN model, indicating that weak supervision is sufficient to train competitive sentiment detectors. In opinion extrac-

Restaurant Review **Rating:** 4/5

As with any family-run hole in the wall, service can be slow. What the staff lacked in speed, they made up for in charm. The food was good, but nothing wowed me. I had the Pierogis while my friend had swedish meatballs. Both dishes were tasty, as were the sides. One thing that was disappointing was that the food was a a little cold (lukewarm). The restaurant itself is bright and clean. I will go back again when i feel like eating outside the box.

		Extracted via HierNet:	**Att.**	**Pol.**
EDU Summaries	▲	The food was good	0.13	+0.26
	▲	but nothing wowed me	0.10	+0.26
	▲	The restaurant itself is bright and clean	0.09	+0.26
	▲	Both dishes were tasty	0.13	+0.26
	▲	I will go back again	0.18	+0.26
		Extracted via MilNet:		
	▲	The food was good	0.16	+0.12
	▲	The restaurant itself is bright and clean	0.12	+0.43
	▲	I will go back again	0.19	+0.15
	▼	but nothing wowed me	0.09	–0.07
	▼	the food was a little cold (lukewarm)	0.10	–0.10

		Extracted via HierNet:	**Att.**	**Pol.**
Sentence Summaries	▲	Both dishes were tasty, as were the sides.	0.12	+0.23
	▲	The food was good, but nothing wowed me.	0.18	+0.23
	▲	One thing that was disappointing was that the food was a little cold (lukewarm).	0.22	+0.23
		Extracted via MilNet:		
	▲	Both dishes were tasty, as were the sides	0.13	+0.26
	▲	I will go back again when I feel like eating outside the box	0.20	+0.59
	▼	The food was good, but nothing wowed me.	0.18	–0.12

Figure 4.10: Example EDU- and sentence-based opinion summaries of a Yelp review, produced by HierNet$_{gt}$ and MilNet$_{gt}$.

tion, our human evaluation studies also showed that MILNET summaries are preferred by participants and are effective at capturing informativeness and polarity, especially when using EDU segments.

In the following chapter, we move away from the analysis of sentiment in reviews and focus on aspect extraction, i.e., the identification of specific features of the reviewed entities and the review segments that discuss them. Aspect extraction, combined with sentiment analysis, will form the basis of our multi-document opinion summarization framework, presented in Chapter 6.

Chapter 5

Aspect Extraction with Minimal Supervision

In this chapter, we shift our focus away from the detection of sentiment polarity and investigate a second important dimension of opinion analysis in reviews; the extraction and categorization of *aspect-specific* expressions. Aspects are identifiable features of the entity under review (e.g., the *sound* of a television), whose characteristics may influence customer satisfaction and, therefore, are the main targets of reviewer opinions.

A number of tasks relating to the detection of aspects have been addressed in previous literature (Liu and Zhang, 2012). Here, we define and contrast two particular formulations of the problem; *aspect discovery*, a task similar to topic modeling where the aspects are not known beforehand; and *aspect extraction*, where the goal is to identify aspect-specific review segments and group them under a predefined set of categories.[1]

We first present a state-of-the-art neural topic model that has been recently applied to aspect discovery, improving upon traditional LDA-type methods (Brody and Elhadad, 2010; Mukherjee and Liu, 2012; Yan et al., 2013). We argue that the unsupervised nature of such techniques is offset by their inherent need for post-hoc human interpretation of the discovered aspects. On that basis, we present our Multi-Seed Aspect Extractor, a weakly supervised aspect extraction model that requires minimal human intervention and no large-scale annotation efforts for training. As part of our OPOSUM corpus, we create a new dataset for the evaluation of aspect discovery and extraction models and experimentally show that our approach brings consistent improvements over the original model across product domains.

[1] Previous literature is inconsistent in naming the different task formulations relating to aspect detection. In an effort to avoid confusion, we provide our own definitions of aspect discovery and extraction and refer to them consistently throughout the chapter.

5.1 Introduction

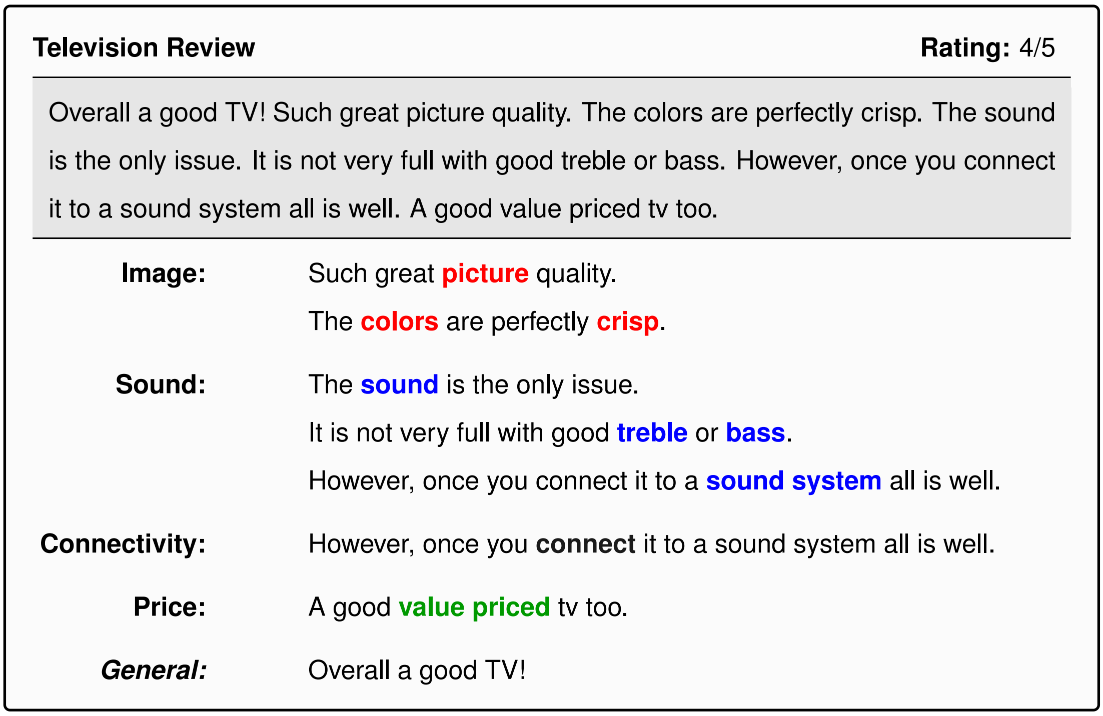
Opinion analysis from user reviews can take many forms; from the simple prediction of a reviewer's attitude towards a product and its features; to more complex tasks, like the generation of abstractive summaries from multiple reviews that highlight popular opinions in an easy to digest format. In its most practical instantiation, opinion analysis primarily combines two components of opinionated expressions: the sentiment polarity they convey and the product aspects they target (Pang and Lee, 2008; Liu et al., 2012). Up until now, we have extensively covered the former, while ignoring the latter.

Aspects are identifiable features or attributes of reviewed items that are specific to a particular product domain. For example, *image quality*, *sound quality*, *connectivity*, *durability*, *price*, and so on, are different aspects of televisions. Similarly, *suction power*, *manoeuvrability*, *noise level*, *accessories*, *price*, etc., are aspects of vacuum cleaners (notice that some aspects may be shared across domains, e.g., their *price*).

Multiple individual opinions form the overall attitude of a reviewer and often convey diverging sentiment, as we saw in Chapter 3 (for examples, see Figure 3.4). It is reasonable to think that opinions of opposing polarity within a single review are likely to target different aspects; imagine a customer enjoying the *suction power* of a vacuum, but getting upset about the lack of *accessories*. Even in reviews of mostly uniform sentiment, a specific aspect (like the *camera* of a smartphone) may be very important to some users, but irrelevant to others. Hence, methods that attempt to distill useful opinions for easier consumption need to take aspect information into account.

Figure 5.1 shows an example of a television review, accompanied by an analysis of the aspect expressions it contains. Comments about its image quality (e.g., *"Such great picture quality"*), sound (e.g., *"It is not very full with good treble or bass"*), price (e.g., *"A good value priced tv too"*), and so on, are categorized accordingly, and aspect-denoting words are highlighted. Aspect detection systems may try to extract individual words or phrases (like those highlighted in boldface), or whole segments. The methods presented in this chapter attempt the latter, as we use clauses as our unit of extraction.

In accordance with the overarching theme of the thesis, we explore the extraction and categorization of aspect expressions based on weakly supervised learning methods. We use an unsupervised neural topic model, the *Aspect-Based Autoencoder* (ABAE; He et al., 2017), as a point of departure and explore the different ways in which domain knowledge may be injected into its architecture. ABAE discovers topics (i.e., potential



Television Review **Rating:** 4/5

Overall a good TV! Such great picture quality. The colors are perfectly crisp. The sound is the only issue. It is not very full with good treble or bass. However, once you connect it to a sound system all is well. A good value priced tv too.

Image:	Such great **picture** quality. The **colors** are perfectly **crisp**.
Sound:	The **sound** is the only issue. It is not very full with good **treble** or **bass**. However, once you connect it to a **sound system** all is well.
Connectivity:	However, once you **connect** it to a sound system all is well.
Price:	A good **value priced** tv too.
General:	Overall a good TV!

Figure 5.1: Example of aspect-based analysis of a television review. Review segments are grouped under aspect categories (e.g., Image, Sound, Connectivity) and aspect-denoting words are shown in boldface.

aspects) in review text without any supervision. However, as with most topic-modeling approaches, it requires post-hoc interpretation of the obtained results that usually involves a human expert.

Instead of relying on a human-provided mapping between discovered topics and aspects, we guide the aspect extraction model towards aspects of interest during training. In particular, our *Multi-Seed Aspect Extractor* (MATE) takes advantage of two sources of weak supervision. Firstly, *seed words*, i.e., aspect-signaling terms used to initialize the model's aspect descriptors. Secondly, *multi-tasking*, with a product domain classification objective used to exploit the correlations between aspect- and domain-denoting terms.

5.2 Related Work

The identification of aspects and the expressions that discuss them, has been researched extensively as a stand-alone task (Titov and McDonald, 2008b; He et al., 2017), and as part of aspect-based sentiment analysis (Mei et al., 2007; Mukherjee and Liu, 2012;

Lazaridou et al., 2013) or opinion summarization systems (Hu and Liu, 2004; Titov and McDonald, 2008a; Lu et al., 2009).

The task can be formulated in many ways depending on the granularity of extracted units (i.e., *words, phrases, or whole expressions*), the assumptions made regarding the aspects themselves (i.e., *whether they are known beforehand*), and the amount of supervision used (i.e., *the extent to which aspect-annotated data is available*). Early work on aspect detection focused mainly on the extraction of aspect-denoting terms or phrases, without addressing their grouping into coarse-grained categories. When couched as an unsupervised task, solutions usually involve a combination of association rule mining, syntactic analysis and sentiment lexicons (Hu and Liu, 2004; Popescu and Etzioni, 2005; Ku et al., 2006; Blair-Goldensohn et al., 2008; Qiu et al., 2011; Liu et al., 2012). In their seminal opinion summarization work, Hu and Liu (2004) used association rule mining to identify adjective-noun pairs (e.g., *"great sound"*) that frequently appeared in proximity. This line of work has been subsequently extended and improved. Popescu and Etzioni (2005) used a wider array of syntactic rules and *Point-wise Mutual Information* (PMI) to extract a hierarchy of product aspects and the opinions that discuss them. Qiu et al. (2011) investigated the relations of sentiment words and aspects using bootstrapping. Liu et al. (2012) formulated the task as a word alignment problem, and applied a translation algorithm to mine associations between sentiment and aspect words. A common theme among these methods is their focus on extracting individual words or phrases. In contrast, we are interested in syntactically coherent segments, namely clauses.

Several supervised techniques have also been proposed, where the task is viewed as a sequence labeling problem. In this case, the training signal comes in the form of word-level labels that indicate the beginning and end of aspect expressions. Earlier efforts used Conditional Random Fields (CRFs) and sentence structure information to extract aspect expressions from English (Li et al., 2010) and Chinese (Ma and Wan, 2010) reviews. Hidden-Markov Models have also been employed for the task (Jin and Ho, 2009). As the use of deep learning became more widespread, sequence modeling networks have dominated the literature. Liu et al. (2015) employed recurrent neural networks, whereas Yin et al. (2016) used dependency-based embeddings as features in a CRF. Wang et al. (2016) combined a recursive neural network with CRFs to jointly model aspect and sentiment terms. Again, sequence labeling methods have mostly focused on phrase-level extraction, although they are applicable on the segment-level too, if provided with appropriate training data. The need for large-scale human-annotated datasets is an obvious shortcoming, which we attempt to circumvent by injecting minimal domain knowledge in a model that requires no direct supervision.

While the supervised and unsupervised methods mentioned above only deal with the detection of aspect expressions, much previous research has also tried to simultaneously detect expressions and group them based on the aspects they discuss. The majority of such efforts are based on topic modeling. Traditional topic models which operate on the document level, like pLSA (Hofmann, 1999) and LDA (Blei et al., 2003), lack the ability to capture the highly convoluted mentions of aspects within reviews. This shortcoming has been addressed by a variety of extensions that discover global and local topics (Titov and McDonald, 2008b), identify key phrases from noisy annotations (Branavan et al., 2008), jointly model aspect and sentiment words (Lazaridou et al., 2013), or use semi-supervised learning (Lu and Zhai, 2008; Mukherjee and Liu, 2012). The work of Mukherjee and Liu (2012) is most relevant to ours, as they propose the use of aspect-specific *seed words* to inject domain knowledge to their otherwise unsupervised method, thus guiding the topic model towards meaningful aspects.

Neural networks have also infiltrated topic modeling approaches, as He et al. (2017) proposed the Aspect-Based Autoencoder (ABAE) to discover fine-grained aspects without supervision. Their model learns continuous aspect representations in a word embedding space and, at the same time, an aspect classifier that predicts the most likely category for each input segment. Their experiments showed significant improvements over LDA-style approaches. However, their method, like most topic models, requires a mapping from discovered aspects to actual ones. ABAE forms the basis of our aspect extractor and is presented in detail in Section 5.4.

5.3 Preliminaries

In this section, we provide a formal definition of review data with a focus on aspect detection and categorization, to familiarize the reader with the concepts mentioned in the remainder of the chapter. Then, we define and contrast two formulations of the detection task, *aspect discovery* and *aspect extraction*.

5.3.1 Definitions

Let C denote a corpus of reviews from a domain d_C, e.g., televisions or keyboards. The corpus contains a set of reviews $R_C = \{r_i\}_{i=1}^{|R_C|}$ expressing customer opinions. Each

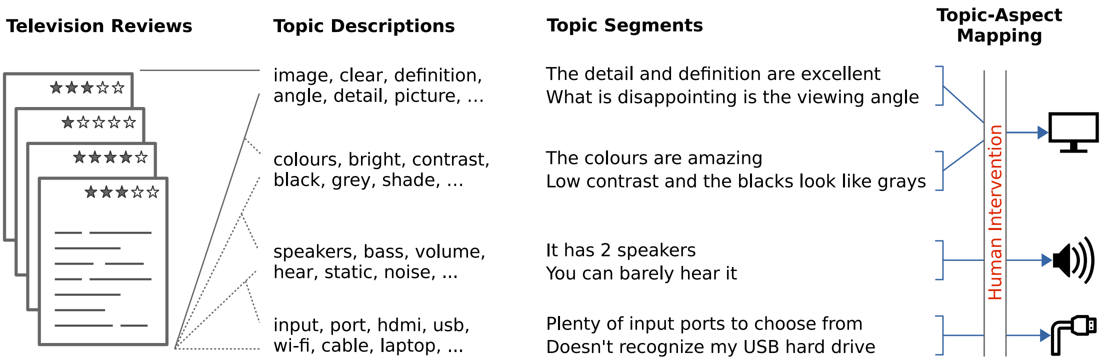

Television Reviews
Topic Descriptions
Topic Segments
Topic-Aspect Mapping
image, clear, definition, angle, detail, picture, ...
colours, bright, contrast, black, grey, shade, ...
speakers, bass, volume, hear, static, noise, ...
input, port, hdmi, usb, wi-fi, cable, laptop, ...
The detail and definition are excellent
What is disappointing is the viewing angle
The colours are amazing
Low contrast and the blacks look like grays
It has 2 speakers
You can barely hear it
Plenty of input ports to choose from
Doesn't recognize my USB hard drive
Human Intervention

Figure 5.2: The task of *Aspect Discovery*. Reviews from a particular domain form the input to the system and the number of topics to be discovered is a parameter. The system learns a description for each topic (e.g., distribution over words) and groups review segments accordingly. Then, a human expert maps the discovered topics to actual product aspects.

review r is split into segments $(s_1, \ldots, s_m)$, where each segment s_j is in turn viewed as a sequence of words $(w_{j1}, \ldots, w_{jn})$. A segment can be a sentence, a phrase, or in our case an *Elementary Discourse Unit* (EDU; Mann and Thompson 1988) obtained from a *Rhetorical Structure Theory* (RST) parser (Feng and Hirst, 2012). EDUs roughly correspond to clauses and have been shown to facilitate performance in summarization (Li et al., 2016), document-level sentiment analysis (Bhatia et al., 2015), and single-document opinion extraction, as indicated by our findings in Chapter 4.

A segment may discuss zero or more *aspects*, i.e., different product attributes. We use $A_C = \{a_i\}_{i=1}^{K}$ to refer to the aspects pertaining to domain d_C. For example, *picture quality*, *sound quality*, and *connectivity* are all aspects of televisions. By convention, a *general* aspect is assigned to segments that do not discuss any specific aspects. We use $A_s \subseteq A_C$ to denote the set of aspects mentioned in segment s.

5.3.2 Aspect Detection Tasks

The nature and applicability of methods relating to aspect detection and categorization varies with the assumptions we make about aspects A_C. Most methods assume no prior knowledge about the aspects themselves, while a few (Mukherjee and Liu, 2012) try to directly guide aspect extraction towards specific aspects of interest. We differentiate between the two formulations below.

Aspect Discovery In a purely unsupervised scenario, an aspect detection method has no preconceptions about the aspects that are discussed in a dataset of reviews.

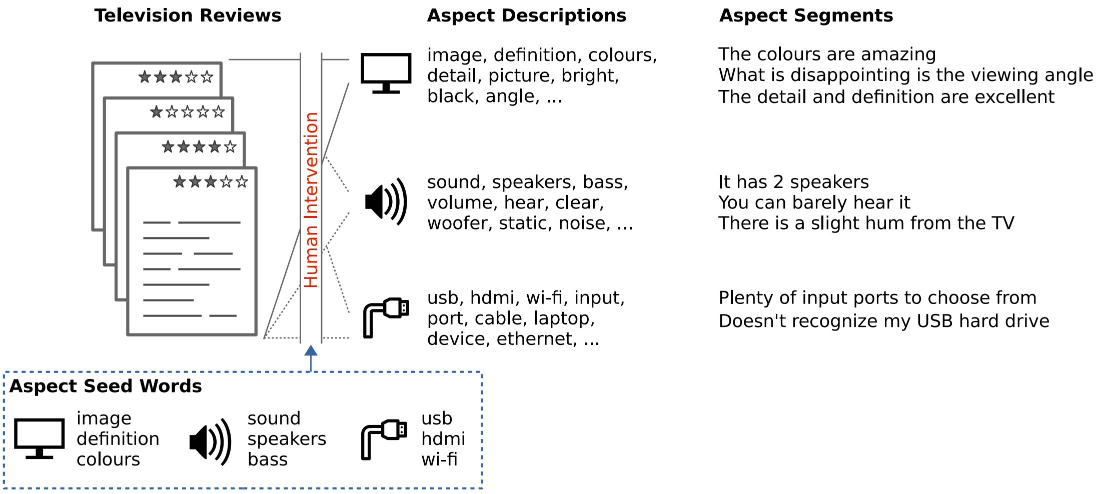

Television Reviews
Human Intervention
Aspect Descriptions
image, definition, colours, detail, picture, bright, black, angle, ...
sound, speakers, bass, volume, hear, clear, woofer, static, noise, ...
usb, hdmi, wi-fi, input, port, cable, laptop, device, ethernet, ...
Aspect Segments
The colours are amazing
What is disappointing is the viewing angle
The detail and definition are excellent
It has 2 speakers
You can barely hear it
There is a slight hum from the TV
Plenty of input ports to choose from
Doesn't recognize my USB hard drive
Aspect Seed Words
image
definition
colours
sound
speakers
bass
usb
hdmi
wi-fi

Figure 5.3: The task of *Aspect Extraction*. Reviews from a particular domain and concise descriptions of the aspects of interest (e.g., aspect seed words provided by an expert) are the system's input. The system may expand the aspect descriptions to a wider set of words and groups review segment accordingly. There is no requirement for post-hoc interpretation of aspects.

Therefore, a corpus C is the only input to the system and the number of topics to be *discovered* is a parameter. After training, a human expert will usually observe the induced topics and map them to actual aspects. This allows for the evaluation of the system against labeled data and facilitates downstream applications like aspect-based opinion summarization. An illustration of the discovery task is provided in Figure 5.2. A dataset of television reviews is used as input to discover four topics. The induced topic descriptions are shown as word lists, each of which is paired with a few matching segments. During post-hoc analysis, two of the topics are mapped to the *image quality* aspect, while the remaining two are mapped to *sound quality* and *connectivity*, respectively.

Aspect Extraction When employing an aspect detection system, it is reasonable to assume that the user has some idea of the important aspects that pertain to the products of interest. In some cases, there may even be specific aspects that one wants to target. As a means of addressing specific user needs, Mukherjee and Liu (2012) first formulated aspect detection as a weakly supervised task, where prior knowledge about the aspects of interest is injected into the model and guides its training. In such a setting, the model's internal description of topics (e.g., word probabilities) may be initialized to reflect the user's prior assumptions. Having that as a starting point, an *aspect ex-*

traction method will use the input review dataset to expand its understanding of the aspects in question, and group the relevant review segments accordingly. This is different from the unsupervised discovery case in the sense that no new aspects can be discovered over and above what has been postulated initially. An abstraction of the process is shown in Figure 5.3, where keywords for three aspects of the televisions domain are used to guide the extraction system towards a meaningful grouping of segments.

We believe that aspect extraction provides a more realistic formulation for the grouping of user opinions based on their aspects. In practice, neither aspect discovery nor aspect extraction is truly unsupervised, as they both require human intervention at different stages of operation. By injecting prior knowledge, an aspect extraction model may identify relevant opinions in a more targeted manner.

In the following section, we present the aspect discovery model developed by He et al. (2017), before moving on to our own Multi-Seed Aspect Extractor in Section 5.5.

5.4 Aspect-Based Autoencoder

The *Aspect-Based Autoencoder* (ABAE; He et al. 2017) is an adaptation of the *Relationship Modeling Network* (Iyyer et al., 2016), originally designed to identify attributes of fictional book characters and their relationships. The model, illustrated in Figure 5.4, learns a segment-level aspect predictor without supervision by attempting to reconstruct the input segment's encoding as a linear combination of aspect embeddings. ABAE starts by pairing each word w with a pre-trained word embedding $\mathbf{v}_w \in \mathbb{R}^d$, thus constructing a word embedding dictionary $\mathbf{L} \in \mathbb{R}^{V \times d}$, where V is the size of the vocabulary. The model also keeps an aspect embedding dictionary $\mathbf{A} \in \mathbb{R}^{K \times d}$ (shown as a multi-coloured $[4 \times 2]$ matrix in Figure 5.4), where K is the number of aspects to be identified and i-th row $\mathbf{a}_i \in \mathbb{R}^d$ is a point in the word embedding space. Matrix $\mathbf{A}$ is initialized using the centroids from a k-means clustering on the vocabulary's word embeddings. The model is, thus, agnostic about the specific aspects expected to be found in reviews.

