Novel Algorithms for Understanding Online Reviews

Tian Shi

Dissertation submitted to the Faculty of the
Virginia Polytechnic Institute and State University
in partial fulfillment of the requirements for the degree of

Doctor of Philosophy
in
Computer Science and Applications

Chandan K. Reddy, Chair
Naren Ramakrishnan
Chang-Tien Lu
Edward A. Fox
Karthik Subbian

August 3, 2021
Blacksburg, Virginia

Keywords: Topic Modeling, Sentiment Analysis, Aspect Detection, Model Interpretation.

Novel Algorithms for Understanding Online Reviews

Tian Shi

(ABSTRACT)

This dissertation focuses on the review understanding problem, which has gained attention from both industry and academia, and has found applications in many downstream tasks, such as recommendation, information retrieval and review summarization. In this dissertation, we aim to develop machine learning and natural language processing tools to understand and learn structured knowledge from unstructured reviews, which can be investigated in three research directions, including understanding review corpora, understanding review documents, and understanding review segments.

For the corpus-level review understanding, we have focused on discovering knowledge from corpora that consist of short texts. Since they have limited contextual information, automatically learning topics from them remains a challenging problem. We propose a semantics-assisted non-negative matrix factorization model to deal with this problem. It effectively incorporates the word-context semantic correlations into the model, where the semantic relationships between the words and their contexts are learned from the skip-gram view of a corpus. We conduct extensive sets of experiments on several short text corpora to demonstrate the proposed model can discover meaningful and coherent topics.

For document-level review understanding, we have focused on building interpretable and reliable models for the document-level multi-aspect sentiment analysis (DMSA) task, which can help us to not only recover missing aspect-level ratings and analyze sentiment of customers, but also detect aspect and opinion terms from reviews. We conduct three studies in this research direction. In the first study, we collect a new DMSA dataset in the healthcare domain and systematically investigate reviews in this dataset, including a comprehensive statistical analysis and topic modeling to discover aspects. We also propose a multi-task learning framework with self-attention networks to predict sentiment and ratings for given aspects. In the second study, we propose corpus-level and concept-based explanation methods to interpret attention-based deep learning models for text classification, including sentiment classification. The proposed corpus-level explanation approach aims to capture causal relationships between keywords and model predictions via learning importance of keywords for predicted labels across a training corpus based on attention weights. We also propose a concept-based explanation method that can automatically learn higher level concepts and their importance to model predictions. We apply these methods to the classification task and show that they are powerful in extracting semantically meaningful keywords and concepts, and explaining model predictions. In the third study, we propose an interpretable and uncertainty aware multi-task learning framework for DMSA, which can achieve competitive performance while also being able to interpret the predictions made. Based on the corpus-level explanation method, we propose an attention-driven keywords ranking method, which can automatically discover aspect terms and aspect-level opinion terms from a review corpus using the attention

weights. In addition, we propose a lecture-audience strategy to estimate model uncertainty in the context of multi-task learning.

For the segment-level review understanding, we have focused on the unsupervised aspect detection task, which aims to automatically extract interpretable aspects and identify aspect-specific segments from online reviews. The existing deep learning-based topic models suffer from several problems such as extracting noisy aspects and poorly mapping aspects discovered by models to the aspects of interest. To deal with these problems, we propose a self-supervised contrastive learning framework in order to learn better representations for aspects and review segments. We also introduce a high-resolution selective mapping method to efficiently assign aspects discovered by the model to the aspects of interest. In addition, we propose using a knowledge distillation technique to further improve the aspect detection performance.

Novel Algorithms for Understanding Online Reviews

Tian Shi

(GENERAL AUDIENCE ABSTRACT)

Nowadays, online reviews are playing an important role in our daily lives. They are also critical to the success of many e-commerce and local businesses because they can help people build trust in brands and businesses, provide insights into products and services, and improve consumers' confidence. As a large number of reviews accumulate every day, a central research problem is to build an artificial intelligence system that can understand and interact with these reviews, and further use them to offer customers better support and services. In order to tackle challenges in these applications, we first have to get an in-depth understanding of online reviews.

In this dissertation, we focus on the review understanding problem and develop machine learning and natural language processing tools to understand reviews and learn structured knowledge from unstructured reviews. We have addressed the review understanding problem in three directions, including understanding a collection of reviews, understanding a single review, and understanding a piece of a review segment. In the first direction, we proposed a short-text topic modeling method to extract topics from review corpora that consist of primary complaints of consumers. In the second direction, we focused on building sentiment analysis models to predict the opinions of consumers from their reviews. Our deep learning models can provide good prediction accuracy as well as a human-understandable explanation for the prediction. In the third direction, we develop an aspect detection method to automatically extract sentences that mention certain features consumers are interested in, from reviews, which can help customers efficiently navigate through reviews and help businesses identify the advantages and disadvantages of their products.

Dedications

To my beloved wife and daughter.

Acknowledgements

First and foremost, I would like to thank my advisor Dr. Chandan K. Reddy for all his guidance during my pursuit of this PhD degree, which brought me to the finish line. In the past five years, he has given me lots of suggestions in my research projects, and provided many great opportunities for me to practice my writing, presentations and teaching, and collaborate with other students and researchers. I believe these skills will help me succeed in my future career. He has also created an amazing research environment for our team, which not only strengthened discussion and collaboration among team members, but also broadened our knowledge via sharing many valuable and up-to-date resources. I have enjoyed this environment and will never forget his help and support to make my doctorate study successful.

I would also like to thank all my committee members: Prof. Naren Ramakrishnan, Prof. Chang-Tien Lu, Prof. Edward A. Fox, and Dr. Karthik Subbian, who have provided valuable suggestions to my work and helped me accomplish my dissertation. Thanks go to Prof. Naren Ramakrishnan for the insightful discussions in the text summarization projects. Thanks go to Prof. Chang-Tien Lu, who was very caring and helpful during my PhD. He also gave me a number of valuable comments in my research and provided many suggestions for my future career path. Thanks go to Prof. Edward A. Fox for providing valuable comments and suggestions in my preliminary proposal, online review related projects and final dissertation. Thanks go to Dr. Karthik Subbian for helping me strengthen the motivation of my research projects and improve my presentation.

Also, I want to thank my collaborators: Dr. Jaegul Choo, Dr. Kyeongpil Kang, Dr. Naren Ramakrishnan, Dr. Yaser Keneshloo, Dr. Vineeth Rakesh, Dr. Suhang Wang, Dr. Xuchao Zhang, Dr. Liuqing Li, Ping Wang, Dr. Sutanay Choudhury, Khushbu Agarwal, Colby M Ham for helping me find research directions, solve challenging problems in my projects, and fix writing issues in my manuscripts.

Moreover, I would like to express my sincere thanks to my colleagues and friends in the Sanghani Center for Artificial Intelligence & Data Analytics and the Department of Computer Science: Khoa Doan, Aman Ahuja, Ming Zhu, Sindhu Tipirneni, Nurendra Choudhary, Akshita Jha, Dr. Rupinder Khandpur, Dr. Yue Ning, Dr. Rongrong Tao, Dr. Xu Shi, Dr. Tianyi Li, Lijing Wang, Dr. Xiangyu Zhang, Dr. Mengmeng Cai, Dr. Jinshan Liu, Dr. Wei

Wang, Dr. Sirui Yao, You Lu, Andreea Sistrunk, Dr. Sathappan Muthiah, Dr. Mohammad Raihanul Islam, Dr. Sneha Mehta, Nikhil Muralidhar, Debanjan Datta, Subhodip Biswas, Dr. Ting Hua, Fanglan Chen, Kaiqun Fu, Dr. Zhiqian Chen, Taoran Ji, Lei Zhang, Dr. Saurabh Chakravarty, Shengzhe Xu, Xiang Chen, who have encouraged and assisted me, and made me better at what I do. Thank you for adding so many good memories and times of happiness to my life. I also would like to take this opportunity to thank Roxanne Paul, Sharon Kinder-Potter, Juanita Victoria, Wanawsha Hawrami, and Joyce Newberry for their incredible support.

Furthermore, I would like to especially thank my parents and parents-in-law for their understanding, support and encouragement.

Finally, I would like to express my special thanks to my beloved wife. Words can never be enough to express how thankful I am. Thanks for her faith, confidence, and never-ending love. Because of her, I became a better man. Because of her, I keep striving to be a better husband.

Table of Contents

List of Figures

List of Tables

Chapter 1

Introduction

Nowadays, online reviews have influenced many aspects of our daily life, such as shopping, traveling, dining, housing, education, healthcare, etc. They are also very important to the success of e-commerce and local business. In recent years, many surveys[1] have been conducted and revealed that (1) most consumers research products and services online before making purchases or finding local businesses; (2) they also read online reviews for guidance; (3) most of them have written reviews for these digital or local businesses. Generally speaking, online reviews can help people build trust in brands and businesses. They can also provide insights into products and services. In addition, they can improve consumers' confidence when they are making choices.

There are many review platforms, some of which are integrated with e-commerce (e.g., Amazon and eBay), while others are independent professional review systems (e.g., Yelp and TripAdvisor) for consumers to share their opinions. Traditionally, most of these platforms only collect reviews and overall ratings from customers. Then, they show the collected reviews and distributions of overall ratings to customers. For most platforms, there are also question and answer systems which allow customers to ask product-related questions and get answers from other customers and businesses. Nowadays, due to the rapid increase in the number of reviews, many new features are needed to help customers navigate through reviews effectively. For example, in the recent few years, *aspect-level* information (e.g., location, cleanliness, service and value for a hotel) has gained increasing attention in a number of online review platforms, such as Amazon, Best Buy, TripAdvisor, BeerAdvocate, and RateMDs. These platforms request users to provide aspect-level feedback in their reviews, e.g., aspect/feature-level ratings, which can benefit digital stores in a variety of ways: (1) They can quickly identify defects of products and provide feedback to their manufacturers. For example, *Sound Quality*, *Picture Quality*, *Smart Features*, and *Remote Control* are important features/aspects for televisions in Amazon product reviews. A low aspect-level rating implies potential problems for the corresponding features. (2) Platforms can build profiles of customers and products

[1] For example, `https://www.brightlocal.com/research/local-consumer-review-survey/`

based on aspect-level information, which can be very useful for personalized recommender systems. For instance, *Camera, Flying, Picture Quality, Video Quality*, and *Ease of Use* are pros mentions for “DJI - Mavic 2 Pro Quadcopter” in the Best Buy review platform; therefore, it is feasible to recommend this product to users who are looking for quadcopters that are easy to control and carry high-resolution cameras. (3) Customers can choose products that meet their needs more efficiently without reading too many reviews. For example, for the same quadcopter, cons mentions are *App, Control from phone, Instructions, Obstacle avoidance, Loud*, which are linked to corresponding reviews. If customers have more concerns about *Control from phone*, they can read those reviews and decide if this defect is acceptable.

1.1 Motivation

There are many research opportunities and challenges related to online reviews. First, several studies have found that customers are less motivated to give aspect-level feedback [125, 163], which makes it difficult to analyze their preference, and it takes a lot of time and effort for human experts to manually annotate them. Automatic rating prediction and sentiment analysis grounded on textual reviews have been used to solve this problem [163, 167, 74, 169, 125]. Second, traditional collaborative filtering (CF) based recommender systems primarily rely on overall ratings (binary or integers) and meta-data of users and items [120]. Recently, recommender systems, which also consider using customer reviews to build representations of users and items, have been widely studied due to advances of deep learning models [13]. On the one hand, review content provides richer contextual information of users and items [76, 175, 93], which can alleviate the sparsity problem in CF based systems and improve the prediction accuracy of recommender systems. On the other hand, review texts can be used to explain and justify recommendations [102, 172, 3]. In these applications, aspect-level information can help models to efficiently capture users' preferences and important features (pros and cons) of items. Third, automatic summarization of advantages and disadvantages of a product has become a promising research direction. However, due to the lack of annotated summarization data, it is difficult to train deep learning models in a supervised manner. Several studies have attempted to solve this problem in unsupervised and self-supervised manners [21, 1, 136]. In order to achieve this goal, aspect-specific review segments, i.e., segments that mention certain aspects of products, have to be extracted first. For example, to summarize advantages of a quadcopter, such as its *camera*, we have to extract segments related to *camera* from its reviews. There are also many other applications and tasks, for example, building question-answering systems to answer product-related questions using reviews.

In order to develop machine learning models to tackle challenges in these applications, we have to have a deep understanding of online reviews. In this dissertation, we focus on the *review understanding* problem, and develop machine learning and natural language processing (NLP) tools to understand reviews and learn structured knowledge from unstructured textual reviews.

Generally speaking, the review understanding problem can be studied along three research directions, including understanding a collection of reviews (corpus-level), a single review (document-level), and a piece of review segment (segment-level): (1) *Corpus-level review understanding* focuses on extracting topics from a review corpus (topic modeling), which helps us understand general concerns of customers and features of products. It can benefit many tasks, such as recommendation systems and review summarization [94]. (2) There are several NLP tasks related to *document-level review understanding*. For example, sentiment analysis [103, 77, 104] aims to predict rating scores and opinion polarities of review documents. As a fine-grained sentiment analysis task, document-level multi-aspect sentiment analysis (DMSA) aims to predict opinion polarities and ratings with respect to given aspects [163, 74, 167]. Aspect mention detection aims to discover which aspects have been mentioned in reviews [108]. These tasks can provide us more detailed information about the sentiment of customers and their primary concerns in a review. (3) When talking about *segment-level (sentence-level) review understanding*, we have to discuss the well-known aspect-based sentiment analysis semantic evaluation tasks (including SemEval-2014 Task 4 [110], SemEval-2015 Task 12 [109], and SemEval-2016 Task 5 [108]), which have received extensive attention from both industry and academia. In these tasks, there are many sub-tasks, such as aspect category and opinion polarity classification, and aspect and opinion term extraction, which can help us learn structured knowledge from unstructured textual reviews.