The autoencoder, first produces a vector $\mathbf{v}_s$ for review segment $s = (w_1, \ldots, w_n)$ using an *attention encoder* that learns to attend to aspect words. A segment encoding

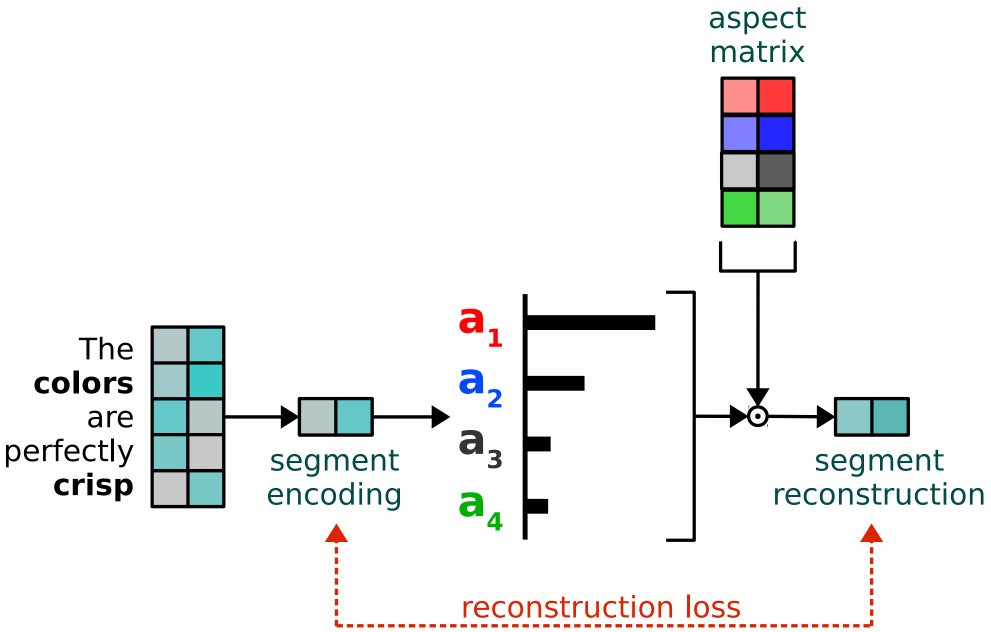


Figure 5.4: The *Aspect-Based Autoencoder* (ABAE). An attention encoder is used to obtain a segment representation, which is then reconstructed as the weighted sum of aspect embeddings. The process gradually learns an aspect predictor and aspect representations.

is computed as the weighted average of word vectors:

$$\mathbf{v}_s = \sum_{i=1}^{n} c_i \mathbf{v}_{w_i} \tag{5.1}$$

$$c_i = \frac{exp(u_i)}{\sum_{j=1}^{n} exp(u_j)} \tag{5.2}$$

$$u_i = \mathbf{v}_{w_i}^{\mathsf{T}} \cdot \mathbf{M} \cdot \mathbf{v}'_s , \tag{5.3}$$

where c_i is the i-th word's attention weight, $\mathbf{v}'_s$ is a simple average of the segment's word embeddings and attention matrix $\mathbf{M} \in \mathbb{R}^{d \times d}$ is learned during training.

Vector $\mathbf{v}_s$ is fed into a softmax classifier to predict a probability distribution over K aspects:

$$\mathbf{p}_s^{asp} = \text{softmax}(\mathbf{W}\mathbf{v}_s + \mathbf{b}) , \tag{5.4}$$

where $\mathbf{W} \in \mathbb{R}^{K \times d}$ and $\mathbf{b} \in \mathbb{R}^{K}$ are the classifier's weight and bias parameters. Distribution $\mathbf{p}_s^{asp}$ is shown as a colour-coded bar chart in Figure 5.4. The segment's vector is then reconstructed as the weighted sum of aspect embeddings:

$$\mathbf{r}_s = \mathbf{A}^{\mathsf{T}} \mathbf{p}_s^{asp} . \tag{5.5}$$

The model is trained by minimizing a reconstruction loss $J_r(\theta)$ that uses randomly sampled segments $n_1, n_2, \ldots, n_{k_n}$ as negative examples:[2]

$$J_r(\theta) = \sum_{s \in C} \sum_{i=1}^{k_n} \max(0, 1 - \mathbf{r}_s \mathbf{v}_s + \mathbf{r}_s \mathbf{v}_{n_i}) \tag{5.6}$$

ABAE is essentially a neural aspect discovery model; it discovers topics which will hopefully map to aspects, without any preconceptions about the aspects themselves, a feature shared with most previous LDA-style approaches (Titov and McDonald, 2008a; He et al., 2017; Mukherjee and Liu, 2012). A many-to-one mapping between discovered topics and genuine aspects which is performed manually, as described previously.

5.5 Multi-Seed Aspect Extractor

Aspect discovery is advantageous since it assumes nothing more than a set of relevant reviews for a product and may discover unusual and interesting aspects (e.g., whether a plasma television has protective packaging). However, it suffers from the fact that the identified aspects are fine-grained, they have to be interpreted post-hoc, and manually mapped to coarse-grained ones.

We present a weakly supervised model for aspect extraction which follows the work of Mukherjee and Liu (2012) and requires little human involvement. For every aspect $a_i \in A_C$, we assume there exists a small set of seed words $\{sw_j\}_{j=1}^{l}$ which are good descriptors of a_i. We can think of these *seeds* as query terms that someone would use to search for segments discussing a_i. They can be set manually by a domain expert or selected using a small number of aspect-annotated reviews (see Section 5.5.1). Figure 5.5 (top) depicts four television aspects (*image*, *sound*, *connectivity* and *price*) and three of their seeds in word embedding space. MATE replaces ABAE's aspect dictionary with multiple seed matrices $\{\mathbf{A}_1, \mathbf{A}_2, \ldots, \mathbf{A}_K\}$. Every matrix $\mathbf{A}_i \in \mathbb{R}^{l \times d}$, contains one row per seed word and holds the seeds' word embeddings, as illustrated by the set of $[3 \times 2]$ matrices in Figure 5.5.

MATE still needs to produce an aspect matrix $\mathbf{A} \in \mathbb{R}^{K \times d}$, in order to reconstruct the input segment's embedding. We accomplish this by reducing each seed matrix to a single aspect embedding with the help of seed weight vectors $\mathbf{z}_i \in \mathbb{R}^l$ ($\sum_j z_{ij} = 1$), and

[2] ABAE also uses a uniqueness regularization objective to discourage the discovery of aspects that are too similar to each other. The term is not shown here and not used in our Multi-Seed Aspect Extractor.

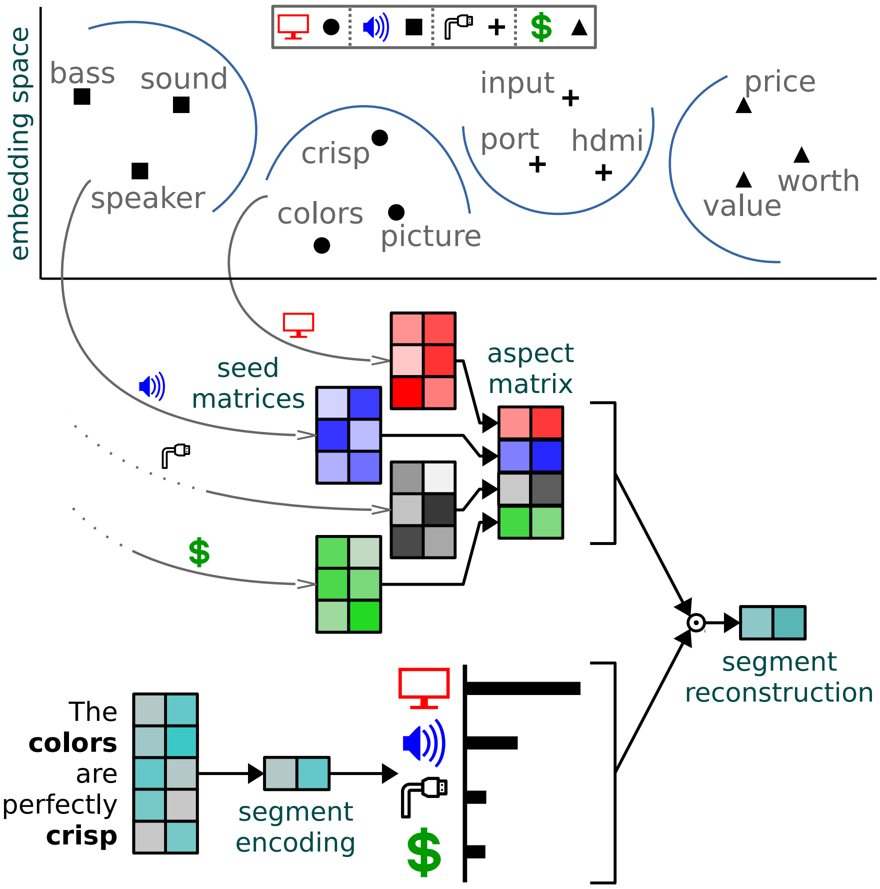


Figure 5.5: The *Multi-Seed Aspect Extractor* (MATE). Our model introduces seed words that are used to construct informed aspect representations. Aspect predictions correspond to actual aspects, and no post-hoc human intervention is required.

concatenating the results into an aspect matrix ([4 × 2] matrix in Figure 5.5):

$$\mathbf{a}_i = \mathbf{A}_i^\top \mathbf{z}_i \tag{5.7}$$

$$\mathbf{A} = [\mathbf{a}_1^\top; \ldots; \mathbf{a}_K^\top]\,. \tag{5.8}$$

The segment is reconstructed as in Equation (5.5). Weight vectors $\mathbf{z}_i$ can be uniform, fixed to specific values, or set dynamically for each input segment, based on the similarity of its encoding to each seed embedding. We describe these variants below.

Uniform Seed Weights In the simplest case, seed weights are set uniformly. This is equivalent to setting aspect embedding $\mathbf{a}_i$ to be the centroid of a_i's seed embeddings, and assumes that all seed words are equally representative of the particular aspect.

Fixed Seed Weights An obvious generalization of the uniform case arises if we set the seed weights to specific values. For example, if we expect the word *"sound"* to be twice as important as the words *"speaker"* and *"bass"* with regards to the aspect *sound*

Aspect	**Top Terms**
Image	picture color quality black bright
Sound	sound speaker quality bass loud
Connectivity	hdmi port computer input component
Price	price value money worth paid
Apps & Interface	netflix user file hulu apps
Ease of Use	easy remote setup user menu
Customer Service	paid support service week replace
Size & Look	size big bigger difference screen
General	tv bought hdtv happy problem

Table 5.1: Highest ranked words for television aspects from a review corpus according to our seed selection Equation (5.10).

quality, we may set their weights to 0.5, 0.25 and 0.25 respectively.

Dynamic Seed Weights We can also obtain segment-specific seed weight vectors that reflect the contents of each input segment and, hopefully, facilitate the reconstruction of its encoding. To achieve this, we use dynamic weights computed for every input segment s individually, based on its encoding's softmax-normalized cosine similarity with seed word embedding $\mathbf{a}_{ij}$:

$$z_{ij} = \frac{e^{cos(\mathbf{v}_s,\mathbf{a}_{ij})}}{\sum_j e^{cos(\mathbf{v}_s,\mathbf{a}_{ij})}} . \tag{5.9}$$

5.5.1 Data-driven Seed Selection

When a small number of aspect-annotated reviews are available, seeds and their seed weights can be selected automatically. To obtain a ranked list of terms that are most characteristic for each aspect, we use a variant of the *clarity* scoring function which was first introduced in information retrieval (Cronen-Townsend et al., 2002). Clarity measures how much more likely it is to observe word w in the subset of segments that discuss aspect a, compared to the corpus as a whole:

$$\text{score}_a(w) = t_a(w)\log_2\frac{t_a(w)}{t(w)} , \tag{5.10}$$

where $t_a(w)$ and $t(w)$ are the l_1-normalized *tf-idf* scores of w in the segments annotated with aspect a and in all annotated segments, respectively. Higher scores indicate higher term importance and truncating the ranked list of terms gives a fixed set of seed words, as well as their seed weights by normalizing the scores to add up to one. Table 5.1 shows the highest ranked terms obtained for every aspect in the *televisions* domain of our corpus (see Section 5.6 for a detailed description of our aspect data).

5.5.2 Multi-Task Objective

MATE and ABAE rely on the attention encoder to identify and attend to each segment's aspect-signalling words. The reconstruction objective only provides a weak training signal, so we devise a multi-task extension to enhance the encoder's effectiveness without additional annotations.

We assume that aspect-relevant words not only provide a better basis for the model's aspect-based reconstruction, but are also good indicators of the product's domain. For example, the words *colors* and *crisp*, in the segment "*The colors are perfectly crisp*" should be sufficient to infer that the segment comes from a television review, whereas the words *keys* and *type* in the segment "*The keys feel great to type on*" are more representative of the keyboard domain. Additionally, both word pairs are characteristic of specific aspects, namely *image quality* and *comfort*, respectively.

Let $\mathcal{C}_{all} = \mathcal{C}_1 \cup \mathcal{C}_2 \cup \dots$ denote the union of multiple review corpora, where $\mathcal{C}_1$ is considered *in-domain* and the rest are considered *out-of-domain*. We use $d_s \in \{d_{\mathcal{C}_1}, d_{\mathcal{C}_2}, \dots\}$ to denote the true domain of segment s and define a classifier that uses the vectors from our segment encoder as inputs:

$$\mathbf{p}_s^{dom} = \text{softmax}(\mathbf{W}_{\mathcal{C}}\mathbf{v}_s + \mathbf{b}_{\mathcal{C}}) \, , \tag{5.11}$$

where $\mathbf{p}_s^{dom} = \langle p^{(d_{\mathcal{C}_1})}, p^{(d_{\mathcal{C}_2})}, \dots \rangle$ is a probability distribution over product domains for segment s and $\mathbf{W}_{\mathcal{C}}$ and $\mathbf{b}_{\mathcal{C}}$ are the classifier's weight and bias parameters. We use the negative log likelihood of the domain prediction as the objective function, combined with the reconstruction loss of Equation (5.6) to obtain a multi-task objective:

$$J_{\text{MT}}(\theta) = J_r(\theta) - \lambda \sum_{s \in \mathcal{C}_{all}} \log p^{(d_s)} \, , \tag{5.12}$$

where λ controls the influence of the classification loss. Note that the negative log-likelihood is summed over all segments in $\mathcal{C}_{all}$, whereas $J_r(\theta)$ is only summed over the in-domain segments $s \in \mathcal{C}_1$. It is important not to use the out-of-domain segments for segment reconstruction, as they will confuse the aspect extractor due to the aspect mismatch between different domains.

5.6 Aspect Extraction Evaluation Data

We created a new dataset for the evaluation of aspect detection models, which is built upon our OPOSUM corpus. We used the six Amazon product domains, namely *Laptop Bags*, *Bluetooth Headsets*, *Boots*, *Keyboards*, *Televisions*, and *Vacuums*. Each domain's full collection, shown in the top section of Figure 5.2, will be used for training.

To evaluate our methods and facilitate research, we produced a human-annotated subset of the corpus. For each domain, we uniformly sampled (across ratings) 10 different products with 10 reviews each, amounting to a total of 600 reviews, to be used only for development (300) and testing (300). We obtained EDU-level aspect annotations as described below. Statistics about the number of products, reviews, and segments are provided in the corresponding section of Table 5.2.

For every domain, we pre-selected nine representative aspects, including the *general* aspect. Using crowd-sourcing platform Figure Eight, we presented the EDU-segmented reviews to three annotators and asked them to select the aspects discussed in each segment. Multiple aspects per segment were allowed. Final labels were obtained using a majority vote among annotators, meaning that if at least two annotators marked a segment with a particular aspect, it was added to the segment's aspect set. The full instructions for the annotation are provided in Appendix A.2. Inter-annotator agreement across domains and annotated segments using Cohen's Kappa coefficient was $K = 0.61$ ($N = 8{,}175$, $k = 3$). Table 5.3 shows the aspects we use in each domain and the number of segments that discuss each one.

5.7 Experiments

We now move on to our experiments, where we compare the original ABAE model against different variants of MATE. We discuss implementation details and present experimental results on our newly created aspect-annotated corpus. We do not compare against non-neural topic models, as He et al. (2017) showed significant improvements with ABAE against such methods.

5.7.1 Implementation Details

Reviews were lemmatized and stop words were removed. We initialized the models' word embedding dictionary using 200-dimensional word embeddings trained on each product domain using skip-gram (Mikolov et al., 2013) with default parameters. We

The OpoSum Corpus

Training Data

	Yelp	IMDb	L. Bags	Bluet/th	Boots	Keyb/s	TVs	Vac/s
Products	–	–	2040	1471	4723	983	1894	1184
Reviews	335K	348K	43K	80K	78K	34K	57K	68K
Sent./Rev	8.9	14.0	5.9	7.5	5.5	7.5	10.7	9.0
EDUs/Rev	19.1	37.4	14.1	18.3	12.7	18.5	26.0	22.0
Words/Rev	128.3	279.2	98.1	122.5	82.6	127.0	180.4	146.6
Classes	5	10	5	5	5	5	5	5

Segment Polarity Evaluation Data

	Yelp	IMDb	L. Bags	Bluet/th	Boots	Keyb/s	TVs	Vac/s
Reviews	100	100	25	25	25	25	25	25
Sentences	1065	1029	162	159	138	161	173	163
EDUs	2100	2398	365	317	301	344	334	357

Aspect Extraction Evaluation Data

	Yelp	IMDb	L. Bags	Bluet/th	Boots	Keyb/s	TVs	Vac/s
Products	–	–	10	10	10	10	10	10
Reviews	–	–	100	100	100	100	100	100
EDUs	–	–	1262	1344	1198	1396	1483	1492

Domains Per Task

	Yelp	IMDb	L. Bags	Bluet/th	Boots	Keyb/s	TVs	Vac/s
Sentim. Classification	✓	✓						
Polarity Ranking	✓	✓	✓	✓	✓	✓	✓	✓
1-Doc Opinion Extr.	✓	✓						
Aspect Extraction			✓	✓	✓	✓	✓	✓

Table 5.2: The OpoSum corpus, showing training and evaluation set statistics, as well as the domains used for the various tasks of this thesis. For aspect extraction, we use all 6 Amazon domains, and annotated 100 reviews from 10 products per domain.

Laptop Bags	**# EDUs**
General	608
Size/Fit	196
Quality	170
Looks	100
Compartments	69
Handles	65
Protection	51
Price	48
Customer Serv.	24

B/T Headsets	**# EDUs**
General	669
Sound	209
Comfort	175
Ease of use	93
Connectivity	81
Durability	65
Battery	58
Price	39
Look	14

Boots	**# EDUs**
General	566
Comfort	234
Size	129
Look	94
Materials	87
Durability	69
Weather Resist.	59
Price	26
Color	25

Keyboards	**# EDUs**
General	773
Feel/Comfort	140
Layout	124
Build Quality	118
Extra function.	79
Connectivity	74
Price	54
Noise	47
Looks	36

Televisions	**# EDUs**
General	803
Image	187
Sound	162
Connectivity	106
Customer Serv.	77
Ease of use	75
Price	63
Apps/Interface	48
Size/Look	45

Vacuums	**# EDUs**
General	803
Accessories	192
Ease of use	165
Suction Power	135
Build Quality	98
Noise	61
Weight	49
Customer Serv.	46
Price	32

Table 5.3: Aspects and their segment frequency in OpoSum's aspect-annotated data.

used 30 seed words per aspect (unless stated otherwise), obtained via Equation (5.10). Word embeddings $\mathbf{L}$ and seed matrices $\{\mathbf{A}_i\}_{i=1}^{K}$ were fixed throughout training. We used the Adam optimizer (Kingma and Ba, 2014) with learning rate 10^{-4} and mini-batch size 50, and trained for 10 epochs. We used 20 negative examples per input for the reconstruction loss and, when used, the multi-tasking coefficient λ was set to 10. Seed words and hyperparameters were selected on the development set and we report results on the test set, averaged over five runs.

5.7.2 Evaluation Metrics

Aspect extraction is essentially a multi-label classification task and, for that reason, it is common practice to evaluate system performance using Micro-F1, i.e., the average of per-class F1 scores, weighted by the class size.

For completeness, we also evaluated our aspect extraction system using two standard clustering measures, *Purity* and *Entropy*, which quantify the quality of the induced clusters. An *aspect cluster* cl_i contains all segments whose predicted aspect category is the same, according to a system. Purity measures the extent to which a cluster cl_i contains segments primarily from one true aspect category a_j, and is summed over all K induced clusters:

$$purity = \frac{1}{K}\sum_j \max_i |cl_i \cap a_j| \,, \tag{5.13}$$

where $|cl_i \cap a_j|$ is the count of segments in cluster cl_i that belong to true aspect a_j. Larger purity values indicate better performance. The *entropy* measure considers how the true aspects are distributed within each predicted cluster of segments. The entropy of a clustering is computed as:

$$entropy = -\sum_i \frac{|cl_i|}{N} \sum_j \frac{|cl_i \cap a_j|}{|cl_i|} \log_2 \frac{|cl_i \cap a_j|}{|cl_i|} \,, \tag{5.14}$$

where N is the total number of segments, and $|cl_i|$ and $|a_j|$ are the sizes of the i-th predicted cluster and j-th true aspect, respectively. Smaller values of entropy indicate better performance.

5.7.3 Results

We trained aspect models on the unlabeled review collections and evaluated their predictions against the aspect-annotated test set the OpoSum corpus (both shown in Table 5.2). We compare three variants of MATE model (with uniform, fixed, and dynamic seed weights) and their multi-task counterparts (MT) against a majority baseline, and ABAE. We used nine aspects for all models. Table 5.4 (top) reports the results using micro-averaged F1. All our model variants significantly outperform ABAE across domains (approximate randomization test, $p < 0.01$; Noreen 1989). MATE$_{uni}$ improves upon the unsupervised model, affirming that informed aspect initialization can facilitate the task. The richer representation of MATE$_{fix}$, however, helps our model achieve a 3.2% increase in F1. Further improvements are gained by the multi-task version of the model, which boosts performance by 2.7%. Interestingly, using dynamic seed weights did not improve performance, resulting in worse F1 scores than the fixed weights variant. We suspect that the softmax-normalized cosine similarities of Equation (5.9) will produce unreliable seed weights in cases were the input segment is not relevant to an aspect, and thus result in less stable training.

When considering the clustering metrics, results are less consistent. Still, all MATE variants perform significantly better than ABAE. Using fixed seed weights still appears to be the best choice, although multi-tasking does not always improve the quality of aspect clusters.

We also investigated how sensitive our models are with respect to important hyperparameters, namely, the number of seed words used and the multi-task coefficient λ. Figure 5.6 shows the performance of four variants of MATE for different seed counts, computed on the human-annotated test instances. We used a variant with uniform seed weights (MATE$_{uni}$) and three variants with fixed weights that have λ set to 0 (no multitasking), 1 and 10, respectively.