1.2 Research Questions and Hypothesis

The primary research question for this dissertation is: *how to develop novel algorithms that can automatically discover and analyze aspect and opinion related knowledge from a review corpus, so that we can utilize the discovered knowledge to better understand reviews at different levels?* This research question can be further decomposed into the following sub-questions.

- *How can we develop topic models that can automatically discover aspect and opinion knowledge from corpora that consist of short summaries of reviews?*
- *How can we associate the knowledge (i.e., aspect and opinion related topics) discovered by topic models to aspect categories and ratings in automatically collected review corpora with customer-provided aspect ratings, namely DMSA corpora?*
- *How can we incorporate the knowledge discovered by topic models into DMSA models to improve sentiment prediction accuracy?*
- *How can we develop algorithms that can automatically discover aspect and opinion knowledge that is salient to aspect categories and ratings in DMSA corpora?*
- *How can we use the aspect and opinion knowledge discovered by topic models and new algorithms to interpret deep sentiment analysis and DMSA models, and get deep insight into review corpora?*

- *How can we develop algorithms that can automatically discover aspect knowledge and extract aspect specific segments from reviews?*

This thesis covers five problems in the three aforementioned directions (i.e., corpus-level, document-level, and segment-level review understanding) to answer these research questions. Accordingly, the central hypothesis of this research is that *the proposed novel algorithms and methods will make us better understand review corpora with minimal human supervision, and discovered knowledge will benefit both downstream model development and practical applications.*

1.3 Research Issues

Although a large number of machine learning and NLP models have been developed to deal with different problems in review understanding, there are still many challenges: (1) Some review platforms, such as Amazon and Yelp, request customers to write a short summary (in a few words) in each review. From these summaries, we can analyze the primary concerns of customers, however, they have been less investigated. Due to their length, they have limited contextual information and are sparse, noisy and ambiguous. Therefore, it is difficult to apply traditional topic models [6] to extract topics from these short summaries [122]. (2) Another challenge is related to document-level multi-aspect sentiment analysis. In recent few years, many online review platforms, such as Amazon, Best Buy and TripAdvisor, have begun to request users to provide aspect-level feedback. However, recent studies have found that users are less motivated to give aspect-level ratings [163, 167], thus it is difficult to analyze their preference, and it takes a lot of time and effort for human annotators to manually annotate them. Document-level multi-aspect sentiment analysis, which aims to detect aspect mentions from reviews and predict the ratings/sentiment at an individual aspect level, can be used to predict missing aspect-level ratings and analyze review documents. Currently, different multi-task learning frameworks have been developed for this task, however, they suffer from several different problems. For example, they make use of hand-crafted keywords to determine aspects in rating prediction, which makes the model less interpretable and more biased. They do not detect aspect mentions before predicting aspect ratings, which makes the model unreliable. (3) The third challenge is related with aspect-based sentiment analysis (ABSA) semantic evaluation tasks. Although a large number of models have been developed for different sub-tasks in ABSA, most of them are supervised and have been tested only on Restaurant and Laptop corpora, which have been previously annotated [108, 109, 110]. Therefore, their applications have been limited to these domains with a lot of annotated samples. In this dissertation, we focus on dealing with the above mentioned challenges in review understanding. The detailed statement of research issues are presented in the following sections.

1.3.1 Short-Text Topic Modeling for Review Understanding

Being a prevalent form of social communications on the Internet, billions of short texts are generated every day. Discovering knowledge from them has gained a lot of interest from both industry and academia. The short texts have limited contextual information, and they are sparse, noisy and ambiguous, and hence, automatically learning topics from them remains an important challenge. To tackle this problem, we propose a semantics-assisted non-negative matrix factorization (SeaNMF) model to discover topics from short texts. It effectively incorporates the word-context semantic correlations into the model, where the semantic relationships between the words and their contexts are learned from the skip-gram view of a corpus. The SeaNMF model is solved using a block coordinate descent algorithm. We also develop a sparse variant of the SeaNMF model which can achieve a better model interpretability [122].

1.3.2 Multi-aspect Sentiment Analysis for Review Understanding

DMSA for Online Reviews of Medical Experts

In the era of big data, online doctor review platforms, which enable patients to give feedback to their doctors, have become one of the most important components in healthcare systems. On the one hand, they help patients to choose their doctors based on the experience of others. On the other hand, they help doctors to improve the quality of their service. Moreover, they provide important sources for us to discover common concerns of patients and existing problems in clinics, which potentially improve current healthcare systems. In this study, we systematically investigate the dataset from one such review platform, namely, ratemds.com, where each review for a doctor comes with an overall rating and ratings of four different aspects. A comprehensive statistical analysis is conducted first for reviews, ratings, and doctors. Then, we explore the content of reviews by extracting latent topics related to different aspects with unsupervised topic modeling techniques. We also propose a multi-task learning framework for the document-level multi-aspect sentiment classification. This task helps us to not only recover missing aspect-level ratings and detect inconsistent rating scores but also identify aspect-keywords for a given review based on ratings. The proposed model takes both features of doctors and aspect-keywords into consideration [125].

Corpus-level and Concept-based Explanations for Interpretable Document Classification

Using attention weights to identify information that is important for models' decision making is a popular approach to interpret attention-based neural networks. This is commonly realized in practice through the generation of a heat-map for each single document based on attention weights. However, this interpretation method is fragile and it is easy to find contradictory examples. In this study, we propose a corpus-level explanation approach, which aims to capture causal relationships between keywords and model predictions via learning importance of keywords for predicted labels across a training corpus based on attention weights. Based on this idea, we further propose a concept-based explanation method that can automatically learn higher level concepts and their importance to model prediction tasks. Our concept-based explanation method is built upon a novel Abstraction-Aggregation Network, which can automatically cluster important keywords during an end-to-end training process. We apply these methods to the document classification task and show that they are powerful in extracting semantically meaningful keywords and concepts. Our consistency analysis results based on an attention-based Naïve Bayes classifier also demonstrate these keywords and concepts are important for model predictions [128].