The results indicate that using larger numbers of seeds is preferable, as all model variants achieve peak performance when using more than 10 seeds, with the highest average performance achieved with 30 seeds. Additionally, when using multi-tasking, giving more weight ($\lambda = 10$) to the domain classification objective is advantageous.

Finally, Figure 5.7, shows how the performance of the fixed (MATE) and dynamic (MATE$_{dyn}$) variants compares for different numbers of seed words. The graphs indicate that dynamic weights result in inferior performance across cases.

5.8 Summary

In this chapter, we described a weakly supervised framework for the extraction and categorization of aspect expressions in reviews. We presented an overview of previous

(Micro-F1)	L. Bags	B/T	Boots	Keyb/s	TVs	Vac/s	**AVG**
Majority	37.9	39.8	37.1	43.2	41.7	41.6	40.2
ABAE	38.1	37.6	35.2	38.6	39.5	38.1	37.9
MATE$_{uni}$	41.6	48.5	41.2	41.3	45.7	40.6	43.2
MATE$_{fix}$	46.2	52.2	45.6	43.5	48.8	42.3	46.4
MATE$_{dyn}$	45.4	51.5	44.1	41.7	46.5	42.5	45.3
MATE$_{fix}$+MT	**48.6**	**54.5**	**46.4**	**45.3**	51.8	**47.7**	**49.1**
MATE$_{dyn}$+MT	48.4	53.1	44.1	44.9	**53.2**	46.8	48.4

(Purity)	L. Bags	B/T	Boots	Keyb/s	TVs	Vac/s	**AVG**
Majority	45.7	47.7	43.9	53.5	51.3	50.8	48.8
ABAE	48.0	48.5	44.3	54.1	51.5	51.0	49.6
MATE$_{uni}$	53.4	61.1	52.5	56.9	58.6	53.2	56.0
MATE$_{fix}$	56.4	63.0	**54.5**	56.5	60.6	54.7	**57.6**
MATE$_{dyn}$	54.4	61.2	52.4	**57.2**	57.5	55.0	56.3
MATE+MT$_{fix}$	**56.5**	**63.1**	52.6	55.0	60.9	54.9	57.2
MATE+MT$_{dyn}$	55.0	63.0	52.3	**57.2**	**61.6**	**55.7**	57.5

(Entropy)	L. Bags	B/T	Boots	Keyb/s	TVs	Vac/s	**AVG**
Majority	78.1	75.6	78.6	73.4	74.9	74.3	75.8
ABAE	72.3	70.8	75.4	70.0	72.3	72.0	71.2
MATE$_{uni}$	54.5	52.8	58.5	55.2	52.8	57.0	55.1
MATE$_{fix}$	52.4	50.6	**57.0**	**54.2**	50.8	56.3	**53.6**
MATE$_{dyn}$	55.7	54.1	60.7	56.5	55.3	56.6	56.5
MATE+MT$_{fix}$	**52.2**	**50.2**	58.7	56.0	**50.7**	**55.5**	53.9
MATE+MT$_{dyn}$	54.6	50.9	60.2	56.2	50.8	55.9	54.7

Table 5.4: Experimental results for the identification of aspect segments on OpoSum’s aspect-annotated evaluation data, according to Micro-F1, Purity, and Entropy (per domain and on average).

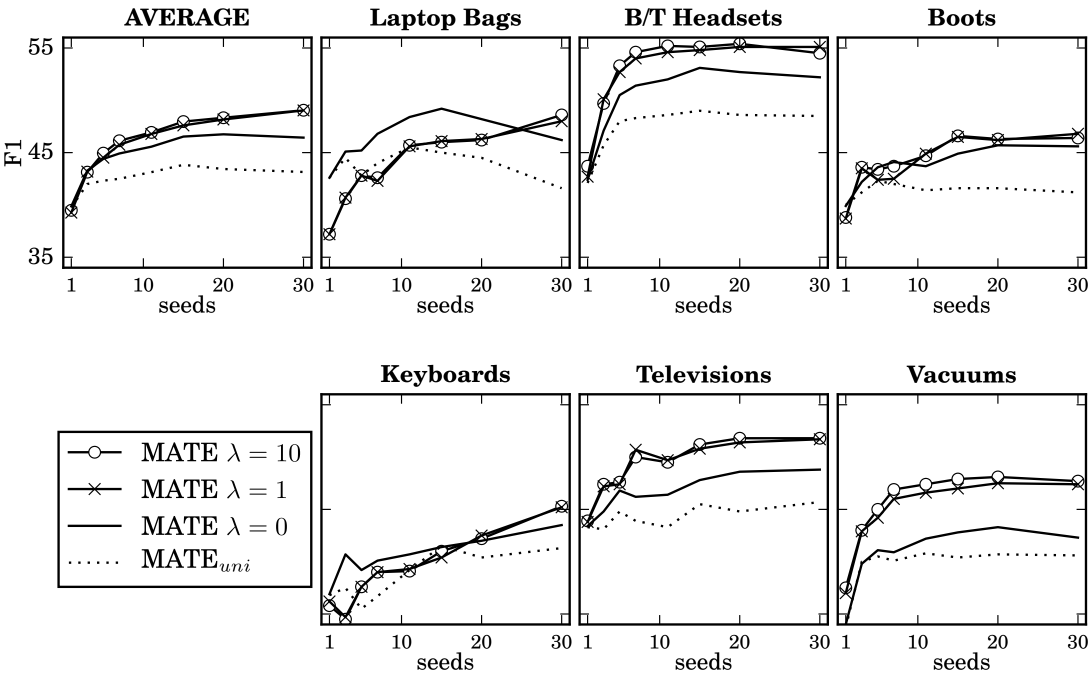


Figure 5.6: The effect of hyperparameters (number of seeds, multi-tasking coefficient) for different variants of MATE.

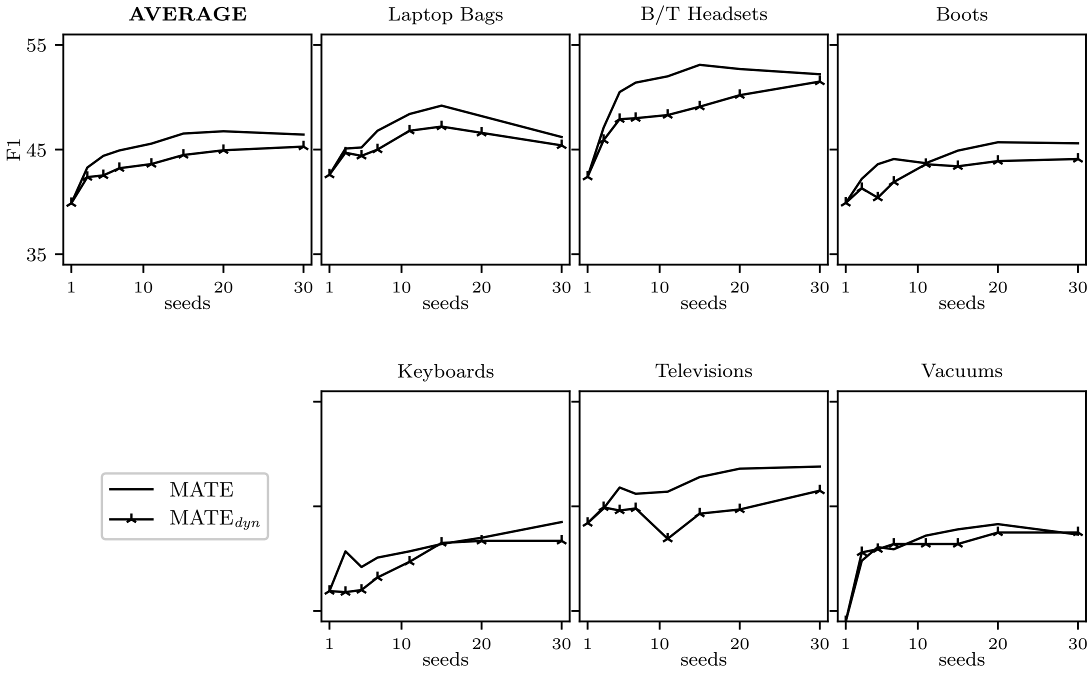


Figure 5.7: The effect of hyperparameters (number of seeds) for the fixed and dynamic variants of MATE.

work on this task, and highlighted two particular formulations; *aspect discovery*, a task similar to topic modeling, where no prior assumptions about potential aspects are made and the discovered topics are mapped to aspects post-hoc; and *aspect extraction*, where domain knowledge is injected into the model and guides the aspect detection process.

With that in mind, we first described the Aspect-Based Autoencoder (He et al., 2017), a neural topic model which has been recently applied with great success to the task of aspect discovery. We then presented a number of extensions and arrived at our Multi-Seed Aspect Extractor, or MATE, that uses two sources of weak supervision: (i) aspect seed words, which are key terms that are characteristic of particular aspects, and (ii) a multi-tasking objective, that takes advantage of the correlations between aspect- and domain-denoting words. We evaluated our models on a new aspect extraction corpus. MATE delivered significant improvements over the original model, and showed consistent patterns of performance across hyperparameter settings.

This chapter concludes the presentation of the weakly supervised neural models of this thesis. Our methodology and experimental evaluation, presented in Chapters 4 and 5, supports the hypothesis that freely available review data and minimal domain knowledge are sufficient to train neural networks which detect fine-grained sentiment and aspects. In the following chapter, we combine our sentiment and aspect detectors, and arrive at a unified framework for the extractive summarization of user opinions from multiple reviews, that requires no direct supervision.

Chapter 6

Extractive Opinion Summarization

In the final chapter of this thesis, we look into the extraction of opinion summaries for a given entity under review. In contrast with the opinion extraction method of Chapter 4, which only utilized polarity scores to detect opinions in *single* reviews, this chapter presents a *multi-review* summarizer that uses two sources of information: sentiment and aspects. In particular, we put to the test our weakly supervised sentiment and aspect detection models and propose a unified extraction framework to identify salient opinions and construct useful summaries. While the majority of related literature has focused on producing structured summaries, which aggregate the attitude of multiple users on particular product aspects, our formulation produces purely textual summaries that highlight aspect-specific comments, while avoiding redundant opinions.

Our summarization framework is based on opinion ranking and brings together sentiment and aspect predictions obtained by our MILNET and MATE models, respectively. We first describe how sentiment polarity or aspect distinctions may be used on their own to rank opinions, and point out why this can lead to the extraction of uninformative comments. We argue that their combination is more likely to promote salient opinions, which we then filter using a greedy algorithm to avoid redundancy.

We further extend our OPOSUM corpus to include EDU-level opinion salience labels and human-curated gold-standard summaries, covering 60 products from six Amazon domains. Automatic evaluation shows that the combination of polarity and aspect information significantly improves the detection of salient opinions. Additionally, when compared with common summarization systems, our method produces extractive summaries that overlap more with gold-standard ones. The effectiveness of our summarizer is further validated by a large-scale user study, where our system was preferred by human judges according to multiple criteria.

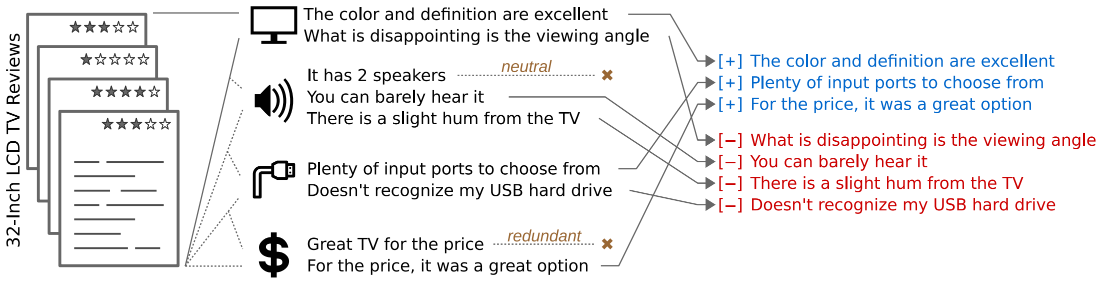

32-Inch LCD TV Reviews
★★★☆☆
★☆☆☆☆
★★★★☆
★★★☆☆
The color and definition are excellent
What is disappointing is the viewing angle
It has 2 speakers
neutral
You can barely hear it
There is a slight hum from the TV
Plenty of input ports to choose from
Doesn't recognize my USB hard drive
Great TV for the price
redundant
For the price, it was a great option
[+] The color and definition are excellent
[+] Plenty of input ports to choose from
[+] For the price, it was a great option
[−] What is disappointing is the viewing angle
[−] You can barely hear it
[−] There is a slight hum from the TV
[−] Doesn't recognize my USB hard drive

Figure 6.1: Aspect-based opinion summarization. Opinions on image quality, sound quality, connectivity, and price of an LCD television are extracted from a set of reviews. Their polarities are then used to group them into positive and negative, while neutral or redundant comments are discarded.

6.1 Introduction

Opinion summarization, i.e., the aggregation of user opinions as expressed in online reviews, blogs, internet forums, or social media, has drawn much attention in recent years due to its potential for various information access applications. For example, consumers have to wade through many product reviews in order to make an informed decision. The ability to summarize these reviews succinctly would allow customers to efficiently absorb large amounts of opinionated text, and manufacturers to keep track of what customers think about their products (Liu, 2012).

The majority of work on opinion summarization is *entity-centric*, aiming to create summaries from text collections that are relevant to a particular entity of interest, e.g., product, person, company, and so on. A popular decomposition of the problem involves three subtasks (Hu and Liu, 2004, 2006): (1) *aspect extraction* which aims to find specific features pertaining to the entity of interest (e.g., battery life, sound quality, ease of use) and identify expressions that discuss them; (2) *sentiment prediction* which determines the sentiment orientation (positive or negative) on the aspects found in the first step, and (3) *summary generation* which presents the identified opinions to the user (see Figure 6.1 for an illustration of the task).

As covered in detail in Chapters 4 and 5, numerous techniques have been proposed for sentiment and aspect detection in user reviews. Various lexicon and rule-based methods (Hu and Liu, 2004; Ku et al., 2006; Blair-Goldensohn et al., 2008) have been adopted for fine-grained sentiment prediction together with a few learning approaches (Lu et al., 2009; Pappas and Popescu-Belis, 2017). For aspects, approaches may involve part of speech tagging (Hu and Liu, 2004), syntactic parsing (Lu et al., 2009), clustering (Mei et al., 2007; Titov and McDonald, 2008b), association rule mining (Ku et al., 2006), and information extraction (Popescu and Etzioni, 2005). As for the summaries, a common format involves a list of aspects and the number of positive and negative opinions for each (Hu and Liu, 2004). While this format gives an overall idea of people's opinion, reading the actual text might be necessary to gain a better understanding of specific details. Textual summaries are created following mostly extractive methods (but see Ganesan et al. 2010 for an abstractive approach), and various formats ranging from lists of words (Popescu and Etzioni, 2005), to phrases (Lu et al., 2009), and sentences (Mei et al., 2007; Blair-Goldensohn et al., 2008; Lerman et al., 2009; Wang and Ling, 2016).

This chapter present a unified framework for opinion extraction from product reviews. We follow the standard architecture for aspect-based summarization, while taking advantage of the success of neural network models in learning continuous features without recourse to preprocessing tools or linguistic annotations. Central to our summarization system is the ability to accurately identify salient, aspect-specific opinions using neural networks (namely, MILNET and MATE) trained on freely available product reviews, using weak supervision signals only, and no gold-standard salience labels or summaries for training. Our system outputs extractive summaries using a greedy algorithm to minimize redundancy.

We expect segments that discuss specific product aspects to be better candidates for useful summaries. We hypothesize that *general* comments mostly describe customers' overall experience, which can also be inferred by their rating, whereas aspect-related comments provide specific reasons for their overall opinion. We also assume that segments conveying highly positive or negative sentiment are more likely to present informative opinions compared to neutral ones, a claim supported by our experiments in Chapter 4. This chapter presents empirical evidence that support these hypotheses.

Our contributions are three-fold: a novel neural framework for the identification and extraction of salient customer opinions that combines aspect and sentiment information and does not require unrealistic amounts of supervision; the introduction of an opinion summarization evaluation corpus which consists of Amazon reviews from six product domains, and includes development and test sets with gold standard salience labels and multi-document extractive summaries; a large-scale user study on the quality of the final summaries paired with automatic evaluations. Our approach outperforms strong baselines in terms of similarity to gold standard summaries. Human evaluation further shows that our summaries are preferred over comparison systems across multiple criteria.

Entity: **Digital Camera**

Aspect: **General**

Positive:	105	<Review sentences>
Negative:	12	<Review sentences>

Aspect: **Picture Quality**

Positive:	95	<Review sentences>
Negative:	10	<Review sentences>

Aspect: **Battery Life**

Positive:	50	<Review sentences>
Negative:	9	<Review sentences>

Figure 6.2: A structured aspect-based opinion summary of a Digital Camera (Liu et al., 2005).

Entity: **Holiday Inn, London**

Aspect: **Food**

The food was excellent, good and delicious. Very good selection of food.

Entity: **Bestwestern Inn, California**

Aspect: **Free Amenities**

Free wine reception in evening.
Free coffee and biscotti and wine.

Figure 6.3: Two abstractive aspect-based opinion summaries of hotels (Ganesan et al., 2010).

6.2 Related Work

Previous work on sentiment analysis and aspect extraction has been covered in Chapters 4 and 5. Here, we focus on the summarization literature, and present selected work from both the opinion mining and generic summarization domains.

6.2.1 Opinion Summarization

The prevalent format for summarizing opinions from a set of reviews involves a list of aspects and a quantitative aggregate of the positive and negative opinions about the entity under review, like the example summary of Figure 6.2 (Hu and Liu, 2004; Liu et al., 2005). The underlying techniques need to identify aspect expressions and detect their sentiment, similarly to our methods, but the system's output differs vastly from our extractive opinion summaries. This non-textual summarization style is useful for large-scale opinion analysis, as it provides a very efficient overview of people's attitude towards a product or a service. However, reading through salient opinions is important for users who want to identify specific advantages or shortcomings and inform their decisions accordingly.

Beineke et al. (2004) were the first to explore the extraction of textual opinion summaries from reviews. They used data from the movie review website Rotten Tomatoes[1], that contained full user reviews paired with single-sentence summaries in the

[1] http://www.rottentomatoes.com

form of verbatim quotations selected by editors of the website. They experimented with a variety of hand-crafted features, which they used to fit Naive Bayes and Logistic Regression models for summary extraction. Carenini et al. (2006) adapted MEAD, a generic multi-document summarizer (Radev et al., 2000), to the task of opinion extraction and compared it against a generation-based abstractive summarizer with mixed results. Ku et al. (2006) proposed an opinion extraction algorithm for news stories and blog posts that identified opinion words, sentences and documents in a bottom-up fashion using retrieval techniques, and predicted their sentiment orientation.

More recently, abstractive opinion summarization has also been explored. The Opinosis summarizer (Ganesan et al., 2010) uses a word graph data structure to represent a set of reviews, and looks for paths that indicate highly redundant text, which may map to popular opinions. Opinosis involves no sentiment or aspect detection mechanism, takes aspect-specific comments as input, and produces concise abstractive summaries, like the ones shown in Figure 6.3. We compare their model to our system in Section 6.6.3. Gerani et al. (2014) introduced an abstractive summarization pipeline that utilized reviews' discourse structure. They proposed an aspect-based adaptation of RST trees (Mann and Thompson, 1988), and a method that aggregated multiple trees into a opinion-representing graph. A template-based language generation component produced the final summaries. To the best of our knowledge, the work of Wang and Ling (2016) is the only example of a neural-based system that tackles multi-document opinion summarization directly. They introduce an encoder-decoder model that requires direct supervision via gold-standard summaries for training, in contrast to our weakly supervised formulation. Once trained, their model was used to produce abstractive summaries from movie reviews.

6.2.2 Multi-Document Summarization

The problem of generic multi-document summarization is commonly approached as a sentence extraction task, and is decomposed into two phases. Firstly, *sentence ranking* deals with the detection of salient sentences using hand-crafted features such as word frequency (Nenkova and Vanderwende, 2005), *tf-idf* scores (Goldstein and Carbonell, 1998), or segment position and length (Radev et al., 2004). Secondly, *sentence selection*, where a small subset of representative candidate sentences are chosen so that redundancy is minimized in the final summaries. Selection strategies include Integer Linear Programming (McDonald, 2007; Gillick and Favre, 2009), graph centrality measures (Erkan and Radev, 2004), and diversity-enforcing optimization (Goldstein and Carbonell, 1998).

A few extractive neural models have been recently applied to generic multi-document summarization. Cao et al. (2015) train a recursive neural network using a ranking objective to identify salient sentences, while follow-up work (Cao et al., 2017) employs a multi-task objective to improve sentence extraction, an idea we used in our aspect extractor. Yasunaga et al. (2017) propose a graph convolution network to represent sentence relations and estimate sentence salience. Contrary to generic summarizers, our summarization framework is tailored to the opinion extraction task, it identifies aspect-specific and salient units, while minimizing the redundancy of the final summary with a greedy selection algorithm (Cao et al., 2015; Yasunaga et al., 2017).

6.3 Definitions

So far, we have focused on specific characteristics of opinionated text, namely their sentiment polarity and aspects. This chapter presents a multi-review opinion extraction system and, for this reason, we provide a more comprehensive set of definitions, together with a formal formulation of the task.

Let C denote a corpus of reviews on a set of products $E_C = \{e_i\}_{i=1}^{|E_C|}$ from a domain d_C, e.g., televisions or keyboards. For every product e, the corpus contains a set of reviews $R_e = \{r_i\}_{i=1}^{|R_e|}$ expressing customers' opinions. Each review r_i is accompanied by the author's overall rating y_i and is split into segments $(s_1, \ldots, s_m)$, where each segment s_j is in turn viewed as a sequence of words $(w_{j1}, \ldots, w_{jn})$. We set segments to be *Elementary Discourse Units* (EDUs; Mann and Thompson 1988) obtained from a *Rhetorical Structure Theory* parser (Feng and Hirst, 2012).

A segment may discuss zero or more *aspects*, i.e., different product attributes. We use $A_C = \{a_i\}_{i=1}^{K}$ to refer to the aspects pertaining to domain d_C. For example, *picture quality*, *sound quality*, and *connectivity* are all aspects of televisions. By convention, a *general* aspect is assigned to segments that do not discuss any specific aspects. Let $A_s \subseteq A_C$ denote the set of aspects mentioned in segment s; $pol_s \in [-1, +1]$ marks the *polarity* a segment conveys, where -1 indicates maximally negative and $+1$ maximally positive sentiment. An opinion is represented by tuple $o_s = (s, A_s, pol_s)$, and $O_e = \{o_s\}_{s \in R_e}$ represents the set of all opinions expressed in R_e. For each product e, our goal is to produce a summary of the most salient opinions expressed in reviews R_e, by selecting a small subset $S_e \subset O_e$.

6.4 Opinion Summarization Framework

We move on to our opinion summarization framework which is based on our polarity prediction model, MILNET, and our aspect extractor, MATE. MILNET is based on Multiple Instance Learning, and learns to predict the sentiment of individual review segments, using document-level labels only for training. MATE uses multiple seed words to target specific aspects of interest, and learns to categorize segments accordingly without direct supervision. Our opinion extraction methodology requires no further training and only uses the outputs of the two models to rank segments based on their salience.