Interpretable and Uncertainty Aware Multi-Task Framework for DMSA

In recent years, several online platforms have seen a rapid increase in the number of review systems that request users to provide aspect-level feedback. Document-level Multi-aspect Sentiment Classification (DMSC), where the goal is to predict the ratings/sentiment from a review at an individual aspect level, has become a challenging problem. To tackle this challenge, we propose a deliberate self-attention based deep neural network model, named as FEDAR, for the DMSC problem, which can achieve competitive performance while also being able to interpret the predictions made. As opposed to the previous studies, which make use of hand-crafted keywords to determine aspects in rating predictions, our model does not suffer from human bias issues since aspect keywords are automatically detected through a self-attention mechanism. FEDAR is equipped with a highway word embedding layer to transfer knowledge from pre-trained word embeddings, an RNN encoder layer with output features enriched by pooling and factorization techniques, and a deliberate self-attention layer. In addition, we also propose an attention-driven keywords ranking method, which can automatically discover aspect keywords and aspect-level opinion keywords from a review corpus based on the attention weights. These keywords are significant for rating predictions by FEDAR. Since crowdsourcing annotation can be an alternate way to recover missing ratings of reviews, we propose a lecture-audience strategy to estimate model uncertainty in the context of multi-task learning, so that valuable human resources can focus on the most uncertain predictions [127].

1.3.3 Unsupervised Aspect Detection for Review Understanding

Unsupervised aspect detection aims at automatically extracting interpretable aspects and identifying aspect-specific segments, such as sentences, from online reviews. However, recent deep learning-based topic models, specifically aspect-based autoencoders, suffer from several problems such as extracting noisy aspects and poorly mapping aspects discovered by models to the aspects of interest. To tackle these challenges, we first propose a self-supervised contrastive learning framework and an attention-based model equipped with a novel smooth self-attention module for the aspect detection task in order to learn better representations for aspects and review segments. Secondly, we introduce a high-resolution selective mapping method to efficiently assign aspects discovered by the model to the aspects of interest. We also propose using a knowledge distillation technique to further improve the aspect detection performance [124].

1.4 Dissertation Organization

The remainder of this dissertation is organized as follows.

In Chapter 2, we review literature of short-text topic modeling, multi-aspect sentiment analysis, concept-based model interpretation, uncertainty estimation, and aspect detection.

In Chapter 3, we propose a semantics-assisted non-negative matrix factorization model to discover topics for the short texts. We also develop a sparse variant of the model which can achieve a better model interpretability.

In Chapter 4, we systematically analyze a dataset for document-level multi-aspect sentiment analysis from a healthcare-related review platform, namely, ratemds.com, where each review for a doctor comes with an overall rating and ratings of four different aspects. We also propose a multi-task learning framework for the multi-aspect sentiment classification task.

In Chapter 5, we propose a corpus-level explanation approach, which aims to capture causal relationships between keywords and model predictions via learning importance of keywords for predicted labels across a training corpus based on attention weights, to interpret self-attention based deep document classification models. We further propose a concept-based explanation method that can automatically learn higher level concepts and their importance to the model prediction task.

In Chapter 6, we propose a deliberate self-attention based deep neural network model for the document-level multi-aspect sentiment analysis problem. We further propose an attention-driven keywords ranking method, which can automatically discover aspect keywords and aspect-level opinion keywords from a review corpus based on the attention weights. We also propose a lecture-audience strategy to estimate model uncertainty in the context of multi-task learning.

In Chapter 7, we propose a self-supervised contrastive learning framework and an attention-based model equipped with a novel smooth self-attention module for the unsupervised aspect detection task. We also introduce a high-resolution selective mapping method to efficiently assign aspects discovered by the model to the aspects of interest. In addition, we propose using a knowledge distillation technique to further improve the aspect detection performance.

In Chapter 8, we summarize our study and discuss future directions.

Chapter 2

Literature Review

2.1 Short-Text Topic Modeling

This section reviews related work on short-text topic modeling. Topic modeling for short texts is a challenging research area and many models have been proposed to overcome the lack of contextual information. Most of the current studies are based on the generative probabilistic model, i.e., LDA [6]. Basically, there are three strategies to tackle the problem. The first strategy can capture the cross-document word co-occurrence via aggregating short texts to pseudo-documents. To aggregate the documents, some studies leverage the rich auxiliary contextual information, like authors, time, locations, etc. [51, 152]. For example, in [51], tweets posted by the same user are aggregated into a pseudo-document. However, this method cannot be applied to corpora without auxiliary information. To overcome this disadvantage, another aggregation method is proposed, where the so-called latent pseudo-document is generated using the short texts according to their own topics [179, 112].

The second strategy considers the word semantic information from external corpora, like Wikipedia and Google News [156, 114, 72]. It benefits a lot from the recently developed word embedding approaches based on neural networks [96, 95], which are efficient in uncovering the syntactic and semantic information of the words. For example, Xun *et al.* [156] train the word embeddings upon Wikipedia and use the semantic information as supplementary sources for their topic model. The third strategy directly makes use of word co-occurrence patterns in documents, i.e., short texts. It is also known as the Biterm model [157], since word-pairs co-occurring in the same short text are extracted during the topic modeling. All of the above strategies have been demonstrated to be useful in discovering topics for short texts.

Although the Non-negative Matrix Factorization (NMF) based methods have been successfully applied to topic modeling [19, 20, 59], very few of them are designed to discover topics for short texts. In [158], Yan *et al.* propose a NMF model to learn topics for short texts by directly factorizing a symmetric term correlation matrix. However, since they formulate a quartic non-convex loss function, the algorithm proposed in the work is not reliable nor stable. The recently proposed SymNMF [65, 66] can overcome this problem. However, it does not provide any good intuition for topic modeling. In addition, we cannot get the document representation from SymNMF directly.

2.2 Multi-Aspect Sentiment Analysis

This section reviews related work of document-level multi-aspect sentiment analysis. The sentiment analysis, also known as opinion mining [77], aims to determine the attitude of a person via analyzing polarity (e.g., positive, neutral, or negative) of given text [104, 142]. Document-level sentiment classification is a fundamental problem of sentiment analysis and opinion mining, which intends to determine the sentiment polarity of documents and online reviews. Many recent studies in this field are based on deep neural networks with hierarchical structures [138, 12, 161]. The document-level multi-aspect sentiment classification, which takes aspect categories and ratings into consideration, can be seen as an extension of document-level sentiment classification (single aspect). Early studies on this topic rely on feature engineering to extract features (e.g., n-gram features) corresponding to different aspects and use regression approaches (e.g., Support Vector Regression [130]) to predict multi-aspect ratings [85, 94, 147]. Recently, Yin *et al.* [163] proposed a multi-task learning framework where each aspect is viewed as a task. For each single task, a hierarchical attention module, which includes input encoders and iterative attention modules, has been used to encode documents for classification. This model requires pre-generated pseudo-questions to perform iterative attention and has only been tested on two small-scale datasets[1]. In [74], Li *et al.* proposed incorporating users' information, overall ratings and aspect keywords into their model, which is also based on a multi-task learning framework. However, it is not suitable for our problem, because, in the ratemds dataset, reviews are written anonymously by patients due to privacy concerns. In other words, user information is not available. In addition, overall ratings are calculated by averaging aspect-level ratings, thus we cannot use overall ratings as the input. Zeng *et al.* [167] introduced a variational approach to weakly supervised sentiment analysis. Another area, known as aspect-based sentiment classification [110, 108], is also related to our study. It consists of several fine-grained sentiment classification tasks, including aspect term extraction, aspect term polarity, aspect category detection, and aspect category polarity. There are many research studies in this area [146, 151, 139]. For example, Tang *et al.* [139] introduced a deep memory network for aspect-level sentiment classification. These models usually focus on sentence-level sentiment classification. Moreover, aspect terms, categories, and entities in this problem need to be carefully annotated by human experts.