In the following sections, we first reiterate the information obtained from our polarity and aspect prediction models and provide empirical evidence, obtained from the human-annotated portion of the OPOSUM corpus, that polarity and aspects can act as good predictors for opinion salience (Sections 6.4.1 and 6.4.2). Based on these observations, we present three opinion ranking alternatives in Section 6.4.3; one based on polarity only, one based on aspects, and our *salience ranking* method that combines both sources of information. Finally, in Section 6.4.4, we describe a greedy redundancy filtering mechanism as a final step towards producing concise opinion summaries. Throughout these sections, we will refer to Figure 6.6 which provides an illustration of these ideas for television review segments.

6.4.1 Opinion Polarity

MILNET, our weakly supervised sentiment prediction model, is trained on document-level labels only, but learns a segment-level predictor using the principles of Multiple Instance Learning. Once trained, the essential by-product of MILNET are segment-level sentiment predictions $\mathbf{p}_s^{stm} = \langle p_s^{(1)}, \dots, p_s^{(L)} \rangle$ for every input segment s, which are transformed into polarities pol_s, by projecting them onto the $[-1, +1]$ range using a uniformly spaced sentiment class weight vector (for a detailed description, see Section 4.6).

Our expectation is that review segments conveying neutral sentiment (i.e., have polarity scores near zero), are less likely to correspond to useful opinions. We set out to empirically confirm this intuition, using OPOSUM's segment-level salience labels, which we gathered from human annotators as described in detail in Section 6.5. The histogram of Figure 6.4 shows the number of salient and non-salient EDUs in our evaluation data, grouped according to their gold-standard polarity scores. It is evident that near-neutral segments (i.e., those with polarity scores towards zero) are much more likely to be considered non-salient by humans. In contrast, salient segments are distributed more evenly, with a slight tendency towards the positive and negative ends of the polarity spectrum. Our opinion ranking methodology, described in Section 6.4.3, takes these observations into account.

The polarities of nine review segments are shown in the corresponding column of Figure 6.6. Segments range from very positive (e.g., segments 1 and 8), to very negative (e.g., segment 4), while an objective segment (segment 3: *"It has 2 speakers"*) receives a polarity score very close to zero.

6.4.2 Opinion Aspects

MATE is responsible for extracting segments that discuss particular product aspects. After training, it produces an aspect prediction $\mathbf{p}_s^{asp} = \langle p_s^{(a_1)}, \ldots, p_s^{(a_K)} \rangle$ for every input segment s, where $\{a_i\}_{i=1}^K$ are the aspects of a particular domain (see Sections 5.4 and 5.5 for details). A *general* aspect is also included in the aspect-set, and accounts for general comments that do not discuss an identifiable product feature.

We hypothesize that non-general segments, i.e., those discussing specific product aspects, will provide useful information concerning product attributes that influence customer satisfaction. These aspect-specific segments should, therefore, be considered more salient than general ones. We investigated this hypothesis using the gold-standard aspect and salience labels of OPOSUM's human-annotated reviews.[2] Specifically, Figure 6.5 plots the proportion of EDUs that are general and non-general, across all 6 product domains in our corpus. In the case of salient segments (top), approximately 80% of EDUs are non-general. On the contrary, the majority of non-salient EDUs (bottom) are general and should be excluded from opinion summaries.

The aspect column of Figure 6.6 presents aspect predictions for the nine example segments. Aspect probabilities are illustrated as bar charts and, for presentation purposes, we have simplified the television aspect-set to only include 4 categories: *image quality* (segments 1 and 2), *sound quality* (segments 3–5), *connectivity* (segments 6 and 7) and *general* (segments 8 and 9).

For summarization purposes, we are interested in ranking opinions according to their non-generality. The next section describes a technique that transforms aspect probability vectors to real-valued scores, and facilitates opinion ranking.

[2] Again, see section 6.5 for details on our segment salience annotation.

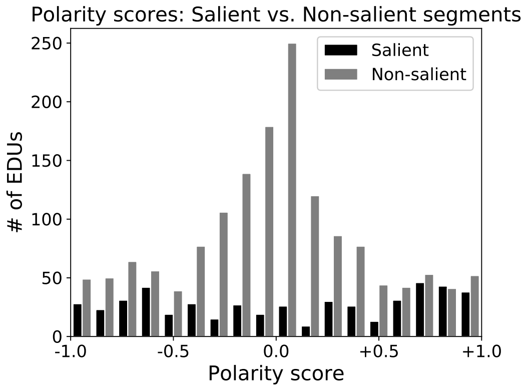


Figure 6.4: Histogram of gold-standard polarities for salient (black bars) and non-salient segments (gray bars) in OpoSum's annotated reviews.

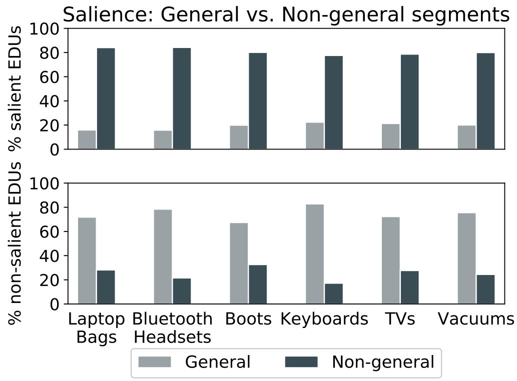
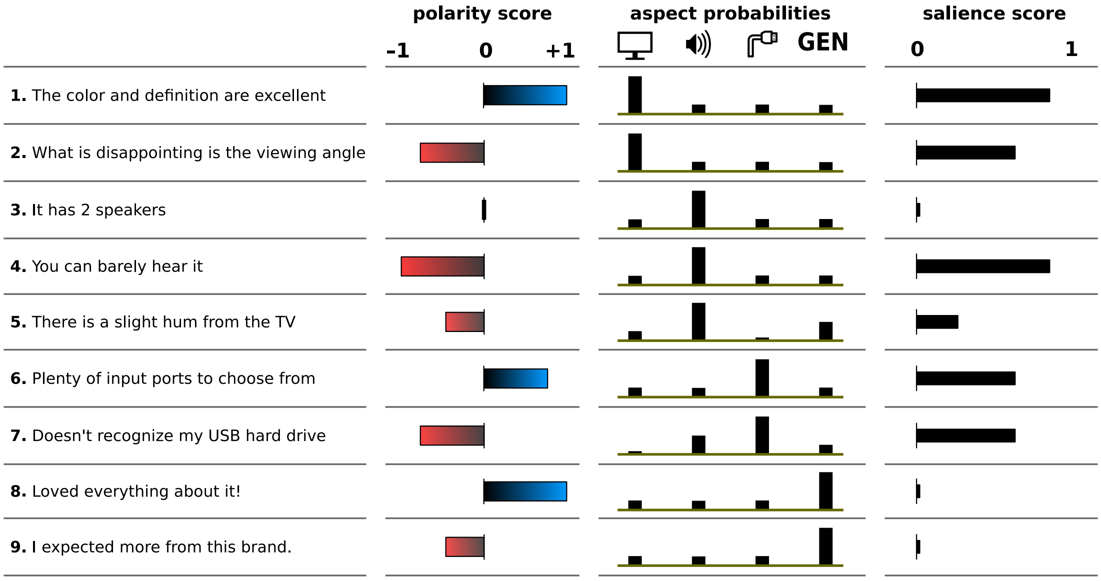


Figure 6.5: Proportion of general (light bars) and non-general (dark bars) segments in OpoSum's annotated reviews (top: salient; bottom: non-salient).



1. The color and definition are excellent

2. What is disappointing is the viewing angle

3. It has 2 speakers

4. You can barely hear it

5. There is a slight hum from the TV

6. Plenty of input ports to choose from

7. Doesn't recognize my USB hard drive

8. Loved everything about it!

9. I expected more from this brand.

Figure 6.6: An illustration of three alternative sources for opinion detection. Polarity scores (shown in the $[-1, +1]$ scale) are good indicators of subjectivity, but may give high scores to general opinions, which tend to be less informative (e.g., segments 8 and 9). Aspect information (shown as probability bar charts) helps identify and discard general comments, but will also promote objective statements (e.g., segment 3). Our salience score (Equation 6.3) combines both sources and alleviates their shortcomings.

6.4.3 Opinion Ranking

Aspect predictions $\mathbf{p}_s^{asp} = \langle p_s^{(a_1)}, \dots, p_s^{(a_K)} \rangle$ and polarities pol_s form the opinion set $O_e = \{(s, A_s, pol_s)\}_{s \in R_e}$ for every product $e \in E_C$. Our goal is to summarize what reviewers are saying about a product, by selecting the most informative opinions from O_e. As discussed in the previous sections, polarity and aspect information are important indicators of opinion salience. On the one hand, polarity scores of higher magnitude (very positive or very negative) are more likely to correspond to highly opinionated expressions. On the other hand, a segment that is predicted with high probability to be non-general, is expected to offer constructive criticism or support for specific aspects of the reviewed item. With these observations in mind, we propose three opinion ranking approaches below:

Polarity-based Ranking: In this case, we rank opinions in O_e according to their absolute polarity score:

$$score_p(o_s) = |pol_s| \, . \tag{6.1}$$

This will promote highly subjective opinions, regardless of their sentiment orientation. Using Figure 6.6 as an example, we observe that segments 1 and 4 (“*The color and definition are excellent*” and “*You can barely hear it*”) would receive the highest scores, but so would segment 8 (“*Loved everything about it!*”) which doesn’t offer any reasons for the reviewer’s satisfaction.

Aspect-based Ranking: When considering aspect information, our objective is to promote segments that are very likely to comment on specific product aspects. We achieve this via the probability difference of the most probable aspect, and the general aspect:

$$score_a(o_s) = \max_i p_s^{(a_i)} - p_s^{(\mathrm{GEN})} \, . \tag{6.2}$$

Preliminary experiments showed that this formula successfully downgrades general segments and produces superior results compared to simply using the probability of the predicted aspect. If correctly identified, general comments like those expressed in segments 8 and 9 of Figure 6.6 (“*Loved everything about it*” and “*I expected more from this brand*”) will receive a score of zero. Segments that are very likely to discuss specific aspects will be scored highly even if they do not communicate an opinion, like segment 3 (“*It has 2 speakers*”).

Salience Ranking: As a means of combining the advantages of polarity and aspect information, we rank every opinion in O_e according to its salience:

$$score_{sal}(o_s) = |pol_s| \cdot (\max_i p_s^{(a_i)} - p_s^{(\mathrm{GEN})}), \tag{6.3}$$

The salience score will be high for opinions that are very positive or very negative and are also likely to discuss a non-general aspect, as seen in Figure 6.6.

6.4.4 Opinion Selection

The final step towards producing summaries is to discard potentially redundant opinions, something that is not taken into account by our ranking methods. Highly ranked segments will hopefully provide useful information, but could still contain paraphrases of the same opinions.

We follow previous work on multi-document summarization (Cao et al., 2015; Yasunaga et al., 2017) and use a greedy algorithm to eliminate redundancy. We start with the highest ranked opinion, and keep adding segments to the final summary one by one, unless the cosine similarity between the candidate segment and any segment already included in the summary is lower than 0.5.

6.5 Opinion Summarization Evaluation Data

We created an evaluation dataset for multi-document opinion summarization models, which further enriches our OpoSum corpus. It contains the same EDU-segmented Amazon reviews used for aspect extraction, originating from our six product domains of choice: *Laptop Bags*, *Bluetooth Headsets*, *Boots*, *Keyboards*, *Televisions*, and *Vacuums*. Table 6.1 presents detailed statistics for the full OpoSum corpus.

As mentioned in the previous chapter, for each domain, we uniformly sampled (across ratings) 10 different products with 10 reviews each, amounting to a total of 600 reviews, to be used only for development (300) and testing (300). We obtained EDU-level salience labels and gold standard opinion summaries for the 60 products in our data using a two-stage procedure, described below. The full instructions for the annotation are provided in Appendices A.3 and A.4.

First, all reviews for a product were shown to three annotators. Each annotator read the reviews one-by-one and selected the subset of segments they thought best captured the most important and useful comments, without taking redundancy into account.

The OpoSum Corpus

Training Data

	Yelp	IMDb	L. Bags	Bluet/th	Boots	Keyb/s	TVs	Vac/s
Products	–	–	2040	1471	4723	983	1894	1184
Reviews	335K	348K	43K	80K	78K	34K	57K	68K
Sent./Rev	8.9	14.0	5.9	7.5	5.5	7.5	10.7	9.0
EDUs/Rev	19.1	37.4	14.1	18.3	12.7	18.5	26.0	22.0
Words/Rev	128.3	279.2	98.1	122.5	82.6	127.0	180.4	146.6
Classes	5	10	5	5	5	5	5	5

Segment Polarity Evaluation Data

	Yelp	IMDb	L. Bags	Bluet/th	Boots	Keyb/s	TVs	Vac/s
Reviews	100	100	25	25	25	25	25	25
Sentences	1065	1029	162	159	138	161	173	163
EDUs	2100	2398	365	317	301	344	334	357

Aspect Extraction Evaluation Data

	Yelp	IMDb	L. Bags	Bluet/th	Boots	Keyb/s	TVs	Vac/s
Products	–	–	10	10	10	10	10	10
Reviews	–	–	100	100	100	100	100	100
EDUs	–	–	1262	1344	1198	1396	1483	1492

Multi-Review Opinion Extraction Evaluation Data

	Yelp	IMDb	L. Bags	Bluet/th	Boots	Keyb/s	TVs	Vac/s
Products	–	–	10	10	10	10	10	10
Reviews	–	–	100	100	100	100	100	100
EDUs	–	–	1262	1344	1198	1396	1483	1492
Ref. Summaries/Prod	–	–	3	3	3	3	3	3

Domains Per Task

	Yelp	IMDb	L. Bags	Bluet/th	Boots	Keyb/s	TVs	Vac/s
Sentim. Classification	✓	✓						
Polarity Ranking	✓	✓	✓	✓	✓	✓	✓	✓
1-Doc Opinion Extr.	✓	✓						
Aspect Extraction			✓	✓	✓	✓	✓	✓
Multi-Doc Op. Extr.			✓	✓	✓	✓	✓	✓

Table 6.1: The full OpoSum corpus, showing training and evaluation set statistics, as well as the domains used for the various tasks of this thesis. For multi-document opinion extraction, we use all six Amazon domains, and obtained salience labels and final extractive summaries for 10 products per domain, using the same reviews as in the aspect extraction tasks.

This phase produced binary *salience* labels against which we can judge the ability of a system to identify important opinions. Using the Kappa coefficient, agreement among annotators was $K = 0.51$ (N = 8,175, k = 3).[3] In the second stage, annotators were shown the salient segments they identified (for every product) and asked to create a final extractive summary by choosing opinions based on their popularity, fluency and clarity, while avoiding redundancy and staying under a budget of 100 words.

6.6 Experiments

In this section, we discuss implementation details and present our experimental setup and results. We evaluate model performance on two subtasks: salient opinion retrieval, and final summary extraction.

6.6.1 Implementation Details

Our opinion retrieval and summarization framework only requires the outputs of our weakly supervised sentiment and aspect detection models. For sentiment, we use the non-gated polarities of MILNET[4] (see Chapter 4). For aspect extraction, we use three variants of our MATE model (see Chapter 5); MATE$_{uni}$, which uses uniform seed word weights; MATE$_{fix}$, which uses fixed weights; and MATE$_{fix}$+MT, which adds a multi-tasking objective. All models were trained using the default parameters described in Section 4.7.2 (for MILNET) and Section 5.7.1 (for MATE).

6.6.2 Salient Opinion Retrieval

We are interested in our system's ability to identify salient opinions in reviews. The first phase of our opinion extraction annotation provides us with binary salience labels, which we use as a gold standard to evaluate system opinion rankings. For every product e, we score each segment $s \in R_e$ using: Equation (6.1) for polarity-based retrieval via MILNET; Equation (6.2) for MATE-based retrieval via MATE's variants; and Equation (6.3) for salience-based retrieval using their combination. We evaluate

[3] While this may seem moderate, Radev et al. (2003) show that inter-annotator agreement for extractive summarization is usually lower ($K < 0.30$).

[4] Gated polarities were advantageous for detecting opinions in single reviews. However, initial experiments showed inconsistent results in the multi-document case, as attention weights are not comparable across reviews.

(MAP)	L.Bags	B/T	Boots	Keyb/s	TVs	Vac/s	**AVG**
MilNet	21.8	19.8	17.0	14.1	14.3	14.6	16.9
MATE$_{uni}$	19.9	27.5	13.8	19.0	16.8	16.1	18.8
MATE$_{fix}$	23.0	30.9	15.4	21.0	18.7	19.9	21.5
MATE$_{fix}$+MT	26.3	37.5	17.3	20.9	23.6	22.4	24.7
MilNet+MATE$_{uni}$	27.1	33.5	19.3	22.4	19.0	20.8	23.7
MilNet+MATE$_{fix}$	28.2	36.0	21.7	24.0	20.8	23.5	25.7
MilNet+MATE$_{fix}$+MT	**32.1**	**40.0**	**23.3**	**24.8**	**23.8**	**26.0**	**28.3**
Human	51.7	53.0	36.6	38.2	33.0	37.1	41.6

(P@5)	L.Bags	B/T	Boots	Keyb/s	TVs	Vac/s	**AVG**
MilNet	40.0	36.7	39.3	28.0	36.0	31.3	35.2
MATE$_{uni}$	48.5	49.7	28.1	44.9	42.4	34.0	41.3
MATE$_{fix}$	57.1	50.7	31.9	43.1	44.7	44.0	45.2
MATE$_{fix}$+MT	60.8	66.7	33.6	44.9	48.0	43.9	49.6
MilNet+MATE$_{uni}$	56.0	66.5	34.8	51.7	43.7	43.5	49.4
MilNet+MATE$_{fix}$	54.7	66.5	39.3	52.0	46.1	49.3	51.3
MilNet+MATE$_{fix}$+MT	**69.2**	**74.7**	**40.4**	**56.4**	**52.8**	**53.1**	**57.8**
Human	76.3	76.7	55.7	64.7	60.0	66.3	66.6

Table 6.2: Experimental results for the retrieval of salient segments on the multi-review opinion extraction portion of OpoSum. Results are presented according to Mean Average Precision (top) and Precision at the 5th retrieved segment (bottom), for six product domains and overall (AVG). Human scores are provided as an upper bound on the achievable performance on the corpus. They correspond to the MAP and P@5 scores of each annotator's gold-standard labels, judged against the two remaining ones, and averaged across annotators.

the obtained rankings via Mean Average Precision (MAP) and Precision at the 5th retrieved segment (P@5).[5] We also computed human scores as an upper bound on the achievable performance in this corpus. They correspond to the MAP and P@5 scores of each annotator's gold-standard labels, judged against the two remaining ones, and averaged across annotators.

Results are shown in Table 6.2. The combined use of polarity and aspect information improves the retrieval of salient opinions across domains, as all model variants that use our salience formula of Equation (6.3) outperform the MILNET- and aspect-only baselines. When comparing between aspect-based alternatives, we observe that retrieval precision correlates with the quality of aspect extraction. In particular, ranking using MILNET+MATE$_{fix}$+MT gives best results, with an average 2.6% increase in MAP against MILNET+MATE$_{fix}$ and 4.6% against MILNET+MATE$_{uni}$. The trend persists when polarities are ignored, but the quality of rankings is worse in this case.

6.6.3 Opinion Summaries

We now turn to the summarization task itself, where we compare our best performing model (MILNET+MATE$_{fix}$+MT), with and without a redundancy filter (RD), against the following methods: a baseline that selects segments *randomly*; a *Lead* baseline that only selects the leading segments from each review; *SumBasic*, a generic frequency-based extractive summarizer (Nenkova and Vanderwende, 2005); *LexRank*, a generic graph-based extractive summarizer (Erkan and Radev, 2004); *Opinosis*, a graph-based abstractive summarizer that is designed for opinion summarization (Ganesan et al., 2010). All extractive methods operate on the EDU level with a 100-word budget. For Opinosis, we tested an aspect-agnostic variant that takes every review segment for a product as input, and a variant that uses the MATE$_{fix}$+MT groupings of segments to produce and concatenate aspect-specific summaries.

To evaluate the systems, we used OPOSUM's human-curated summaries as gold-standard and ROUGE (Lin and Hovy, 2003), a set of metrics for the automatic evaluation of summarization systems against multiple reference summaries. ROUGE variants measure the overlap between system and reference summaries for different subsequence lengths. When more than one reference summaries are available, which is the case for our experiments, individual ROUGE scores are computed per reference summary and are subsequently averaged. We use the unigram (ROUGE-1), bigram

[5] A system's salience ranking is individually compared against labels from each annotator and we report the average.

(ROUGE-1)	L.Bags	B/T	Boots	Keyb/s	TVs	Vac/s	**AVG**
Random	34.5	36.6	37.7	34.1	32.0	35.6	35.1
Lead	36.9	33.3	39.9	35.8	35.2	31.6	35.5
SumBasic	37.1	34.4	34.9	31.6	30.9	35.3	34.0
LexRank	34.9	40.3	39.3	33.9	37.2	40.3	37.7
Opinosis	32.6	39.7	42.1	34.3	34.1	38.0	36.8
Opinosis+MATE$_{fix}$+MT	34.1	42.3	42.7	35.3	36.0	41.7	38.7
MilNet+MATE$_{fix}$+MT	44.0	47.1	43.5	40.9	38.5	47.3	43.5
MilNet+MATE$_{fix}$+MT+RD	**44.7**	**47.8**	**43.9**	**41.7**	**38.6**	**47.6**	**44.1**
Human	54.8	56.8	58.6	48.2	47.2	62.7	54.7

(ROUGE-2)	L.Bags	B/T	Boots	Keyb/s	TVs	Vac/s	**AVG**
Random	10.4	13.0	13.3	10.4	09.5	11.4	11.3
Lead	15.5	12.3	19.1	17.6	15.6	11.3	15.2
SumBasic	14.3	12.2	11.3	08.2	10.3	10.6	11.2
LexRank	12.0	16.3	16.0	09.9	13.9	16.6	14.1
Opinosis	09.4	18.7	21.1	11.0	12.9	12.7	14.3
Opinosis+MATE$_{fix}$+MT	10.7	19.4	20.3	12.1	13.9	18.0	15.8
MilNet+MATE$_{fix}$+MT	23.0	**26.1**	21.1	19.1	**15.9**	25.0	21.7
MilNet+MATE$_{fix}$+MT+RD	**23.3**	25.7	**21.3**	**19.7**	15.6	**25.2**	**21.8**
Human	36.9	40.0	40.4	29.7	27.7	45.2	36.6

(ROUGE-L)	L.Bags	B/T	Boots	Keyb/s	TVs	Vac/s	**AVG**
Random	33.8	35.7	36.9	33.4	31.2	34.9	34.3
Lead	36.2	32.5	39.4	35.1	34.6	30.8	34.8
SumBasic	35.8	33.5	33.2	30.0	29.4	33.6	32.6
LexRank	34.1	39.3	38.5	32.7	35.7	39.0	36.6
Opinosis	31.5	39.0	41.1	33.3	32.7	36.6	35.7
Opinosis+MATE$_{fix}$+MT	32.7	41.2	41.6	34.3	34.4	40.4	37.4
MilNet+MATE$_{fix}$+MT	43.2	46.6	42.6	40.0	37.5	46.8	42.8
MilNet+MATE$_{fix}$+MT+RD	**44.0**	**47.2**	**42.9**	**40.8**	**37.6**	**47.2**	**43.3**
Human	54.0	56.3	57.6	47.3	46.2	61.9	53.9

Table 6.3: Summarization results using OpoSum's gold-standard extractive summaries. We use three variants of ROUGE (Lin and Hovy, 2003). Human scores are provided as an upper bound on the achievable performance on the corpus. They correspond to the ROUGE scores obtained by comparing the reference summaries of an annotator against those of the two remaining ones, averaged across annotators.

	L.Bags	B/T	Boots	Keyb/s	TVs	Vac/s	**AVG**
Polarity							
% positive	57.6	43.1	41.3	61.2	55.1	45.2	50.6
% negative	29.4	34.7	39.8	23.3	32.5	40.8	33.4
% neutral	13.0	22.2	18.9	15.5	12.4	14.0	16.0
avg($\lvert pol_s \rvert$)	0.68	0.72	0.59	0.73	0.65	0.65	0.67
Aspects							
% general	12.6	18.9	18.7	21.0	19.6	22.3	18.9
% non-general	87.4	81.1	81.3	79.0	80.4	77.7	81.1
Disagreement							
$P(opposite\ pol.\lvert same\ asp.)$	0.13	0.20	0.22	0.18	0.27	0.25	0.21

Table 6.4: Quantitative analysis on the content of the summaries generated by our best-performing summarizer (MILNET+MATE$_{fix}$+MT+RD). **Polarity:** We provide (a) the percentage of positive (gold pol$_s > 0.33$), negative (gold pol$_s < -0.33$), and neutral (*otherwise*) segments; (b) the average gold-standard polarity score of the segments included in the generated summaries. **Aspects:** We provide the percentage of general and non-general segments included in the generated summaries, according to gold-standard aspect labels. **Disagreement:** We provide the empirical probability that two summary segments discussing the same aspect, will have opposite polarity (i.e., one positive and one negative). This is computed by considering the gold-standard polarities of all pairs of segments that discuss the same aspect in a single summary.

(ROUGE-2) and *longest common subsequence* (ROUGE-L) F1-style measures. Again, human scores are provided as an upper bound on the achievable performance, and correspond to the ROUGE obtained by comparing the reference summaries of an annotator against those of the two remaining ones, averaged across annotators.

Table 6.3 presents the results for each domain and on average. Our best-performing model (MILNET+MATE$_{fix}$+MT) significantly outperforms all comparison systems ($p < 0.05$; paired bootstrap resampling; Koehn 2004), whilst using a redundancy filter slightly improves performance. Assisting Opinosis with aspect predictions is beneficial, however, it remains significantly inferior to our model for every domain.

Additionally, we computed a series of statistics on the summaries generated by our best performing system, using the gold-standard polarity and aspect labels of segments. Table 6.4 summarizes our findings. With respect to the segments' polarities, we observe that our summarizer does a good job of including mostly positive and negative segments (50.6% and 33.4% of all summary segments, respectively), while disregarding neutral ones (16.0%). This is also reflected by the average absolute gold-standard polarity score of summary segments (0.67). The higher prevalence of positive segments, is likely related to the dataset's bias towards positive review labels (see Figure 3.10), which may in turn influence MILNET's predictions. In terms of aspects, out system does a good job of discarding general comments, as 81.1% of extracted segments discuss specific product attributes.

However, one issue which is not explicitly addressed in our methodology is the existence of disagreeing opinions, i.e., opinions included in a summary which discuss the same aspect, with opposing sentiment. The final row of Table 6.4 shows the likelihood that our summarizer will include a pair of disagreeing opinions in a summary. Approximately 1 out of 5 pairs of summary segments pertaining to the same aspect will not agree in terms of sentiment. This should be considered when presenting the summaries to users.

We also performed a large-scale user study. For every product in the OPOSUM test set, participants were asked to compare summaries produced by: a (randomly selected) human annotator, our best performing model (MILNET+MATE$_{fix}$+MT+RD), Opinosis, and the Lead baseline. The study was conducted on the Figure Eight platform using *Best-Worst Scaling* (BWS; Louviere and Woodworth 1991; Louviere et al. 2015), a less labour-intensive alternative to paired comparisons that has been shown to produce more reliable results than rating scales (Kiritchenko and Mohammad, 2017) and was also used for annotating our sentiment polarity dataset, presented in Chapter 3. We arranged every 4-tuple of competing summaries into four triplets. Every triplet was shown to three crowdworkers, who were asked to decide which summary was *best* and which one was *worst* according to four criteria: *Informativeness* (How much useful information about the product does the summary provide?), *Polarity* (How well does the summary highlight positive and negative opinions?), *Coherence* (How coherent and easy to read is the summary?) *Redundancy* (How successfully does the summary avoid redundant opinions?). The full instructions given to participants are provided in Appendix B.2.

For every criterion, a system's score is computed as the percentage of times it was selected as best minus the percentage of times it was selected as worst (Orme, 2009). The scores range from -100 (unanimously worst) to +100 (unanimously best) and are shown in Table 6.5. Participants favored our model over comparison systems across

	Informativeness	Polarity	Coherence	Redundancy
Gold	2.04	**8.70**	**10.93**	**6.11**
This work	**9.26**	3.15	1.11	2.96
Opinosis	-12.78	-10.00	-9.08	-9.45
Lead	1.48	-1.85	-2.96	0.37

Table 6.5: *Best-Worst Scaling* human evaluation.

all criteria (all differences are statistically significant at $p < 0.05$ using post-hoc HD Tukey tests). Human summaries are generally preferred over our model, however the difference is significant only in terms of coherence ($p < 0.05$).

Finally, Figures 6.7 and 6.8 show example summaries for products from our televisions and boots domains, respectively. The summaries are produced by one of our annotators and by 3 comparison systems (LexRank, Opinosis and our best-performing model MILNET+MATE$_{fix}$+MT+RD). The human summary is primarily focused on aspect-relevant opinions, a characteristic that is also captured to a large extent by our method. There is substantial overlap between extracted segments, although our redundancy filter fails to identify a few highly similar opinions (e.g., those relating to the picture quality in the television summary). The LexRank summary is inferior as it only identifies a few useful opinions, and instead selects many general or non-opinionated comments. Lastly, the abstractive summary of Opinosis does a good job of capturing opinions about specific aspects but lacks in fluency, as it produces grammatical errors.

6.7 Summary

In this chapter, we presented a neural-based framework for textual opinion summarization. Our method combined MATE, our seeded aspect extractor that is trained under a multi-task objective without direct supervision, and MILNET, our multiple instance learning sentiment predictor, to identify and extract useful opinions in reviews. We evaluated our system on a newly created opinion summarization corpus on two subtasks: salient opinion retrieval, and opinion summarization. Our approach delivered significant improvements over baselines in each of the subtasks, while a large-scale judgement elicitation study showed that crowdworkers favor our summarizer over competitive extractive and abstractive systems. The next chapter concludes this thesis by summarizing our main findings, and discusses how these relate to our thesis hypotheses. We identify limitations of our work, and discuss avenues for future work.

Product domain: Televisions

Product name: Sony BRAVIA 46-Inch HDTV

Human	Plenty of ports and settings. Easy hookups to audio and satellite sources. The sound is good and strong. This TV looks very good. and the price is even better. The on-screen menu/options is quite nice. and the internet apps work as expected. The picture is clear and sharp. which is TOO SLOW to stream HD video... The software and apps built into this TV. are difficult to use and setup. Their service is handled off shore making. communication a bit difficult. :(
LexRank	Get a Roku or Netflix box. I watch cable, Netflix, Hulu Plus, YouTube videos and computer movie files on it. Sound is good much better. DO NOT BUY! this SONY Bravia ‘ Smart ’ TV... and avoid the Sony apps at all costs. Because of these two issues, I returned the Sony TV. Also you can change the display and sound settings on each port. However, the streaming speed for netflix is just down right terrible. Most of the time I just quit. Since I do not own the cable box, So, I have the cable.
Opinosis	The picture and not bright at all even compared to my 6-year old sony lcd tv. It will not work with an hdmi. Connection because of a conflict with comcast’s dhcp. Being generous because I usuallly like the design and attention to detail of sony products). I am very disappointed with this tv for two reasons: picture brightness and channel menu. Numbers of options available in the on-line area of the tv are numerous and extremely useful. Wow look at the color, look at the sharpness of the picture, amazing and the amazing.
This work	Plenty of ports and settings and have been extremely happy with it. The sound is good and strong. The picture is beautiful. And the internet apps work as expected. And the price is even better. Unbelieveable picture and the setup is so easy. Wow look at the color, look at the sharpness of the picture. The Yahoo! widgets do not work. And avoid the Sony apps at all costs. Communication a bit difficult. :(

Figure 6.7: Human and system summaries for a product in the *Televisions* domain.

Product domain: Boots

Product name: Minnetonka Women's Double Fringe Side Zip Boot

Human	They are awesome looking. Well made and just something different to wear. The colour is beautiful. the attention to detail is great! So well made so very comfortable! They do not fit well, are horribly uncomfortable. and make a squeaky noise no matter what type of floor you are walking on. The seam on the top of the foot sits too low and rubs against the top of the foot. These were very narrow. They are too big even with heavy socks,
LexRank	I used to be a big fan of MInnetonka. They are too big even with heavy socks, but sent them back because too big. and I do love them. I really love, Great buy - love Minnetonka! these boots are very nice, well made, the bottom tread is cut to this narrow width. They are a comfortable fit on width, but a little long in length. the sizing issue doesn't give me any problems. Tried multiple sizes and they all had the same seam issue. even though they are too big though - but they are still too big,
Opinosis	the top of the foot sits too low and rubs against the top of the foot. i didn't want to send back and wait more time to get them again. and make a squeaky noise no matter what type of floor you are walking on. they are a comfortable fit on width and too big even with heavy socks. a beautiful moccasin but a shame they didn't seem true to size. and rested some heavy books on top of them for a few days. the fringe was all crazy and going in every which direction.
This work	They are awesome looking. I'd recommend these any day. Love the style and taupe color is perfect. I love wearing them with skinny jeans or with tights and a dress. the attention to detail is great! So well made so very comfortable! I love the colour plum. as other reviewers have pointed out : Narrow. They have stretched a little, but maybe get a size or half size up. They do not fit well, are horribly uncomfortable. Not worth it to me. but not good for an every day use. dont waste your money on these,

Figure 6.8: Human and system summaries for a product in the *Boots* domain.

Chapter 7

Conclusions and Future Work

7.1 Conclusions

The analysis and summarization of opinions expressed in online user reviews, has become an important Natural Language Processing problem, with applications that benefit customers and service providers alike. The proliferation of reviews on the Web has increased the utility of systems that aggregate user attitudes and, at the same time, provides incredibly rich data for the development of machine learning methods. This thesis tackled the extraction of textual opinion summaries without recourse to expensive annotations or gold-standard data. Central to our approach were two core features of opinionated expression: the *aspects* discussed and the *sentiment polarity* conveyed towards them. Three main research questions motivated our work:

1. *Can we utilize freely available information, e.g., in the form of user ratings and product domain labels, to train weakly supervised neural networks that detect fine-grained sentiment and aspects in reviews?*

A significant portion of this thesis was devoted to gathering data and evaluating novel methods with regards to this question. Our first contribution was the construction of a large-scale training and evaluation corpus, OPOSUM (shorthand for **Op**inio**n** **Sum**marization). OPOSUM contains more than one million user reviews from eight diverse domains. Every review is accompanied by its user rating, but comes with no fine-grained sentiment or aspect annotations, which aligns with our overarching goal of using no direct supervision for training our neural models. In order to evaluate our methods, we set aside a few hundred reviews, and used human annotators to manually label them with sentiment and aspect distinctions. We obtained sentiment labels on the sentence and clause level, and aspect labels on the clause level.

We took a first step towards understanding how sentiment is expressed in reviews by analyzing the distribution of positive, neutral, and negative comments within them and their constituent sentences. Our exploratory analysis, presented in Chapter 3, showed that 45% of reviews contain opinions of mixed polarity. Moreover, we observed that more than a third of multi-clause review sentences communicate mixed opinions. These findings posed strong indication that (a) methods that attempt to detect fine-grained sentiment must handle cases where mixed opinions coexist within a single review, and (b) when the goal is to extract coherent opinions, methods need to examine reviews at a subsentence granularity.

With this in mind, we presented our weakly supervised neural models for segment-level sentiment prediction in Chapter 4. Our first model, HIERNET, is an attention-based hierarchical network that composes segment vectors into a single review representation, used to predict its overall sentiment. The model uses segments' individual attention weights to differentiate between opinions and neutral statements, in a process called *gating*. Our second sentiment model, MILNET, overcomes HIERNET's inability to identify segments of opposing polarity by first predicting the sentiment of individual segments, and then combining those prediction via attention-weighted averaging. MILNET naturally produces independent segment predictions, which are again fine-tuned via gating. Both models are trained on review-level labels only. Our experiments showed that MILNET outperformed HIERNET on a series of sentiment detection tasks, and performed comparably well to a fully supervised and a state-of-the-art lexicon-based model. This was a significant result and an indication that weak supervision can produce accurate sentiment predictors, without human annotations.

In Chapter 5, we set out to evaluate the weak supervision hypothesis on the problem of aspect detection, which we formulated as a *guided* extraction task. We described an autoencoder architecture (He et al., 2017), that was previously used for discovering aspects, without any preconception about the product domain at hand. Using that as a starting point, we presented our Multi-Seed Aspect Extractor, or MATE, which uses aspect keywords to initialize descriptors of the targeted aspects, and is subsequently trained without supervision. We also proposed a multi-tasking extension to improve the model's ability to identify aspect-signaling words. Experimental results showed that our seeded aspect representations and our multi-tasking objective significantly improved the autoencoder's ability to detect and categorize aspect-specific opinions.

2. *Can we use sentiment and aspect predictions as evidence for identifying salient opinions, and extracting textual summaries?*

To answer this question, we first extended the human-annotated portion of our OpoSum dataset to include opinion salience labels and reference summaries for 60 products across six domains. Once again, the gold-standard data were only used for testing our methods. We employed our sentiment predictor, MilNet, and our aspect extractor, MATE, for the task of multi-review opinion summarization. In accordance with previous work on structured opinion summarization, we hypothesized that aspect-specific, positive or negative expressions are more likely to correspond to salient opinions. We devised a formula for combining the two sources of information and obtained *salience* scores for all review segments about a product. Salience rankings were used to produce extractive opinion summaries, while a greedy redundancy filter ensured that repeated opinions were discarded. Automatic evaluation against OpoSum's reference summaries showed that our summarizer performed significantly better than common summarization baselines. Furthermore, participants in a large-scale judgement elicitation study preferred our summaries across multiple criteria.

3. *Given the extractive nature of our summarizers, what should be the preferred unit of extraction?*

Chapter 4 presented a human evaluation study that, among others, focused on the granularity of extracted segments. In particular, we compared single-review summaries produced by our best performing system, using sentence- and clause-based extraction. A significant majority of participants indicated that the clause-based summaries were more informative and did a better job at highlighting positive and negative opinions. However, clause-based summarization lacked in fluency, as it tends to produce more terse and telegraphic text, which may seem unnatural or include segmentation errors.

7.2 Future Work

Possible directions of future work are many and varied. Below, we point out some limitations of our methods and discuss how these may lead to new research.

Joint Modeling of Sentiment and Aspects In this thesis, we explored the synergy between sentiment and aspects for identifying salient opinions, but restricted our methodology to model each one independently. Previous work has hinted at the benefits of jointly modeling both features of subjective text (Titov and McDonald, 2008a; Brody and Elhadad, 2010; Zhao et al., 2010; Lazaridou et al., 2013; Diao et al., 2014). However, researchers have only recently approached this idea in a neural setting, and have done so using human-annotated data for training (Schmitt et al., 2018). Combining findings of this thesis in support of weakly supervised neural networks with the lessons learned from previous work on joint aspect and sentiment modeling may lead to exciting new research avenues. One potential direction is to use aspect-specific ratings that accompany many reviewing interfaces as a training signal to detect fine-grained sentiment and aspect simultaneously. This may be formulated under a Multiple Instance Learning framework, similarly to the non-neural approach of Pappas and Popescu-Belis (2017). When not relying on aspect-specific ratings, a joint modeling approach may still benefit from using shared representations for aspects and sentiment, under a combination of weakly supervised (e.g., Multiple Instance Learning) and unsupervised (e.g., autoencoder) objectives. Methods should aim to take advantage of the synergy between aspect- and sentiment-signaling terms, as attempted previously by non-neural approaches (Zhao et al., 2010; Brody and Elhadad, 2010; Lazaridou et al., 2013).

Abstractive Opinion Summarization Neural language modeling and generation has received significant attention, with applications that include machine translation (Bahdanau et al., 2015; Sennrich et al., 2016), document summarization (Cheng and Lapata, 2016; Narayan et al., 2018), data-to-text generation (Perez-Beltrachini and Lapata, 2018) and sentence simplification (Zhang and Lapata, 2017). This thesis presented an extractive approach for the summarization of opinions but ignored the generation of abstractive summaries, an idea previously explored using graph-based (Ganesan et al., 2010; Gerani et al., 2014) or fully supervised neural models (Wang and Ling, 2016). As with many extractive approaches, our methods may often produce incoherent summaries which lack in fluency. This is more pronounced in the case of clause-based extraction, as indicated by the human evaluation of Chapter 4. An abstractive opinion summarizer, on the other hand, may use extracted salient opinions to produce well formed summaries. One approach would be to use a template-based method, similar to work by Gerani et al. (2014). More interestingly, learning-based methods that do not rely on human-curated summaries for training may be used. A weakly supervised abstractive summarizer needs to model opinion-style language generation through indirect means. For example, the abstraction procedure can be decomposed into two phases. Firstly, aspect-specific, positive or negative opinions are identified using a method similar to those described in this thesis. Then, a neural-based generation model may first encode the context of an salient opinion (e.g., the whole review), and then try to generate the opinion itself, while conditioning on the context and its specific aspect and polarity. Similar encoder-decoder architectures have been used for language modeling and machine translation with great success (Cho et al., 2014).

Latent Document Structure Previous work on non-neural sentiment analysis (Bhatia et al., 2015) and summarization (Li et al., 2016) has indicated that document structure information, obtained from Rhetorical Structure Theory (Mann and Thompson, 1988) can improve performance. More recently, work by Liu and Lapata (2018) showed that neural networks can uncover latent document structure in review text, using a carefully crafted attention mechanism. Whether predicted explicitly using a discourse parser, or induced as a latent hierarchy, document structure may help identify primary opinions (e.g., those with multiple dependant expressions), or produce more fluent summaries (e.g., by favoring the extraction of coherent segment combinations). Future work that overcomes the reliance on a discourse parser would be very significant; RST parsing (Mann and Thompson, 1988; Feng and Hirst, 2012) is a computational bottleneck of our methods, and a source of segmentation errors.

Additional Domains / Languages Although extensive, our evaluation throughout this thesis focused on a handful of product or service domains, and only dealt with English reviews. Multilingual review corpora, like the newly-created *Amazon Customer Review Dataset*[1] which includes product reviews from hundreds of domains across five languages, should inspire new research on this direction. Since discourse parsing methods are available only for a handful of languages, methods that induce latent structure, as discussed above, might be necessary.

[1] https://s3.amazonaws.com/amazon-reviews-pds/readme.html

Appendix A

Instructions for OPOSUM's Annotation

A.1 Sentiment Polarity Annotation

We provide the sentiment polarity annotation instructions given to participants of the Figure Eight crowd-sourcing effort described in Section 3.2.2.

I. General instructions:

- Attempt HITs only if you are a **native speaker of English**.
- We are happy to **receive feedback** and improve this job accordingly. We regularly go over all contested and high miss % test questions and fix potential errors.
- Your responses are **confidential**. Any publications based on these responses will not include your specific responses, but rather aggregate information from many individuals. We will not ask any information that can be used to identify who you are.

II. Sentiment Annotation Instructions

Reviews (e.g., for products, restaurants or movies) often contain sentences, phrases and words associated with degrees of positive or negative sentiment. For example, the phrase *"Food was outstanding"* conveys more positive sentiment than *"We ate most of it"*. Similarly, the sentence *"I won't be going back there!"* is more negative than *"I'd never seen this shop before"*.

The task:

In this task, you are given **4 sentences** taken from different online reviews. Your job is to compare the sentiment of the **highlighted** part of each sentence. In many cases, the whole sentence will be highlighted. **When just part of a sentence is highlighted, the rest is only shown to provide context.** You must **only** judge the sentiment of the **highlighted part**. You can also select to view the full review, if you need to.

We ask you to answer 2 questions:

Q1. Which highlighted segment is **the most positive**?

Q2. Which highlighted segment is **the most negative**?

In some cases, sentiment is expressed explicitly: *"we tried some excellent starters"*, *"loved the decoration"*, *"this was the worst!"*. In other cases, the sentiment is more implicit: *"When we questioned the high parking rate the attendant laughed at us."*

If no text segment conveys a positive sentiment, then the response to Q1 should be the **least negative** choice. Similarly, if no negative segments exist, the response to Q2 should be the **least positive** choice.

Example question 1:

1. We complained about the wait and **we got a half-hearted apology.**
2. **Take my advise**, avoid them.
3. He said the hot dog was good, **and fries were fine.**
4. **Each time seems to be better than the last.**

Most positive segment: **4**

Most negative segment: **1**

3 is positive, but less so than **4**.

2 is neutral, although the sentence as a whole is negative.

Example question 2:

1. They bring you some olives to snack on **as you wait for your food.**
2. **The prices are much more reasonable** than other theaters.
3. **I tried the broccoli cheese soup** and it wasn't great.
4. The hostess **others seem to fear and loathe** was very pleasant.

Most positive segment: **2**

Most negative segment: **4**

Under a different context, **1** could convey negative sentiment (wait is usually bad), but in this case it does not.