[1] Both datasets only keep reviews with different aspect-level ratings [69].

2.3 Concept-Based Model Interpretation

This section reviews related work of concept-based model interpretation methods which are beyond per-sample features. Increasing interpretability on machine learning models has become an important topic of research in recent years. Most prior studies [38, 78, 86] focus on interpreting models via feature-based explanations, which alters individual features such as pixels and word-vectors in the form of either deletion [117] or perturbation [137]. However, these methods usually suffer from reliability issues when adversarial perturbations [36] or even simple shifts occur in the input [62]. Moreover, the feature-based approaches explain the model behavior locally [117] for each data sample without a global explanation [58, 37] on how the models make their decisions. In addition, feature-based explanation is not necessarily the most effective way for human understanding.

To alleviate the issues of feature-based explanation models, some research has focused on explaining the model results in the form of high-level human concepts [176, 140, 24, 16, 154, 8, 165]. Unlike assigning the importance scores to individual features, the concept-based methods use the corpus-level concepts as the interpretable units. For instance, the concept "wheels" can be used for detecting the vehicle images and the concept "Olympic Games" for identifying the sports documents. However, most of the existing concept-based approaches require human supervision in providing hand-labeled examples of concepts, which is labor intensive and some human bias can be introduced in the explanation process [58]. Recently, automated concept-based explanation methods [162, 9] are proposed to identify higher-level concepts that are meaningful to humans. However, they have not shown semantically meaningful concepts on text data. In the text classification area, most of the existing approaches focus on improving the classification performance, but ignore the interpretability of the model behaviors [161]. Liu *et al.* [78] utilize the feature attribution method to help users interpret the model behavior. Bouchacourt *et al.* [9] propose a self-interpretable model through unsupervised concept extraction. However, it requires another unsupervised model to extract concepts.

2.4 Uncertainty Estimation

This section reviews related work of uncertainty estimation of deep learning models. Model uncertainty of deep neural networks (NNs) is another research topic related to this study. Bayesian NNs, which learn a distribution over weights, have been studied extensively and achieved competitive results for measuring uncertainty [7, 101, 84]. However, they are often difficult to implement and computationally expensive compared with standard deep NNs. Gal and Ghahramani [32] proposed using Monte Carlo dropout to estimate uncertainty by applying dropout [133] at testing time, which can be interpreted as a Bayesian approximation of the Gaussian process [115]. This method has gained popularity in practice [56, 92] since it is simple to implement and computationally more efficient. Recently, Zhang *et al.* [170] applied dropout-based uncertainty estimation methods to text classification.

2.5 Aspect Detection

This section reviews related work of aspect detection for online reviews. Aspect detection is an important problem of aspect-based sentiment analysis [168, 125]. Existing studies attempt to solve this problem in several different ways, including rule-based, supervised, unsupervised, and weakly supervised approaches. *Rule-based approaches* focus on lexicons and dependency relations, and utilize manually defined rules to identify patterns and extract aspects [111, 81], which require domain-specific knowledge or human expertise. *Supervised approaches* usually formulate aspect extraction as a sequence labeling problem that can be solved by hidden Markov models (HMM) [54], conditional random fields (CRF) [73, 99, 159], and recurrent neural networks (RNN) [149, 80]. These approaches have shown better performance compared to the rule-based ones, but require large amounts of labeled data for training. *Unsupervised approaches* do not need labeled data. Early unsupervised systems are dominated by Latent Dirichlet Allocation (LDA)-based topic models [10, 174, 17, 34, 122, 49]. Wang *et al.* [148] proposed a restricted Boltzmann machine (RBM) model to jointly extract aspects and sentiments. Recently, deep learning based topic models [132, 87, 43] have shown strong performance in extracting coherent aspects. Specifically, aspect-based autoencoder (ABAE) [43] and its variants [87] have also achieved competitive results in detecting aspect-specific segments from reviews. The main challenge is that they need some human effort for aspect mapping. Tulkens *et al.* [141] propose a simple heuristic model that can use nouns in the segment to identify and map aspects, however, it strongly depends on the quality of word embeddings, and its applications have so far been limited to restaurant reviews. *Weakly-supervised approaches* usually leverage aspect seed words as guidance for aspect detection [1, 55, 177] and achieve better performance than unsupervised approaches. However, most of them rely on human annotated data to extract high-quality seed words and are not flexible enough to discover new aspects from a new corpus.

Chapter 3

Short-Text Topic Modeling with SeaNMF

This chapter introduces a semantics-assisted non-negative matrix factorization (SeaNMF) model to discover topics for the short texts. A sparse variant of this model, namely SSeaNMF, which can achieve a better model interpretability, has also been developed. The introduction of this chapter is first presented in Section 3.1. The proposed SeaNMF model, SSeaNMF model and parameter inference solutions are presented in Section 3.2. In Section 3.3, we introduce the datasets used in our experiments, comparison methods, evaluation methods, and implementation details, as well as analyze experimental results. Section 3.4 concludes the discussion of this study.

3.1 Background and Motivation

Every day, large amounts of short texts are generated, such as tweets, search queries, questions, image tags, ad keywords, headlines, and others. They have played an important role in our daily lives. Discovering knowledge from them becomes an interesting yet challenging research task which has gained a lot of attention [131, 152, 173, 157, 51]. Since short texts have only a few words, they can be arbitrary, noisy and ambiguous. All these factors make it difficult to effectively represent short texts and discover knowledge from them. Traditionally, topic modeling has been widely used to automatically uncover the hidden thematic information from the documents with rich content [6, 50, 25]. Generally speaking, there are two groups of topic models, i.e., generative probabilistic models, such as latent Dirichlet allocation (LDA) [6], and non-negative matrix factorization (NMF) [68]. The NMF-based models learn topics by directly decomposing the term-document matrix, which is a bag-of-word matrix representation of a text corpus, into two low-rank factor matrices. The NMF based models have shown outstanding performance in dimension reduction and clustering [20, 66, 64] for high-dimensional data.

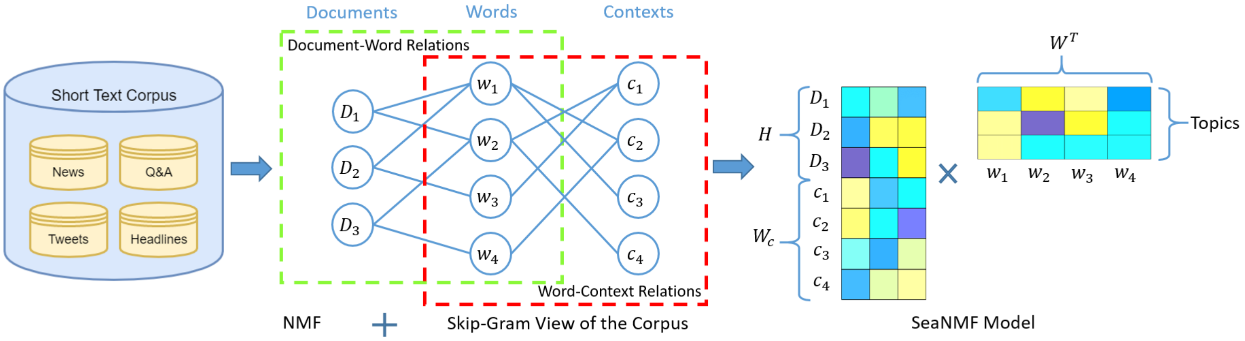


Figure 3.1: The overview of the proposed SeaNMF model for learning topics from short text corpora, which is represented by a bi-relational matrix with both word-document and word-context correlations.

Although the conventional topic models have achieved great success for regular-sized documents, they do not work well on short text collections. Since a short text only contains a few meaningful keywords, the word co-occurrence information is difficult to be captured [173, 51]. In the last few years, many efforts have been dedicated to tackle this challenge. A popular strategy is to aggregate short texts to pseudo-documents and uncover the cross-document word co-occurrence [152, 51, 179, 112]. However, the topics discovered by these models may be biased by the pseudo-documents generated heuristically. More specifically, many irrelevant short texts may be aggregated into the same pseudo-document.

Another strategy is to use the internal semantic relationships of the words to overcome the problem of lacking word co-occurrence. This strategy is proposed due to the fact that semantic information of words can be effectively captured by word embedding techniques, such as word2vec [95] and GloVe [106]. Several attempts [156, 114, 72] have been made to discover topics for short texts via leveraging semantic information of the words from the existing sources, such as the word embeddings based on GoogleNews[1] and Wikipedia[2]. However, since there are many differences between Wikipedia articles and target short texts, such word semantic representations may introduce noise and bias into the topics.

Generally speaking, the word embedding can be useful for short text topic modeling because the words with similar semantic attributes are projected into the same region in the continuous vector space which will improve the clustering performance of the topic models. However, we find another way to boost the performance of the topic models using the skip-gram model with the negative sampling (SGNS). It is well known that SGNS can successfully capture the relationships between a word and its context in a small sliding window [96, 95]. Interestingly, for a short text corpus, each document can naturally be selected as a window. Therefore, the word-context semantic correlations will be effectively captured by SGNS. These correlations can be viewed as an alternative form of the word co-occurrence. It potentially overcomes the problem that arises due to data sparsity.

[1] https://github.com/mmihaltz/word2vec-GoogleNews-vectors

[2] http://nlp.stanford.edu/projects/glove/

There are a few recent studies which show that the SGNS algorithm is equivalent to factorizing a term correlation matrix [70, 71]. Thus, we raise some natural questions: **1)** Can we convert the matrix factorization problem to a non-negative matrix factorization problem? **2)** Can we incorporate this result into the conventional NMF for term-document matrix? **3)** Will the proposed model perform well on discovering topics for short texts? Motivated by these questions, we propose a novel semantics-assisted NMF (SeaNMF) model for short-text topic modeling which is outlined in Fig. 3.1. In this figure, the documents, words and contexts are denoted as D_i, w_i and c_i, respectively. The proposed SeaNMF model can capture the semantics from a short text corpus based on word-document and word-context correlations, and our objective function combines the advantages of both the NMF model for topic modeling and the skip-gram model for capturing word-context semantic correlations. In Fig. 3.1, H, W_c and W are the vector representations of documents, contexts and words in the latent space. Each column of W represents a topic. We use a block coordinate descent algorithm to solve the optimizations. To achieve better interpretability, we also introduce a sparse version of the SeaNMF model.

The proposed models are compared with the other state-of-the-art methods on four real-world short text datasets. The quantitative experiments demonstrate the superiority of our models over several other existing methods in terms of topic coherence and document classification accuracy. The stability and consistency of SeaNMF are verified by parameter sensitivity analysis. Finally, we design an experiment to investigate the interpretability of the SeaNMF model. By visualizing the top keywords of different topics and analyzing their networks, we demonstrate that the topics discovered by SeaNMF are meaningful and their representative keywords are more semantically correlated. Hence, the proposed SeaNMF is an effective topic model for short texts.

3.2 Proposed Methods

In this section, we will first provide some preliminaries along with the block coordinate descent method and its applications in NMF for topic modeling. Then, we will propose our SeaNMF model, and a block-coordinate descent algorithm to estimate latent representations of terms and short documents.

3.2.1 Notations and Preliminaries

The frequently used notations in this section are summarized in Table 3.1.

Table 3.1: Notations used in this study.

Name	**Description**
A	Term-document (word-document) matrix.
S	Word-context (semantic) correlation matrix.
W	Latent factor matrix of words.
W_c	Latent factor matrix of contexts.
H	Latent factor matrix of documents.
$\overrightarrow{w}_j$	Vector representation of word w_j.
$\overrightarrow{c}_j$	Vector representation of context c_j.
$\mathbb{R}_+$	Non-negative real numbers.
N	Number of documents in a corpus.
M	Number of distinct words in the vocabulary.

NMF for Topic Modeling

The NMF method has been successfully applied to topic modeling, due to its superior performance in clustering high-dimensional data [20, 19, 64]. Given a corpus with N documents and M distinct words/terms/keywords in the vocabulary $\mathbb{V}$, we can use a term-document matrix $A \in \mathbb{R}_+^{M\times N}$ to represent it, where $\mathbb{R}_+$ denotes non-negative real numbers. Each column vector $A_{(:,j)} \in \mathbb{R}_+^{M\times 1}$ corresponds to a bag-of-word representation of document j in terms of M keywords. The term-document matrix can be approximated by two lower-rank matrices $W \in \mathbb{R}_+^{M\times K}$ and $H \in \mathbb{R}_+^{N\times K}$, i.e., $A \approx WH^T$, where $K \ll \min(M, N)$ is the number of latent factors (i.e., topics). Usually, this approximation can be formulated as follows:

$$\min_{W,H\geq 0} \|A - WH^T\|_F^2. \tag{3.1}$$

In topic models, the column vector $W_{(:,k)} \in \mathbb{R}_+^{M\times 1}$ represents the k-th topic in terms of M keywords, and its elements are the weights of the corresponding keywords. The row vector $H_{(j,:)} \in \mathbb{R}_+^{1\times K}$ is the latent representation for document j in terms of K topics. Similarly, we can view the row vector $W_{(i,:)} \in \mathbb{R}_+^{1\times K}$ as the latent semantic representation of word i. It is worth mentioning that there are other divergence measures, which can be found in [22].