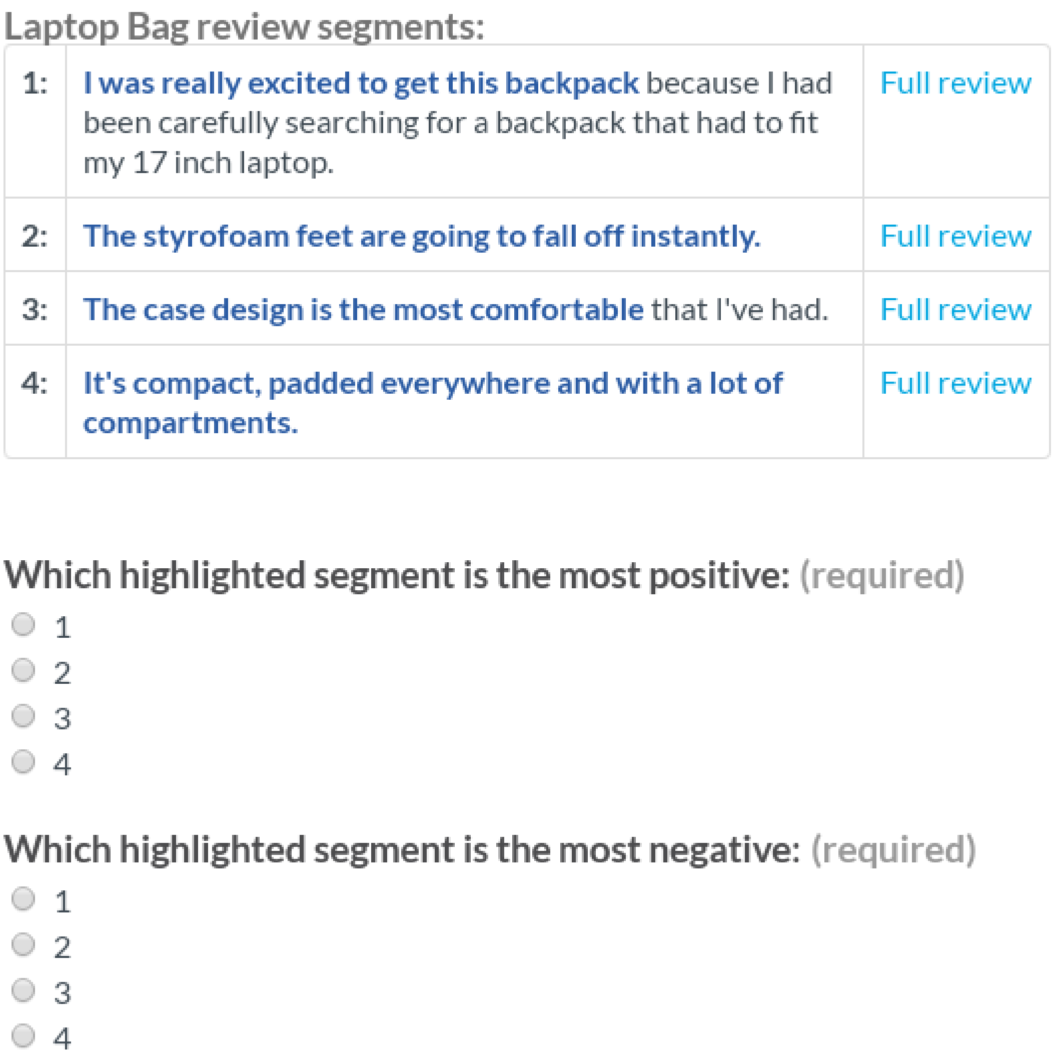
3 is neutral, although the sentence as a whole is negative.

4 is negative, although the sentence as a whole is positive.

Important Notes:

- Provide an answer that most speakers of English would agree with.
- The answer to Q2 is always different from the answer to Q1.
- In rare cases, the ”Full Review” link might fail to work. Refreshing the page should fix this.
- If you do not know the meaning of a word, please either skip the page or look up the meaning in your favorite dictionary:
  - `http://www.merriam-webster.com/`
  - `http://www.google.com/`

Interface Screenshot:



Laptop Bag review segments:

1:	**I was really excited to get this backpack** because I had been carefully searching for a backpack that had to fit my 17 inch laptop.	Full review
2:	**The styrofoam feet are going to fall off instantly.**	Full review
3:	**The case design is the most comfortable** that I've had.	Full review
4:	**It's compact, padded everywhere and with a lot of compartments.**	Full review

Which highlighted segment is the most positive: (required)

○ 1
○ 2
○ 3
○ 4

Which highlighted segment is the most negative: (required)

○ 1
○ 2
○ 3
○ 4

A.2 Aspect Annotation

We provide the aspect annotation instructions given to participants of the Figure Eight crowd-sourcing effort described in Section 5.6.

I. General instructions:

- Attempt HITs only if you are a **native speaker of English**.
- We are happy to **receive feedback** and improve this job accordingly. We regularly go over all contested and high miss % test questions and fix potential errors.
- Your responses are **confidential**. Any publications based on these responses will not include your specific responses, but rather aggregate information from many individuals. We will not ask any information that can be used to identify who you are.

II. Product Aspect Annotation in Customer Reviews

Product reviews contain comments about the overall experience of the author, as well as opinions on particular aspects of the product/service. For example, the following review snippets don't discuss any particular product aspects:

[None]	This was great!
[None]	This was recommended to me by my personal trainer.
[None]	I wasn't impressed at all.
[None]	I guess I'm stuck with this TV now.

In contrast, the following snippets comment on one or more specific aspects [shown in brackets] of the corresponding products:

[Battery]	The damn thing won't charge.
[Price]	I paid just under $25.00 for it.
[Image Quality + Accessories]	The picture is a bit blurry and the remote control won't work.

The task:

In this task, you are asked to annotate different parts of product reviews with their corresponding aspects. On the top of each page, you can find the **product name**, the **product type**, and a **list of aspects** (with descriptions) that are relevant to the particular product type. Then, you are given **5 reviews** for that product. Each review is shown as a whole and also split into segments (sentences or phrases).

We ask you to select the aspects that are discussed in each segment, using the provided checkboxes. All segments are assigned to ”None” by default. You should:

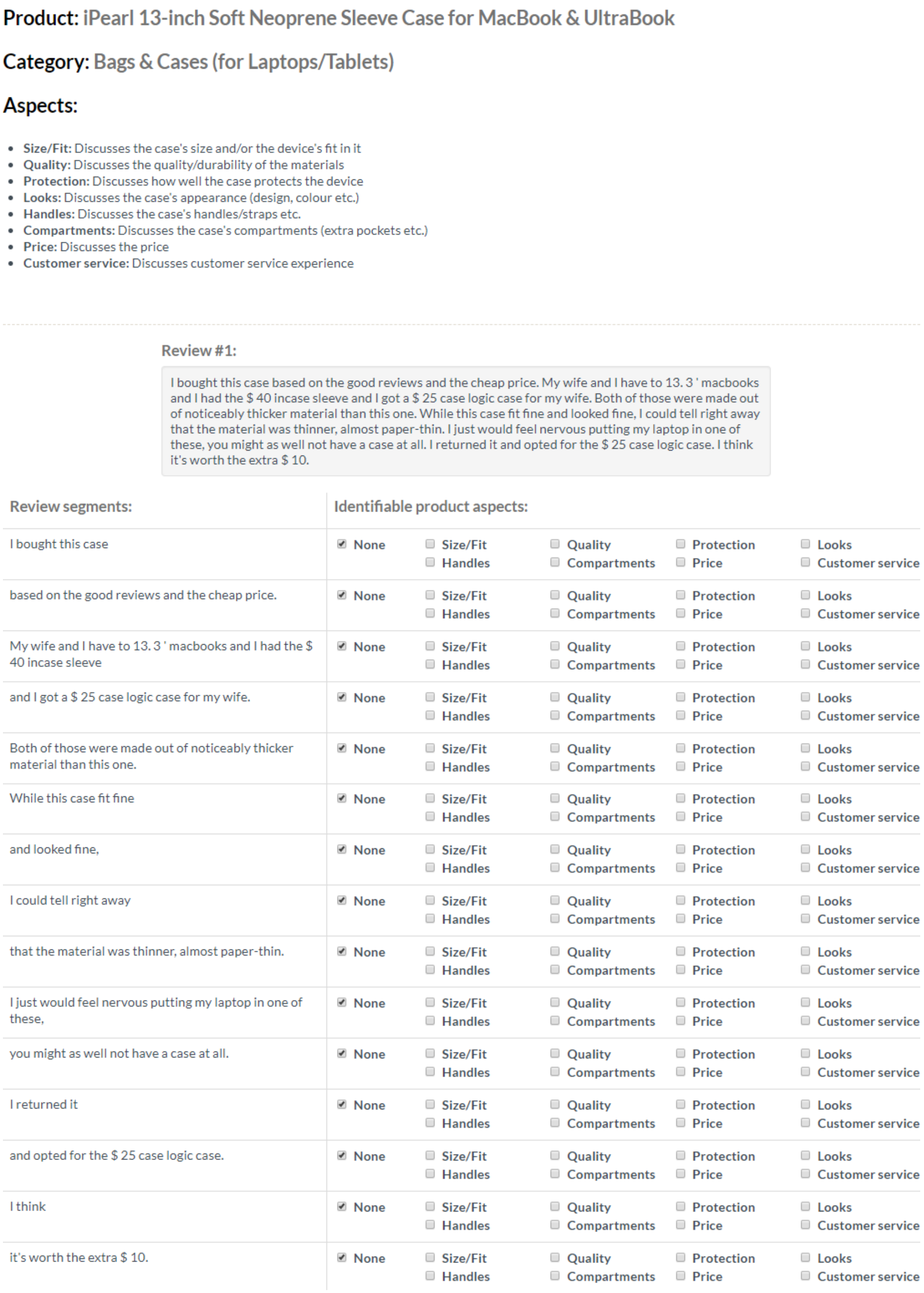
- Use checkbox **“None”** if a segment does not discuss an identifiable product aspect, or if none of the provided aspects is a good enough fit.
- Use **one or multiple** checkboxes (other than ”None”) for segments discussing any of the provided aspects.

Important note: In many cases, a segment will discuss a particular aspect without explicitly mentioning it. This happens often when a segment is part of a larger coherent sequence (e.g. sentence). You should use the segment's context (i.e. surrounding segments) to help you select the appropriate aspects. For example, notice how the following 3 segments of a bluetooth headset device should ideally be annotated:

[None]	After having this headset for just over a year,
[Comfort]	I can say it's very light and hardly feels like
[Comfort]	you 're wearing it.

The 1st segment is not directly related to the aspect discussed, so it's labeled as ”None”. The 2nd and 3rd segments convey the user's opinion regarding how comfortable the headset is to wear. They are both annotated accordingly, even though the 3rd segment on its own does not explicitly mention anything about the device's comfort level.

Interface Screenshot:



Product: iPearl 13-inch Soft Neoprene Sleeve Case for MacBook & UltraBook

Category: Bags & Cases (for Laptops/Tablets)

Aspects:

- **Size/Fit:** Discusses the case's size and/or the device's fit in it
- **Quality:** Discusses the quality/durability of the materials
- **Protection:** Discusses how well the case protects the device
- **Looks:** Discusses the case's appearance (design, colour etc.)
- **Handles:** Discusses the case's handles/straps etc.
- **Compartments:** Discusses the case's compartments (extra pockets etc.)
- **Price:** Discusses the price
- **Customer service:** Discusses customer service experience

Review #1:

I bought this case based on the good reviews and the cheap price. My wife and I have to 13. 3 ' macbooks and I had the $ 40 incase sleeve and I got a $ 25 case logic case for my wife. Both of those were made out of noticeably thicker material than this one. While this case fit fine and looked fine, I could tell right away that the material was thinner, almost paper-thin. I just would feel nervous putting my laptop in one of these, you might as well not have a case at all. I returned it and opted for the $ 25 case logic case. I think it's worth the extra $ 10.

Review segments:	Identifiable product aspects:				
I bought this case	☑ None	□ Size/Fit □ Handles	□ Quality □ Compartments	□ Protection □ Price	□ Looks □ Customer service
based on the good reviews and the cheap price.	☑ None	□ Size/Fit □ Handles	□ Quality □ Compartments	□ Protection □ Price	□ Looks □ Customer service
My wife and I have to 13. 3 ' macbooks and I had the $ 40 incase sleeve	☑ None	□ Size/Fit □ Handles	□ Quality □ Compartments	□ Protection □ Price	□ Looks □ Customer service
and I got a $ 25 case logic case for my wife.	☑ None	□ Size/Fit □ Handles	□ Quality □ Compartments	□ Protection □ Price	□ Looks □ Customer service
Both of those were made out of noticeably thicker material than this one.	☑ None	□ Size/Fit □ Handles	□ Quality □ Compartments	□ Protection □ Price	□ Looks □ Customer service
While this case fit fine	☑ None	□ Size/Fit □ Handles	□ Quality □ Compartments	□ Protection □ Price	□ Looks □ Customer service
and looked fine,	☑ None	□ Size/Fit □ Handles	□ Quality □ Compartments	□ Protection □ Price	□ Looks □ Customer service
I could tell right away	☑ None	□ Size/Fit □ Handles	□ Quality □ Compartments	□ Protection □ Price	□ Looks □ Customer service
that the material was thinner, almost paper-thin.	☑ None	□ Size/Fit □ Handles	□ Quality □ Compartments	□ Protection □ Price	□ Looks □ Customer service
I just would feel nervous putting my laptop in one of these,	☑ None	□ Size/Fit □ Handles	□ Quality □ Compartments	□ Protection □ Price	□ Looks □ Customer service
you might as well not have a case at all.	☑ None	□ Size/Fit □ Handles	□ Quality □ Compartments	□ Protection □ Price	□ Looks □ Customer service
I returned it	☑ None	□ Size/Fit □ Handles	□ Quality □ Compartments	□ Protection □ Price	□ Looks □ Customer service
and opted for the $ 25 case logic case.	☑ None	□ Size/Fit □ Handles	□ Quality □ Compartments	□ Protection □ Price	□ Looks □ Customer service
I think	☑ None	□ Size/Fit □ Handles	□ Quality □ Compartments	□ Protection □ Price	□ Looks □ Customer service
it's worth the extra $ 10.	☑ None	□ Size/Fit □ Handles	□ Quality □ Compartments	□ Protection □ Price	□ Looks □ Customer service

A.3 Opinion Extraction: Phase 1 (Salience Labels)

We provide the salient opinion extraction instructions given to participants of the Figure Eight crowd-sourcing effort described in Section 6.5.

I. General instructions:

- Attempt HITs only if you are a **native speaker of English**.
- We are happy to **receive feedback** and improve this job accordingly. We regularly go over all contested and high miss % test questions and fix potential errors.
- Your responses are **confidential**. Any publications based on these responses will not include your specific responses, but rather aggregate information from many individuals. We will not ask any information that can be used to identify who you are.

II. Summarizing Product Reviews

The purpose of this annotation task is to create short summaries of the most important opinions expressed in a set of reviews for a product. In particular, we want to produce a 100-word opinion summary for every product, by selecting the most representative, informative and fluent segments found in 10 customer reviews. You will not be asked to write your own summary – you will simply select segments found in the original reviews.

Going directly from the 10 original reviews to a 100-word summary is a challenging task. For this reason, we have split the summarization task into 2 phases:

- **Phase 1:** Opinion extraction: the annotator selects the most important opinions expressed in each review independently.
- **Phase 2:** Final Summaries: the annotator produces a final summary, by selecting the most appropriate segments from those extracted in Phase 1. Below, we describe Phase 1. You need to finish Phase 1 to move on to Phase 2.

III. Phase 1: Opinion Extraction

For this task, you will be given **10 reviews for a particular product**. For each review, you should select the segments that you believe best capture the most **important**

and useful comments of the reviewer. You should judge the contents of each review independently – don't worry if similar, identical or opposite opinions are selected in multiple reviews (you will take care of this in Phase 2). There are no lower or upper limits to the number of segments you select per review.

Judging whether a particular segment is worth selecting is, to some extent, subjective and **you** should select the segments that you would want to see in a summary of opinions for the particular product. However, some basic rules should be taken into account:

- Only select segments that express **primary opinions** of the reviewer and not elaborations or further details about his experience. For example, in the following review excerpt, only the segments marked with a [✓] should be selected:

[✓]	These boots fit perfectly	[primary opinion]
[✓]	and they are VERY comfortable.	[primary opinion]
[ ]	I can wear them all day at work	[elaboration]
[ ]	and my feet do not hurt at all.	[elaboration]

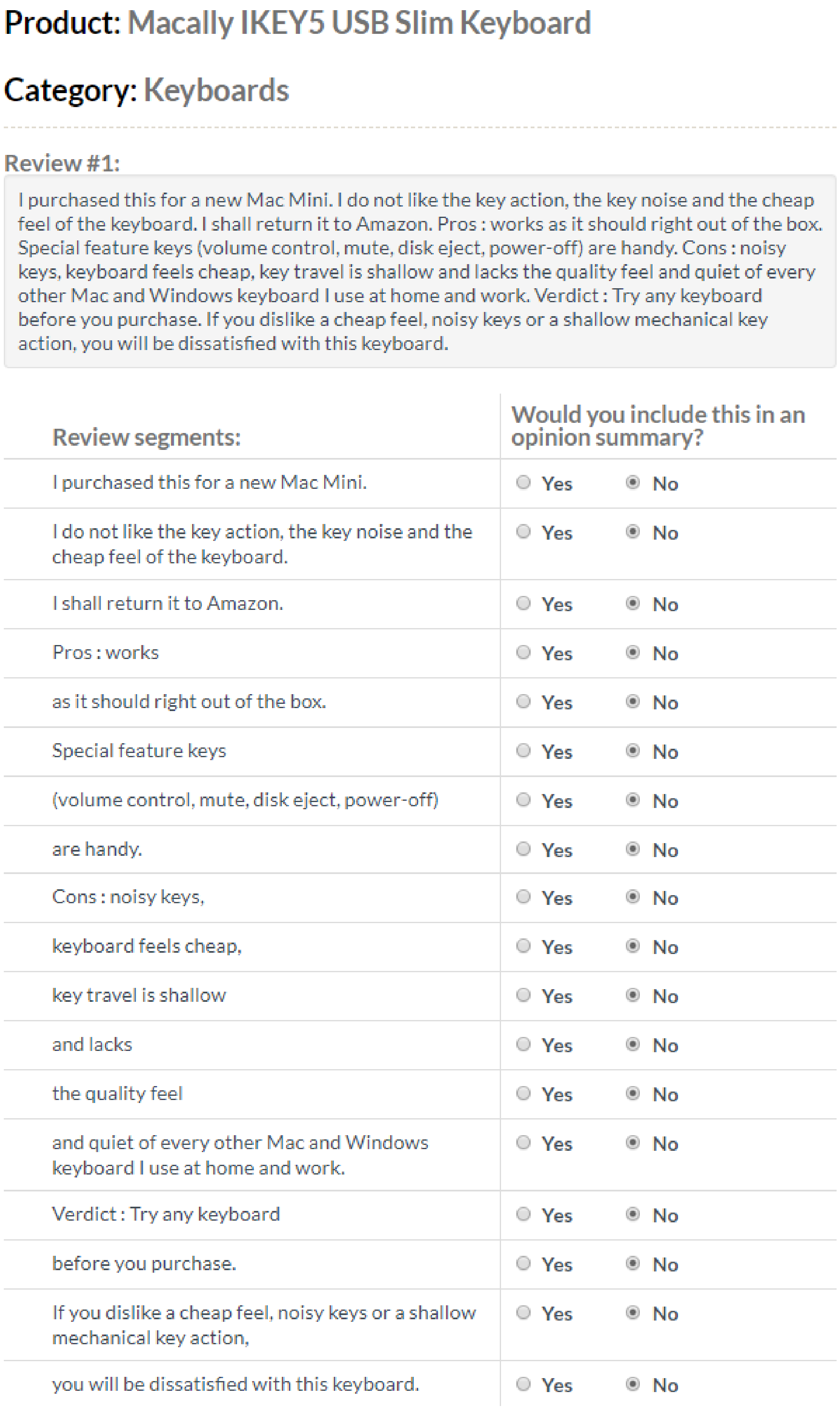
- Comments which are not about the product itself should not be selected, as they provide no useful information about the product under review. The following review example illustrates such cases:

[ ]	Was looking for a comfortable pair of shoes,	[not about the product]
[ ]	because I hate the fit in my old summer pair.	[not about the product]
[✓]	These fit as expected and are so comfortable.	[primary opinion]

- If a primary opinion of the reviewer spans more than one segment, select all segments appropriately. In the following example, note how the 1st and 4th segments are not selected, because they don't contribute to the expressed opinion. Without them the opinion still makes sense and is syntactically correct.

[ ]	That begin said	[not part of the opinion]
[✓]	I love the way	[primary opinion 1/2]
[✓]	these shoes feel on my feet,	[primary opinion 2/2]
[ ]	despite what other reviewers say.	[secondary comment]

Interface Screenshot:



Product: Macally IKEY5 USB Slim Keyboard

Category: Keyboards

Review #1:

I purchased this for a new Mac Mini. I do not like the key action, the key noise and the cheap feel of the keyboard. I shall return it to Amazon. Pros : works as it should right out of the box. Special feature keys (volume control, mute, disk eject, power-off) are handy. Cons : noisy keys, keyboard feels cheap, key travel is shallow and lacks the quality feel and quiet of every other Mac and Windows keyboard I use at home and work. Verdict : Try any keyboard before you purchase. If you dislike a cheap feel, noisy keys or a shallow mechanical key action, you will be dissatisfied with this keyboard.

Review segments:	Would you include this in an opinion summary?	
I purchased this for a new Mac Mini.	○ Yes	● No
I do not like the key action, the key noise and the cheap feel of the keyboard.	○ Yes	● No
I shall return it to Amazon.	○ Yes	● No
Pros : works	○ Yes	● No
as it should right out of the box.	○ Yes	● No
Special feature keys	○ Yes	● No
(volume control, mute, disk eject, power-off)	○ Yes	● No
are handy.	○ Yes	● No
Cons : noisy keys,	○ Yes	● No
keyboard feels cheap,	○ Yes	● No
key travel is shallow	○ Yes	● No
and lacks	○ Yes	● No
the quality feel	○ Yes	● No
and quiet of every other Mac and Windows keyboard I use at home and work.	○ Yes	● No
Verdict : Try any keyboard	○ Yes	● No
before you purchase.	○ Yes	● No
If you dislike a cheap feel, noisy keys or a shallow mechanical key action,	○ Yes	● No
you will be dissatisfied with this keyboard.	○ Yes	● No

A.4 Opinion Extraction: Phase 2 (Final Summaries)

We provide the instructions given to participants of the Figure Eight crowd-sourcing effort for the generation of reference extractive summaries, described in Section 6.5.

I. General instructions:

- Attempt HITs only if you are a **native speaker of English**.
- We are happy to **receive feedback** and improve this job accordingly. We regularly go over all contested and high miss % test questions and fix potential errors.
- Your responses are **confidential**. Any publications based on these responses will not include your specific responses, but rather aggregate information from many individuals. We will not ask any information that can be used to identify who you are.

II. Summarizing Product Reviews

The purpose of this annotation task is to create short summaries of the most important opinions expressed in a set of reviews for a product. In particular, we want to produce a 100-word opinion summary for every product, by selecting the most representative, informative and fluent segments found in 10 customer reviews. You will not be asked to write your own summary – you will simply select segments found in the original reviews.

Going directly from the 10 original reviews to a 100-word summary is a challenging task. For this reason, we have split the summarization task into 2 phases:

- **Phase 1:** Opinion extraction: the annotator selects the most important opinions expressed in each review independently.
- **Phase 2:** Final Summaries: the annotator produces a final summary, by selecting the most appropriate segments from those extracted in Phase 1. Below, we describe Phase 1. You need to finish Phase 1 to move on to Phase 2.

III. Phase 2: Final Summaries

It is now time to create your final opinion summaries for each product. In this task, you are provided with the set of opinions you extracted from all 10 reviews of a product in

Phase 1. You are now asked to construct a summary of **100 words at most**, by only selecting a subset of these opinions, while grouping them into positive and negative ones. Your summaries don't necessarily need to be near 100-words long, but you should make sure you cover **all important opinions**. Often, this is possible with much fewer than 100 words.

Segments should be selected based on:

- **Redundancy:** Similar or identical opinions should not appear in the summary more than once.
- **Clarity:** When deciding between similar opinions, pick the segments that state an opinion in a clearer, more 'to the point' way.
- **Fluency:** Also, when deciding between similar opinions, the more fluent segments should be selected.
- **Popularity:** When word count is an issue, discard the opinions that appear fewer times.
- **Polarity:** Opinions should be appropriately split into positive and negative ones. Opposite opinions ("These were cheap" / "Way too expensive for me") should appear in the summary even if they contradict each other.