Problem Statement

Due to the data sparsity, the short texts are too short for the conventional topic models to effectively capture document-level word co-occurrence, which leads to poor performance in topic learning. To tackle this problem, we first investigate the algorithms for estimating the factor matrices in NMF. For example, in the block coordinate descent (BCD) algorithm [60], the updating rules for W and H are shown as follows:

- **Update** W.

$$W_{(:,k)} \leftarrow \left[W_{(:,k)} + \frac{(AH)_{(:,k)} - (WH^TH)_{(:,k)}}{(H^TH)_{(k,k)}} \right]_+ \tag{3.2}$$

- **Update** H.

$$H_{(:,k)} \leftarrow \left[H_{(:,k)} + \frac{(A^TW)_{(:,k)} - (HW^TW)_{(:,k)}}{(W^TW)_{(k,k)}} \right]_+ \tag{3.3}$$

where $[x]_+ = \max(x, 0), \forall x \in \mathbb{R}$.

From the algorithm, we observe that the following lemma holds.

Lemma 1 *For the BCD algorithm, within each iteration:*

1. *The keyword-vector $W^{t+1}_{(i,:)}$ is independent of vector $W^t_{(j,:)}$, when $1 \leq j \neq i \leq M$.*
2. *The document-vector $H^{t+1}_{(i,:)}$ is independent of vector $H^t_{(j,:)}$, when $1 \leq j \neq i \leq N$.*

where t represents the t-th iteration.

Proof *To prove that $W^{t+1}_{(i,:)}$ is independent of $W^t_{(j,:)}$, $\forall j \neq i$, we only need to prove that $(WH^TH)_{(i,k)}$ is independent of $W_{(j,:)}$, $\forall 1 \leq k \leq K$. To simplify the proof, we use a symmetric matrix $B \in \mathbb{R}^{K \times K}_+$ to represent H^TH. Thus, we get $(WH^TH)_{(i,k)} = (WB)_{(i,k)} = W_{(i,:)} \cdot B_{(:,k)}$ which only depends on $W_{(i,:)}$. Hence, $W^{t+1}_{(i,:)}$ is independent of $W^t_{(j,:)}$,$\forall j \neq i$. Similarly, we can also prove that $H^{t+1}_{(i,:)}$ is independent of $H^t_{(j,:)}$.*

We also have the same conclusion for the gradient descent (GD) algorithm. Generally speaking, the relationship between different keywords strongly depends on the documents and vice-versa (see Fig. 3.1). However, due to the data sparsity, i.e., each document has only several keywords, the relationships of keywords are biased by a lot of unrelated documents which results in poor clustering performance. Moreover, the relationships between the keywords and their contexts, i.e., semantic relationships, are not directly discovered by the BCD or GD algorithms in NMF. Therefore, a standard NMF model cannot effectively capture the word co-occurrence for short texts. In this study, we will overcome this drawback by introducing additional dependence of the keywords on their contexts via neural word embedding (see Fig. 3.1).

Neural Word Embedding

Word embedding has been demonstrated to be an effective tool in capturing semantic relationships of the words. Represented by dense vectors, words with similar semantic and syntactic attributes can be found in the same area in the continuous vector space. One of the most successful word embedding methods is proposed by Mikolov et al. [95, 96], known as Skip-Gram with Negative-Sampling (SGNS). The objective function of SGNS is expressed as:

$$\log \sigma(\overrightarrow{w} \cdot \overrightarrow{c}) + \kappa \cdot \mathbb{E}_{c_{neg} \sim p(c)}[\log \sigma(-\overrightarrow{w} \cdot \overrightarrow{c}_{neg})], \tag{3.4}$$

where w and c represent a word and one of its contexts in a sliding window, respectively. $\overrightarrow{w} \in \mathbb{R}^K$ and $\overrightarrow{c} \in \mathbb{R}^K$ are vector representations of them. $\sigma(\overrightarrow{w} \cdot \overrightarrow{c}) = 1/(1 + e^{-\overrightarrow{w} \cdot \overrightarrow{c}})$. c_{neg} is the sampled contexts, known as negative samples, drawn based on a unigram distribution $p(c)$. κ is the number of negative samples.

Recently, Levy et al. [70] have proven that SGNS is equivalent to factorizing a (shifted) word correlation matrix:

$$\overrightarrow{w} \cdot \overrightarrow{c} = \log \left(\frac{\#(w, c) \cdot \mathcal{D}}{\#(w) \cdot \#(c)} \right) - \log \kappa \tag{3.5}$$

where $\#(w, c)$ denotes the number of (w, c) pairs in a corpus. The total number of word-context pairs is $\mathcal{D} = \sum_{w,c \in \mathbb{V}} \#(w, c)$. Similarly, $\#(w) = \sum_{c \in V} \#(w, c)$ and $\#(c) = \sum_{w \in \mathbb{V}} \#(w, c)$ represent the number of times w and c occur in all possible word-context pairs, respectively. $p(c)$ in Eq. (3.4) is expressed as $p(c) = \#(c)/\mathcal{D}$. It is worth mentioning that the $\log((\#(w, c) \cdot \mathcal{D})/(\#(w) \cdot \#(c)))$ is known as the pointwise mutual information (PMI). Therefore, based on this concern, an alternative word representation method was proposed in [70], where the positive constraint is applied to the PMI matrix (PPMI), and then it is factorized by a singular value decomposition method. Eq. (3.5) reveals the internal relationships between the word and its context, which is critical to overcoming the problem of lacking word co-occurrence. In this study, we will leverage the word-context semantic relationships to boost the performance of our models.

3.2.2 The SeaNMF Model

In this section, we propose a novel semantics-assisted NMF (SeaNMF) model to learn topics from the short texts. Our model incorporates the semantic information using the word embeddings into the model training, which enable SeaNMF to recover word co-occurrences from semantic relationships between keywords and their contexts (see Fig. 3.1).

Model Formulation

One challenge of our study is to appropriately introduce the word semantics to NMF. Since the latent matrix $W \in \mathbb{R}_{+}^{M \times K}$ (the elements of W are non-negative), we apply the non-negative constraints on both word and context vectors. Therefore, $\overrightarrow{w} \in \mathbb{R}_{+}^{K}$ and $\overrightarrow{c} \in \mathbb{R}_{+}^{K}$ hold. Given a keyword $w_i \in \mathbb{V}$, we set $W_{(i,:)} = \overrightarrow{w}_i$. To reveal the semantic relationships between the keywords and their context, a matrix W_c is defined for the words in contexts. Thus, $W_c(j, :) = \overrightarrow{c}_j$ for $c_j \in \mathbb{V}$.