Things that should not influence your selections are:

- **Capitalization:** It is fine for a segment to begin with a lowercase letter.
- **Punctuation:** It is fine for a segment to end without a punctuation symbol (.!?) or to end in a comma or semicolon even if the rest of the sentence is not selected.

Below, we provide an example that illustrates the above criteria. Positive selected segments are marked with a [✓], whereas negative selected segments are marked with a [✓].

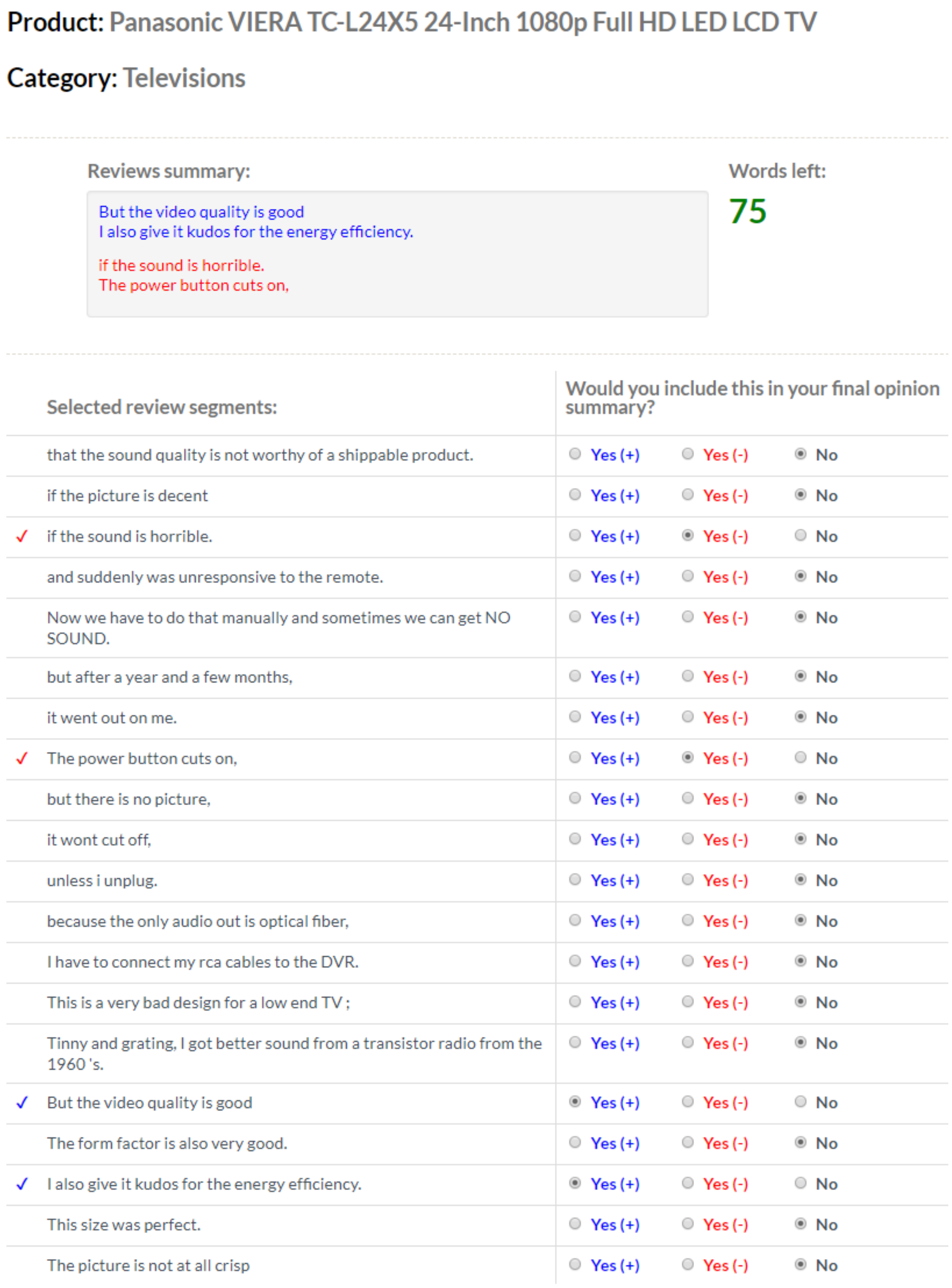
(1) [ ] and are amazingly waterproof. [less fluent than (7)]
(2) [ ] They are perfect for being on your feet all day [less clear than (4)]
(3) [✓] these shoes definately run small. [lowercase 1st letter is fine]
(4) [✓] I liked these shoes for the fit and comfort, [comma at the end is fine]
(5) [✓] Nice looking shoes too.
(6) [✓] My pair was anything but waterproof. [opposite opinion to (7)]
(7) [✓] they are waterproof!!
(8) [ ] A complete waste of money. [redundant due to (9)]
(9) [✓] Definitely too expensive.

The above selection would result in the following summary (actual summaries will be longer than this):

I liked these shoes for the fit and comfort.
Nice looking shoes too.
they are waterproof!!
these shoes definitely run small.
My pair was anything but waterproof.
Definitely too expensive.

Again, judging whether a particular segment is worth putting into the final summary is, to some extent, subjective. We provide you with a flexible interface that allows you to view the summary as you build it and make changes to the selections you have already made so that the final summary is as good as possible. The remaining number of words is also shown at all times.

Interface Screenshot:



Product: Panasonic VIERA TC-L24X5 24-Inch 1080p Full HD LED LCD TV

Category: Televisions

Reviews summary:

But the video quality is good
I also give it kudos for the energy efficiency.

if the sound is horrible.
The power button cuts on,

Words left:
75

	Selected review segments:	Would you include this in your final opinion summary?		
	that the sound quality is not worthy of a shippable product.	○ Yes (+)	○ Yes (-)	● No
	if the picture is decent	○ Yes (+)	○ Yes (-)	● No
✓	if the sound is horrible.	○ Yes (+)	● Yes (-)	○ No
	and suddenly was unresponsive to the remote.	○ Yes (+)	○ Yes (-)	● No
	Now we have to do that manually and sometimes we can get NO SOUND.	○ Yes (+)	○ Yes (-)	● No
	but after a year and a few months,	○ Yes (+)	○ Yes (-)	● No
	it went out on me.	○ Yes (+)	○ Yes (-)	● No
✓	The power button cuts on,	○ Yes (+)	● Yes (-)	○ No
	but there is no picture,	○ Yes (+)	○ Yes (-)	● No
	it wont cut off,	○ Yes (+)	○ Yes (-)	● No
	unless i unplug.	○ Yes (+)	○ Yes (-)	● No
	because the only audio out is optical fiber,	○ Yes (+)	○ Yes (-)	● No
	I have to connect my rca cables to the DVR.	○ Yes (+)	○ Yes (-)	● No
	This is a very bad design for a low end TV ;	○ Yes (+)	○ Yes (-)	● No
	Tinny and grating, I got better sound from a transistor radio from the 1960 's.	○ Yes (+)	○ Yes (-)	● No
✓	But the video quality is good	● Yes (+)	○ Yes (-)	○ No
	The form factor is also very good.	○ Yes (+)	○ Yes (-)	● No
✓	I also give it kudos for the energy efficiency.	● Yes (+)	○ Yes (-)	○ No
	This size was perfect.	○ Yes (+)	○ Yes (-)	● No
	The picture is not at all crisp	○ Yes (+)	○ Yes (-)	● No

Appendix B

Instructions for Human Evaluation of Summarization Systems

B.1 Single-Review Summarization

We provide the human evaluation study instructions given to participants of our single-review summarization system comparison. The study was performed on Amazon Mechanical Turk[1], and was described in Section 4.7.3.3.

I. General instructions:

- Attempt HITs only if you are a **native speaker of English**.
- We are happy to **receive feedback** and improve this job accordingly. For comments please contact `<e-mail address>`.
- Your responses are **confidential**. Any publications based on these responses will not include your specific responses, but rather aggregate information from many individuals. We will not ask any information that can be used to identify who you are.

[1] `https://www.mturk.com/`

II. Judging Review Summaries:

In this task, you are given an **original** review, taken from the Yelp website, and **two summaries** produced by different **automatic** summarization systems. The summaries use a bullet-style format to capture the **most salient sentiment-carrying** elements of the original review. A summary may consist of any number of positive (blue) or negative (red) bullets.

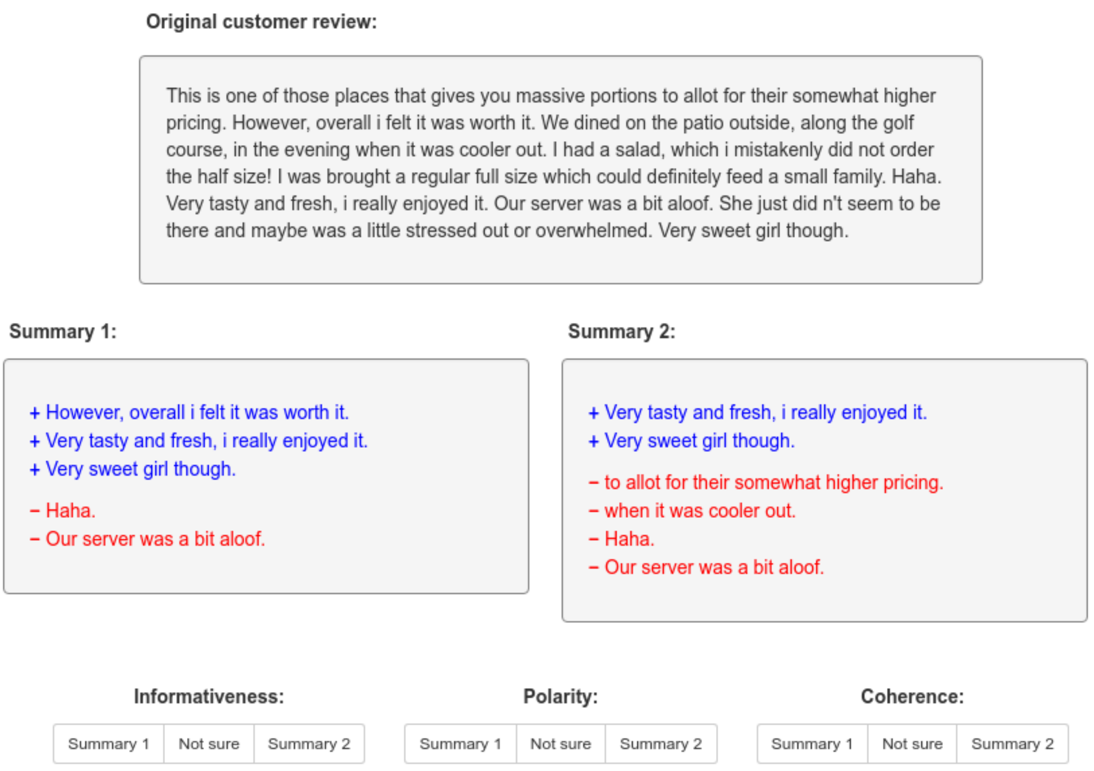
We ask you to carefully read the review and the two summaries and decide which one best captures the sentiment of the original text. You will judge the summaries on the following criteria:

Informativeness: Which summary best captures the salient points of the review?

Polarity: Which summary best highlights the positive and negative comments?

Coherence: Which summary is more coherent and easier to read?

Interface Screenshot:



Original customer review:

This is one of those places that gives you massive portions to allot for their somewhat higher pricing. However, overall i felt it was worth it. We dined on the patio outside, along the golf course, in the evening when it was cooler out. I had a salad, which i mistakenly did not order the half size! I was brought a regular full size which could definitely feed a small family. Haha. Very tasty and fresh, i really enjoyed it. Our server was a bit aloof. She just did n't seem to be there and maybe was a little stressed out or overwhelmed. Very sweet girl though.

Summary 1:

+ However, overall i felt it was worth it.
+ Very tasty and fresh, i really enjoyed it.
+ Very sweet girl though.

– Haha.
– Our server was a bit aloof.

Summary 2:

+ Very tasty and fresh, i really enjoyed it.
+ Very sweet girl though.

– to allot for their somewhat higher pricing.
– when it was cooler out.
– Haha.
– Our server was a bit aloof.

B.2 Multi-Review Summarization

We provide the human evaluation study instructions given to participants of our multi-review summarization system comparison. The study was performed on Figure Eight[2], and was described in Section 6.6.3.

I. General instructions:

- Attempt HITs only if you are a **native speaker of English**.
- We are happy to **receive feedback** and improve this job accordingly. We regularly go over all contested and high miss % test questions and fix potential errors.
- Your responses are **confidential**. Any publications based on these responses will not include your specific responses, but rather aggregate information from many individuals. We will not ask any information that can be used to identify who you are.

II. Evaluating Opinion Summaries of Product Reviews

Your task it to read three short texts which have been produced by different automatic summarization systems. The texts summarize the most important opinions expressed in customer reviews for various products sold on Amazon. Please read the three summaries carefully and judge how good each summary is according to the following criteria:

Informativeness: How much useful information about the product does the summary provide?

Polarity: How well does the summary highlight positive and negative opinions?

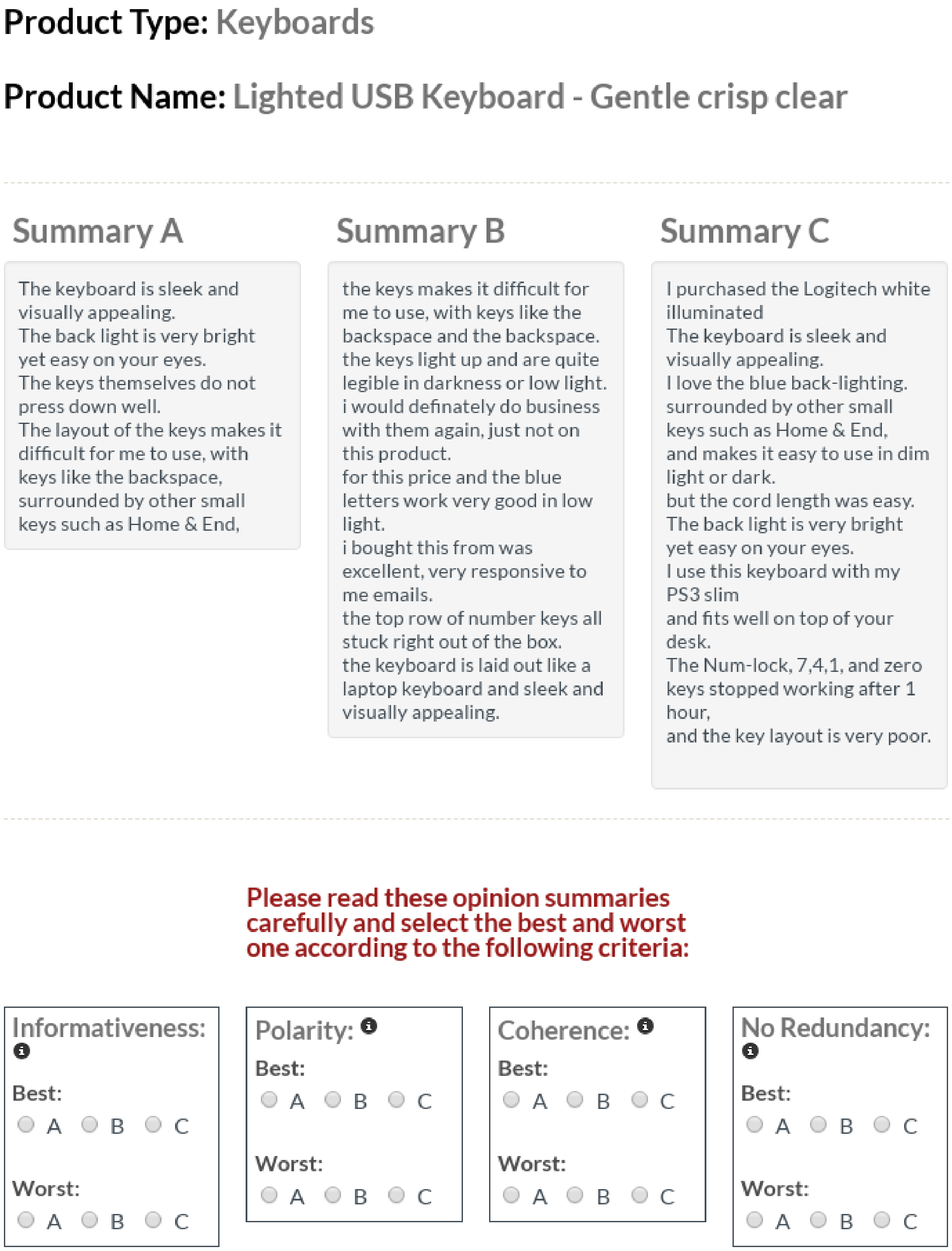
Coherence: How coherent and easy to read is the summary?

No redundancy: Is the summary successful at avoiding redundant and repeated opinions?

For each of the criteria, you must select the summary you believe is the **best** and the summary you believe is the **worst**, using the interface below.

[2] `https://www.figure-eight.com/`

Interface Screenshot:



Product Type: Keyboards

Product Name: Lighted USB Keyboard - Gentle crisp clear

Summary A

The keyboard is sleek and visually appealing.
The back light is very bright yet easy on your eyes.
The keys themselves do not press down well.
The layout of the keys makes it difficult for me to use, with keys like the backspace, surrounded by other small keys such as Home & End,

Summary B

the keys makes it difficult for me to use, with keys like the backspace and the backspace.
the keys light up and are quite legible in darkness or low light.
i would definately do business with them again, just not on this product.
for this price and the blue letters work very good in low light.
i bought this from was excellent, very responsive to me emails.
the top row of number keys all stuck right out of the box.
the keyboard is laid out like a laptop keyboard and sleek and visually appealing.

Summary C

I purchased the Logitech white illuminated
The keyboard is sleek and visually appealing.
I love the blue back-lighting.
surrounded by other small keys such as Home & End,
and makes it easy to use in dim light or dark.
but the cord length was easy.
The back light is very bright yet easy on your eyes.
I use this keyboard with my PS3 slim
and fits well on top of your desk.
The Num-lock, 7,4,1, and zero keys stopped working after 1 hour,
and the key layout is very poor.

Please read these opinion summaries carefully and select the best and worst one according to the following criteria:

Informativeness: ⓘ
Best: ○ A ○ B ○ C
Worst: ○ A ○ B ○ C

Polarity: ⓘ
Best: ○ A ○ B ○ C
Worst: ○ A ○ B ○ C

Coherence: ⓘ
Best: ○ A ○ B ○ C
Worst: ○ A ○ B ○ C

No Redundancy: ⓘ
Best: ○ A ○ B ○ C
Worst: ○ A ○ B ○ C

Bibliography

Andrews, S. and Hofmann, T. (2004). Multiple instance learning via disjunctive programming boosting. In *Advances in Neural Information Processing Systems 16*, pages 65–72. Curran Associates, Inc.

Angelidis, S. and Lapata, M. (2018a). Multiple Instance Learning Networks for Fine-Grained Sentiment Analysis. *Transactions of the Association for Computational Linguistics*, 6:17–31.

Angelidis, S. and Lapata, M. (2018b). Summarizing opinions: Aspect extraction meets sentiment prediction and they are both weakly supervised. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, Brussels, Belgium.

Baccianella, S., Esuli, A., and Sebastiani, F. (2010). SentiWordNet 3.0: An enhanced lexical resource for sentiment analysis and opinion mining. In *Proceedings of the 5th Conference on International Language Resources and Evaluation*, volume 10, pages 2200–2204, Valletta, Malta.

Bahdanau, D., Cho, K., and Bengio, Y. (2015). Neural machine translation by jointly learning to align and translate. In *Proceedings of the 3rd International Conference on Learning Representations*, San Diego, California, USA.

Barbosa, L. and Feng, J. (2010). Robust sentiment detection on twitter from biased and noisy data. In *Proceedings of the 23rd International Conference on Computational Linguistics: Posters*, COLING '10, pages 36–44, Stroudsburg, PA, USA. Association for Computational Linguistics.

Beineke, P., Hastie, T., Manning, C., and Vaithyanathan, S. (2004). An exploration of sentiment summarization. In *Proceedings of the AAAI Spring Symposium on Exploring Attitude and Affect in Text: Theories and Applications*, volume 3, pages 12–15, Stanford, US.

Bhatia, P., Ji, Y., and Eisenstein, J. (2015). Better document-level sentiment analysis from RST discourse parsing. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 2212–2218, Lisbon, Portugal.

Blair-Goldensohn, S., Hannan, K., McDonald, R., Neylon, T., Reis, G., and Reynar, J. (2008). Building a sentiment summarizer for local service reviews. In *Proceedings of the WWW Workshop on NLP Challenges in the Information Explosion Era (NLPIX)*, Beijing, China.

Blei, D. M., Ng, A. Y., and Jordan, M. I. (2003). Latent dirichlet allocation. *Journal of machine Learning research*, 3(Jan):993–1022.

Branavan, S., Chen, H., Eisenstein, J., and Barzilay, R. (2008). Learning document-level semantic properties from free-text annotations. In *Proceedings of ACL-08: HLT*, pages 263–271. Association for Computational Linguistics.

Brody, S. and Elhadad, N. (2010). An unsupervised aspect-sentiment model for online reviews. In *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 804–812. Association for Computational Linguistics.

Cao, Z., Li, W., Li, S., and Wei, F. (2017). Improving multi-document summarization via text classification. In *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence*, pages 3053–3059, San Francisco, California, USA.

Cao, Z., Wei, F., Dong, L., Li, S., and Zhou, M. (2015). Ranking with recursive neural networks and its application to multi-document summarization. In *Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence*, pages 2153–2159, Austin, Texas, USA.

Carbonetto, P., Dorkó, G., Schmid, C., Kück, H., and De Freitas, N. (2008). Learning to recognize objects with little supervision. *International Journal of Computer Vision*, 77(1):219–237.

Carenini, G., Ng, R., and Pauls, A. (2006). Multi-document summarization of evaluative text. In *Proceedings of the 11th Conference of the European Chapter of the Association for Computational Linguistics*, pages 305–312, Trento, Italy.

Chen, D. and Manning, C. (2014). A fast and accurate dependency parser using neural networks. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 740–750. Association for Computational Linguistics.

Cheng, J., Dong, L., and Lapata, M. (2016). Long short-term memory-networks for machine reading. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 551–561. Association for Computational Linguistics.

Cheng, J. and Lapata, M. (2016). Neural summarization by extracting sentences and words. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 484–494, Berlin, Germany.

Cho, K., van Merrienboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., and Bengio, Y. (2014). Learning phrase representations using rnn encoder–decoder for statistical machine translation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1724–1734. Association for Computational Linguistics.

Chung, J., Gulcehre, C., Cho, K., and Bengio, Y. (2014). Empirical evaluation of gated recurrent neural networks on sequence modeling. In *Proceedings of the Deep Learning and Representation Learning Workshop: NIPS 2014*.

Cour, T., Sapp, B., and Taskar, B. (2011). Learning from partial labels. *Journal of Machine Learning Research*, 12(May):1501–1536.

Cronen-Townsend, S., Zhou, Y., and Croft, W. B. (2002). Predicting query performance. In *Proceedings of the 25th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '02, pages 299–306, New York, NY, USA.

David, H. A. (1963). *The Method of Paired Comparisons*. Hafner Publishing Company.

Denil, M., Demiraj, A., and de Freitas, N. (2014). Extraction of salient sentences from labelled documents. Technical report, University of Oxford.

Di Fabbrizio, G., Stent, A., and Gaizauskas, R. (2014). A hybrid approach to multi-document summarization of opinions in reviews. In *Proceedings of the 8th Inter-*

national Natural Language Generation Conference (INLG), pages 54–63, Philadelphia, Pennsylvania, USA.

Diao, Q., Qiu, M., Wu, C.-Y., Smola, A. J., Jiang, J., and Wang, C. (2014). Jointly modeling aspects, ratings and sentiments for movie recommendation (JMARS). In *Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 193–202, New York, NY, USA.

Dietterich, T. G., Lathrop, R. H., and Lozano-Pérez, T. (1997). Solving the multiple instance problem with axis-parallel rectangles. *Artificial Intelligence*, 89(1):31 – 71.

Dong, L. and Lapata, M. (2018). Coarse-to-fine decoding for neural semantic parsing. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 731–742. Association for Computational Linguistics.

Duan, W., Gu, B., and Whinston, A. B. (2008). The dynamics of online word-of-mouth and product sales—an empirical investigation of the movie industry. *Journal of Retailing*, 84(2):233 – 242.

Dyer, C., Ballesteros, M., Ling, W., Matthews, A., and Smith, N. A. (2015). Transition-based dependency parsing with stack long short-term memory. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 334–343. Association for Computational Linguistics.

Erkan, G. and Radev, D. R. (2004). Lexrank: Graph-based lexical centrality as salience in text summarization. *Journal of Artificial Intelligence Research*, 22:457–479.

Feng, W. V. and Hirst, G. (2012). Text-level discourse parsing with rich linguistic features. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics*, volume 1, pages 60–68, Jeju Island, Korea.

Filippova, K., Alfonseca, E., Colmenares, C. A., Kaiser, L., and Vinyals, O. (2015). Sentence compression by deletion with lstms. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 360–368. Association for Computational Linguistics.

Ganesan, K., Zhai, C., and Han, J. (2010). Opinosis: a graph-based approach to abstractive summarization of highly redundant opinions. In *Proceedings of the 23rd*

International Conference on Computational Linguistics, pages 340–348, Beijing, China.

Gerani, S., Mehdad, Y., Carenini, G., Ng, R. T., and Nejat, B. (2014). Abstractive summarization of product reviews using discourse structure. In *EMNLP*, pages 1602–1613.

Gillick, D. and Favre, B. (2009). A scalable global model for summarization. In *Proceedings of the Workshop on Integer Linear Programming for Natural Language Processing*, pages 10–18. Association for Computational Linguistics.

Goldstein, J. and Carbonell, J. (1998). Summarization: (1) using mmr for diversity-based reranking and (2) evaluating summaries. In *Proceedings of TIPSTER Text Program Phase III Workshop*.

Gregor, K., Danihelka, I., Graves, A., Rezende, D., and Wierstra, D. (2015). Draw: A recurrent neural network for image generation. In Bach, F. and Blei, D., editors, *Proceedings of the 32nd International Conference on Machine Learning*, volume 37 of *Proceedings of Machine Learning Research*, pages 1462–1471, Lille, France. PMLR.

He, R., Lee, W. S., Ng, H. T., and Dahlmeier, D. (2017). An unsupervised neural attention model for aspect extraction. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics*, volume 1, pages 388–397, Vancouver, Canada.

He, R. and McAuley, J. (2016). Ups and downs: Modeling the visual evolution of fashion trends with one-class collaborative filtering. In *proceedings of the 25th international conference on world wide web*, pages 507–517. International World Wide Web Conferences Steering Committee.

Hochreiter, S. and Schmidhuber, J. (1997). Long short-term memory. *Neural Comput.*, 9(8):1735–1780.

Hoffmann, R., Zhang, C., Ling, X., Zettlemoyer, L., and Weld, D. S. (2011). Knowledge-based weak supervision for information extraction of overlapping relations. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies-Volume 1*, pages 541–550, Portland, Oregon, USA.

Hofmann, T. (1999). Probabilistic latent semantic indexing. In *Proceedings of the 22Nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR ’99, pages 50–57, New York, NY, USA. ACM.

Hu, M. and Liu, B. (2004). Mining and summarizing customer reviews. In *Proceedings of the 10th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 168–177, Seattle, Washington, USA.

Hu, M. and Liu, B. (2006). Opinion extraction and summarization on the web. In *Proceedings of the 21st National Conference on Artificial Intelligence*, pages 1621–1624, Boston, Massachusettes, USA.

Iyyer, M., Guha, A., Chaturvedi, S., Boyd-Graber, J., and Daumé III, H. (2016). Feuding families and former friends: Unsupervised learning for dynamic fictional relationships. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1534–1544, San Diego, California.

Jin, W. and Ho, H. H. (2009). A novel lexicalized hmm-based learning framework for web opinion mining. In *Proceedings of the 26th Annual International Conference on Machine Learning*, ICML ’09, pages 465–472, New York, NY, USA. ACM.

Johnson, R. and Zhang, T. (2015a). Effective use of word order for text categorization with convolutional neural networks. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 103–112, Denver, Colorado, USA.

Johnson, R. and Zhang, T. (2015b). Semi-supervised convolutional neural networks for text categorization via region embedding. In *Advances in Neural Information Processing Systems 28*, pages 919–927. Curran Associates, Inc.

Jurgens, D. A., Turney, P. D., Mohammad, S. M., and Holyoak, K. J. (2012). Semeval-2012 task 2: Measuring degrees of relational similarity. In *Proceedings of the First Joint Conference on Lexical and Computational Semantics-Volume 1: Proceedings of the main conference and the shared task, and Volume 2: Proceedings of the Sixth International Workshop on Semantic Evaluation*, pages 356–364. Association for Computational Linguistics.

Kalchbrenner, N. and Blunsom, P. (2013). Recurrent convolutional neural networks for discourse compositionality. In *Proceedings of the Workshop on Continuous Vector Space Models and their Compositionality*, pages 119–126. Association for Computational Linguistics.

Keeler, J. and Rumelhart, D. E. (1992). A self-organizing integrated segmentation and recognition neural net. In *Advances in Neural Information Processing Systems 4*, pages 496–503. Morgan-Kaufmann.

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

Kim, Y., Jernite, Y., Sontag, D., and Rush, A. (2016). Character-aware neural language models. In *30th AAAI Conference on Artificial Intelligence, AAAI 2016*, pages 2741–2749. AAAI press.

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

Kiritchenko, S. and Mohammad, S. (2017). Best-worst scaling more reliable than rating scales: A case study on sentiment intensity annotation. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, volume 2, pages 465–470, Vancouver, Canada.

Kiritchenko, S. and Mohammad, S. M. (2016). Sentiment composition of words with opposing polarities. In *Proceedings of The 15th Annual Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL), San Diego, California*. Association for Computational Linguistics.

Koehn, P. (2004). Statistical significance tests for machine translation evaluation. In Lin, D. and Wu, D., editors, *Proceedings of EMNLP 2004*, pages 388–395, Barcelona, Spain.

Kotzias, D., Denil, M., De Freitas, N., and Smyth, P. (2015). From group to individual labels using deep features. In *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 597–606, Sydney, Australia.

Ku, L.-W., Liang, Y.-T., and Chen, H.-H. (2006). Opinion extraction, summarization and tracking in news and blog corpora. In *AAAI Syposium on Computational Approaches to Analysing Weblogs*, pages 100–107, Palo Alto, California, USA.

Lazaridou, A., Titov, I., and Sporleder, C. (2013). A bayesian model for joint unsupervised induction of sentiment, aspect and discourse representations. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1630–1639. Association for Computational Linguistics.

Le, Q. and Mikolov, T. (2014). Distributed representations of sentences and documents. In *Proceedings of the 31st International Conference on Machine Learning*, pages 1188–1196, Beijing, China.

Lerman, K., Blair-Goldensohn, S., and McDonald, R. (2009). Sentiment summarization: Evaluating and learning user preferences. In *Proceedings of the 12th Conference of the European Chapter of the Association for Computational Linguistics*, EACL '09, pages 514–522. Association for Computational Linguistics.

Li, F., Han, C., Huang, M., Zhu, X., Xia, Y.-J., Zhang, S., and Yu, H. (2010). Structure-aware review mining and summarization. In *Proceedings of the 23rd International Conference on Computational Linguistics (Coling 2010)*, pages 653–661. Coling 2010 Organizing Committee.

Li, J. J., Thadani, K., and Stent, A. (2016). The role of discourse units in near-extractive summarization. In *Proceedings of the SIGDIAL 2016 Conference, The 17th Annual Meeting of the Special Interest Group on Discourse and Dialogue*, pages 137–147, Los Angeles, California, USA.

Lin, C.-Y. and Hovy, E. (2003). Automatic evaluation of summaries using n-gram co-occurrence statistics. In *Proceedings of the 2003 Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics*, pages 71–78. Association for Computational Linguistics.

Liu, B. (2012). Sentiment analysis and opinion mining. *Synthesis lectures on human language technologies*, 5(1):1–167.

Liu, B., Hu, M., and Cheng, J. (2005). Opinion observer: Analyzing and comparing opinions on the web. In *Proceedings of the 14th International Conference on World Wide Web*, WWW '05, pages 342–351, New York, NY, USA. ACM.

Liu, B. and Zhang, L. (2012). A survey of opinion mining and sentiment analysis. *Mining Text Data, Springer*, pages 415–463.

Liu, K., Xu, L., and Zhao, J. (2012). Opinion target extraction using word-based translation model. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, pages 1346–1356. Association for Computational Linguistics.

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

Liu, Y. and Lapata, M. (2018). Learning structured text representations. *Transactions of the Association for Computational Linguistics*, 6:63–75.

Louviere, J. J., Flynn, T. N., and Marley, A. A. J. (2015). *Best-worst scaling: Theory, methods and applications*. Cambridge University Press.

Louviere, J. J. and Woodworth, G. G. (1991). Best-worst scaling: A model for the largest difference judgments. *University of Alberta: Working Paper*.

Lu, Y. and Zhai, C. (2008). Opinion integration through semi-supervised topic modeling. In *Proceedings of the 17th International Conference on World Wide Web*, WWW '08, pages 121–130, New York, NY, USA. ACM.

Lu, Y., Zhai, C., and Sundaresan, N. (2009). Rated aspect summarization of short comments. In *Proceedings of the 18th International Conference on World Wide Web*, pages 131–140, Madrid, Spain.

Luong, T., Pham, H., and Manning, C. D. (2015). Effective approaches to attention-based neural machine translation. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 1412–1421. Association for Computational Linguistics.

Ma, T. and Wan, X. (2010). Opinion target extraction in chinese news comments. In *Coling 2010: Posters*, pages 782–790. Coling 2010 Organizing Committee.

Mann, W. C. and Thompson, S. A. (1988). Rhetorical structure theory: Toward a functional theory of text organization. *Text-Interdisciplinary Journal for the Study of Discourse*, 8(3):243–281.

Maron, O. and Ratan, A. L. (1998). Multiple-instance learning for natural scene classification. In *Proceedings of the 15th International Conference on Machine Learning*, volume 98, pages 341–349, San Francisco, California, USA.

McDonald, R. (2007). A study of global inference algorithms in multi-document summarization. In *Proceedings of the 29th European Conference on Information Retrieval (ECIR)*, pages 557–564.

Mei, Q., Ling, X., Wondra, M., Su, H., and Zhai, C. (2007). Topic sentiment mixture: Modeling facets and opinions in weblogs. In *Proceedings of the 16th International Conference on World Wide Web*, pages 171–180, Banff, Alberta, Canada.

Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., and Dean, J. (2013). Distributed representations of words and phrases and their compositionality. In *Advances in Neural Information Processing Systems*, pages 3111–3119, Lake Tahoe, California, USA.

Mohammad, S. M. and Bravo-Marquez, F. (2017). WASSA-2017 shared task on emotion intensity. In *Proceedings of the Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis (WASSA)*, Copenhagen, Denmark.

Mukherjee, A. and Liu, B. (2012). Aspect extraction through semi-supervised modeling. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 339–348. Association for Computational Linguistics.

Nair, V. and Hinton, G. E. (2010). Rectified linear units improve restricted boltzmann machines. In *Proceedings of the 27th International Conference on International Conference on Machine Learning*, ICML'10, pages 807–814, USA. Omnipress.

Nallapati, R., Zhai, F., and Zhou, B. (2017). SummaRuNNer: A recurrent neural network based sequence model for extractive summarization of documents. In *Proceedings of the 31st AAAI Conference on Artificial Intelligence*, pages 3075–3081, San Francisco, California.

Narayan, S., Cohen, S. B., and Lapata, M. (2018). Don't give me the details, just the summary! topic-aware convolutional neural networks for extreme summarization.

Nenkova, A. (2005). Automatic text summarization of newswire: Lessons learned from the document understanding conference. In *Proceedings of the 20th AAAI*, pages 1436–1441, Pittsburgh, Pennsylvania, USA.

Nenkova, A. and Vanderwende, L. (2005). The impact of frequency on summarization. Technical report.

Noreen, E. (1989). *Computer-intensive Methods for Testing Hypotheses: An Introduction*. Wiley.

Orme, B. (2009). Maxdiff analysis: Simple counting, individual-level logit, and HB. Technical report.

Pang, B. and Lee, L. (2005). Seeing stars: Exploiting class relationships for sentiment categorization with respect to rating scales. In *Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics*, pages 115–124. Association for Computational Linguistics.

Pang, B. and Lee, L. (2008). Opinion mining and sentiment analysis. *Foundations and Trends® in Information Retrieval*, 2(1–2):1–135.

Pang, B., Lee, L., and Vaithyanathan, S. (2002). Thumbs up? sentiment classification using machine learning techniques. In *Proceedings of the 2002 Conference on Empirical Methods in Natural Language Processing*, pages 79–86, Pittsburgh, Pennsylvania, USA.

Pappas, N. and Popescu-Belis, A. (2014). Explaining the stars: Weighted multiple-instance learning for aspect-based sentiment analysis. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing*, pages 455–466, Doha, Qatar.

Pappas, N. and Popescu-Belis, A. (2017). Explicit document modeling through weighted multiple-instance learning. *Journal of Artificial Intelligence Research*, 58:591–626.

Pennington, J., Socher, R., and Manning, C. (2014). Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 1532–1543.

Perez-Beltrachini, L. and Lapata, M. (2018). Bootstrapping generators from noisy data. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1516–1527. Association for Computational Linguistics.

Popescu, A.-M. and Etzioni, O. (2005). Extracting product features and opinions from reviews. In *Proceedings of Human Language Technology Conference and Conference on Empirical Methods in Natural Language Processing*, pages 339–346, Vancouver, British Columbia, Canada.

Qiu, G., Liu, B., Bu, J., and Chen, C. (2011). Opinion word expansion and target extraction through double propagation. *Computational Linguistics*, 37(1).

Qu, L., Ifrim, G., and Weikum, G. (2010). The bag-of-opinions method for review rating prediction from sparse text patterns. In *Proceedings of the 23rd International Conference on Computational Linguistics*, pages 913–921, Beijing, China.

Radev, D., Allison, T., Blair-Goldensohn, S., Blitzer, J., Çelebi, A., Dimitrov, S., Drabek, E., Hakim, A., Lam, W., Liu, D., Otterbacher, J., Qi, H., Saggion, H., Teufel, S., Topper, M., Winkel, A., and Zhang, Z. (2004). Mead - a platform for multidocument multilingual text summarization. In *Proceedings of the Fourth International Conference on Language Resources and Evaluation (LREC'04)*. European Language Resources Association (ELRA).

Radev, D. R., Jing, H., and Budzikowska, M. (2000). Centroid-based summarization of multiple documents: sentence extraction, utility-based evaluation, and user studies. In *NAACL-ANLP 2000 Workshop: Automatic Summarization*.

Radev, D. R., Teufel, S., Saggion, H., Lam, W., Blitzer, J., Qi, H., Çelebi, A., Liu, D., and Drabek, E. (2003). Evaluation challenges in large-scale document summarization. In *Proceedings of the 41st Annual Meeting of the Association for Computational Linguistics*, pages 375–382. Association for Computational Linguistics.

Schmitt, M., Steinheber, S., Schreiber, K., and Roth, B. (2018). Joint aspect and polarity classification for aspect-based sentiment analysis with end-to-end neural networks. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, Brussels, Belgium.

Sennrich, R., Haddow, B., and Birch, A. (2016). Neural machine translation of rare words with subword units. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1715–1725. Association for Computational Linguistics.

Socher, R., Pennington, J., Huang, E. H., Ng, A. Y., and Manning, C. D. (2011). Semi-supervised recursive autoencoders for predicting sentiment distributions. In *Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing*, pages 151–161, Edinburgh, Scotland, UK.

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

Taboada, M., Brooke, J., Tofiloski, M., Voll, K., and Stede, M. (2011). Lexicon-based methods for sentiment analysis. *Computational Linguistics*, 37(2):267–307.

Täckström, O. and McDonald, R. (2011). Discovering fine-grained sentiment with latent variable structured prediction models. In *Proceedings of the 39th European Conference on Information Retrieval*, pages 368–374, Aberdeen, Scotland, UK.

Tang, D., Qin, B., and Liu, T. (2015a). Document modeling with gated recurrent neural network for sentiment classification. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 1422–1432, Lisbon, Portugal.

Tang, D., Qin, B., and Liu, T. (2015b). Learning semantic representations of users and products for document level sentiment classification. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 1014–1023, Beijing, China. Association for Computational Linguistics.

Titov, I. and McDonald, R. (2008a). A joint model of text and aspect ratings for sentiment summarization. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 308–316, Columbus, Ohio, USA.

Titov, I. and McDonald, R. (2008b). Modeling online reviews with multi-grain topic models. In *Proceedings of the 17th International Conference on World Wide Web*, pages 111–120, Beijing, China.

Turney, P. D. (2002). Thumbs up or thumbs down?: Semantic orientation applied to unsupervised classification of reviews. In *Proceedings of the 40th annual meeting on association for computational linguistics*, pages 417–424, Pittsburgh, Pennsylvania, USA.

TurnTo Report (2017). TurnTo Consumer Study: User-Generated Content and the Commerce Experience. `http://www2.turntonetworks.com/2017consumerstudy`. [Online; accessed 18-September-2018].

Wang, L. and Ling, W. (2016). Neural network-based abstract generation for opinions and arguments. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 47–57. Association for Computational Linguistics.

Wang, S. and Manning, C. D. (2012). Baselines and bigrams: Simple, good sentiment and topic classification. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics: Short Papers-Volume 2*, pages 90–94, Jeju Island, Korea.

Wang, W., Pan, S. J., Dahlmeier, D., and Xiao, X. (2016). Recursive neural conditional random fields for aspect-based sentiment analysis. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 616–626, Austin, Texas, USA.

Wei, X.-S., Wu, J., and Zhou, Z.-H. (2014). Scalable multi-instance learning. In *Proceedings of the IEEE International Conference on Data Mining*, pages 1037–1042, Shenzhen, China.

Weidmann, N., Frank, E., and Pfahringer, B. (2003). A two-level learning method for generalized multi-instance problems. In *Proceedings of the 14th European Conference on Machine Learning*, pages 468–479, Dubrovnik, Croatia.

Wiebe, J., Wilson, T., and Cardie, C. (2005). Annotating expressions of opinions and emotions in language. *Language resources and evaluation*, 39(2):165–210.

Wu, J., Yu, Y., Huang, C., and Yu, K. (2015). Deep multiple instance learning for image classification and auto-annotation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3460–3469, Boston, Massachusetts, USA.

Xia, R. and Zong, C. (2010). Exploring the use of word relation features for sentiment classification. In *Proceedings of the 23rd International Conference on Computational Linguistics: Posters*, pages 1336–1344, Beijing, China.

Xu, K., Ba, J., Kiros, R., Cho, K., Courville, A., Salakhudinov, R., Zemel, R., and Bengio, Y. (2015). Show, attend and tell: Neural image caption generation with visual attention. In Bach, F. and Blei, D., editors, *Proceedings of the 32nd International Conference on Machine Learning*, volume 37 of *Proceedings of Machine Learning Research*, pages 2048–2057, Lille, France. PMLR.

Xu, W., Ritter, A., Callison-Burch, C., Dolan, W. B., and Ji, Y. (2014). Extracting lexically divergent paraphrases from Twitter. *Transactions of the Association for Computational Linguistics*, 2:435–448.

Xu, X. and Frank, E. (2004). Logistic regression and boosting for labeled bags of instances. In *Proceedings of the Pacific-Asia Conference on Knowledge Discovery and Data Mining*, pages 272–281. Springer-Verlag.

Yan, X., Guo, J., Lan, Y., and Cheng, X. (2013). A biterm topic model for short texts. In *Proceedings of the 22Nd International Conference on World Wide Web*, WWW '13, pages 1445–1456, New York, NY, USA. ACM.

Yang, Z., Yang, D., Dyer, C., He, X., Smola, A., and Hovy, E. (2016). Hierarchical attention networks for document classification. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1480–1489, San Diego, California, USA.

Yasunaga, M., Zhang, R., Meelu, K., Pareek, A., Srinivasan, K., and Radev, D. (2017). Graph-based neural multi-document summarization. In *Proceedings of the 21st Conference on Computational Natural Language Learning (CoNLL 2017)*, pages 452–462, Vancouver, Canada.

Ye, Q., Law, R., and Gu, B. (2009). The impact of online user reviews on hotel room sales. *International Journal of Hospitality Management*, 28(1):180 – 182.

Yelp Report (2017). Yelp investor presentation – fourth quartet 2017. `http://www.yelp-ir.com/events/event-details/q4-2017-yelp-inc-earnings-conference-call`. [Online; accessed 18-September-2018].

Yin, Y., Wei, F., Dong, L., Xu, K., Zhang, M., and Zhou, M. (2016). Unsupervised word and dependency path embeddings for aspect term extraction. In *Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence*, pages 2979–2985, New York, NY, USA.

Zeiler, M. D. (2012). ADADELTA: an adaptive learning rate method. *CoRR*, abs/1212.5701.

Zhang, B., Su, J., Xiong, D., Lu, Y., Duan, H., and Yao, J. (2015). Shallow convolutional neural network for implicit discourse relation recognition. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 2230–2235. Association for Computational Linguistics.

Zhang, C., Platt, J. C., and Viola, P. A. (2006). Multiple instance boosting for object detection. In *Advances in Neural Information Processing Systems 18*, pages 1417–1424. MIT Press.

Zhang, Q., Goldman, S. A., Yu, W., and Fritts, J. E. (2002). Content-based image retrieval using multiple-instance learning. In *Proceedings of the 19th International Conference on Machine Learning*, volume 2, pages 682–689, Sydney, Australia.

Zhang, X. and Lapata, M. (2017). Sentence simplification with deep reinforcement learning. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 584–594. Association for Computational Linguistics.

Zhao, X., Jiang, J., Yan, H., and Li, X. (2010). Jointly modeling aspects and opinions with a maxent-lda hybrid. In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*, pages 56–65. Association for Computational Linguistics.

Zhou, Z.-H., Sun, Y.-Y., and Li, Y.-F. (2009). Multi-instance learning by treating instances as non-iid samples. In *Proceedings of the 26th Annual International Conference on Machine Learning*, pages 1249–1256, Montréal, Quebec.