With the word and context representations, we can define a semantic (word-context) correlation matrix S which reveals relationships between the keywords and their contexts. Hence, we have

$$S \approx W W_c^T. \tag{3.6}$$

The matrix S can be obtained from the skip-gram view of the corpus. Here, we define each element S_{ij} as follows:

$$S_{ij} = \left[\log \left(\frac{\#(w_i, c_j)}{\#(w_i) \cdot p(c_j)} \right) - \log \kappa \right]_+ , \tag{3.7}$$

where $p(c_j)$ is a unigram distribution for sampling a context c_j. Different from Eq. (3.5), it is defined as

$$p(c_j) = \frac{\#(c_j)^\gamma}{\sum_{c_j \in \mathbb{V}} \#(c_j)^\gamma}, \tag{3.8}$$

where γ is a smoothing factor. It should be noted that S need not necessarily be symmetric. Specifying the sliding windows is a critical component of the skip-gram model. However, for the short texts, this study turns out to be simple. That is, we can naturally view each short document as a window, since each window will have only a few words. Therefore, the total number of windows is equal to the number of documents. Finally, $\#(w_i, c_j)$, $\#(w_i)$, $\#(c_j)$ and $\mathcal{D}$ will be calculated accordingly.

REMARK 1 *The semantic correlation matrix S is not required to be symmetric.*

REMARK 2 *In this study, each short text is viewed as a window. Therefore, the size of each window in the skip-gram model is equal to the length of the corresponding short text. The total number of windows is equal to the number of short texts.*

With the term-document matrix and the semantic correlation matrix, the objective function is expressed as follows:

$$\min_{W, W_c, H \geq 0} \left\| \begin{pmatrix} A^T \\ \sqrt{\alpha} S^T \end{pmatrix} - \begin{pmatrix} H \\ \sqrt{\alpha} W_c \end{pmatrix} W^T \right\|_F^2 + \psi(W, W_c, H), \tag{3.9}$$

where $\alpha \in \mathbb{R}_+$ is a scale parameter. $\psi(W, W_c, H)$ is a penalty function for SeaNMF, which will be specified for a different purpose, such as the sparsity. In this study, we will primarily demonstrate that SeaNMF is an effective topic model for the short texts.

Optimization

Suppose $\psi(W, W_c, H) = 0$, a block coordinate descent (BCD) algorithm can be used to solve Eq. (3.9). We take the derivatives of the objective function with respect to the vectors $W_{(:,k)}$, $W_{c(:,k)}$ and $H_{(:,k)}$. By setting them to zero, we get the updating rules as follows:

- **Update** W

$$\begin{aligned} W_{(:,k)} \leftarrow & [W_{(:,k)} \\ & + \frac{(AH)_{(:,k)} + \alpha(SW_c)_{(:,k)} - (WH^TH)_{(:,k)} - \alpha(WW_c^TW_c)_{(:,k)}}{(H^TH)_{(k,k)} + \alpha(W_c^TW_c)_{(k,k)}}]_+ \end{aligned} \tag{3.10}$$

- **Update** W_c

$$W_{c(:,k)} \leftarrow \left[W_{c(:,k)} + \frac{(S^TW)_{(:,k)} - (W_cW^TW)_{(:,k)}}{(W^TW)_{(k,k)}}\right]_+ \tag{3.11}$$

From lemma 1, the document representation H is independent of W_c and S, therefore, the update rule for H is the same as Eq. (3.3).

Algorithm 1: The SeaNMF Algorithm

```
Input: Term-document matrix $A$;
       Semantic correlation matrix $S$;
       Number of topics $K$, $\alpha$;
Output: $W$, $W_c$, $H$;
Initialize: $W \geq 0$, $W_c \geq 0$, $H \geq 0$ random non-negative real numbers;
$t = 1$;
repeat
    for $k$=1,$K$ do
        Compute $W^t_{(:,k)}$ by Eq. (3.10);
        Compute $W^t_{c(:,k)}$ by Eq. (3.11);
        Compute $H^t_{(:,k)}$ by Eq. (3.3);
    end
    $t = t + 1$;
until Converge;
```

The BCD algorithm for SeaNMF is summarized in Algorithm 1. We first build the term-document matrix A using the bag-of-words representation. Then, we calculate the semantic correlation matrix S by Eq. (3.7). The latent factor matrices W, W_c and H are initialized randomly with non-negative real numbers. Then, within each iteration, their coordinates will be updated column-wise. After each update, $W_{(:,k)}$ and $W_{c(:,k)}$ will be normalized to have a unit ℓ_2-norm. We will repeat this iteration until the algorithm converges.

Intuitive Explanation

We further demonstrate that Eq. (3.10) is equivalent to the following three updating procedures.

$$W^1_{(:,k)} \leftarrow W_{(:,k)} + \frac{(AH)_{(:,k)} - (WH^TH)_{(:,k)}}{(H^TH)_{(k,k)}} \tag{3.12}$$

$$W^2_{(:,k)} \leftarrow W_{(:,k)} + \frac{(SW_c)_{(:,k)} - (WW_c^TW_c)_{(:,k)}}{(W_c^TW_c)_{(k,k)}} \tag{3.13}$$

$$W_{(:,k)} \leftarrow \left[\lambda W^1_{(:,k)} + (1-\lambda)W^2_{(:,k)}\right]_+ \tag{3.14}$$

where $\lambda = \frac{(H^TH)_{(k,k)}}{(H^TH)_{(k,k)}+\alpha(W_c^TW_c)_{(k,k)}} \in [0,1]$. □

As we can see, Eq. (3.12) is the same as Eq. (3.2) for the standard NMF. It tries to project the words in the same documents into the same region of the space using the term-document matrix. On the other hand, Eq. (3.13) tries to move the words close to each other if they share the common context keywords. Therefore, it increases the coherence of the topics. For example, in Fig. 3.1, w_1 and w_4 do not appear in the same document. However, since they both have w_2 as context keyword, they may be semantically correlated. Take two short texts “iphone ios system” and “galaxy android system” as an example. “iphone” and “ios” do not appear in the second sentence, and “galaxy” and “android” do not appear in the first sentence. Thus, the correlations between “iphone, ios” and “galaxy, android” are minor in the standard NMF. However, in SeaNMF, the correlations are enhanced by Eq. (3.13) using the fact that they share the common keyword “system”. The overall updating procedure, given in Eq. (3.14), is a linear combination of Eq. (3.12) and (3.13) which guarantees the top keywords in each topic are highly correlated.

Computational Complexity

We have noticed that the proposed SeaNMF model maintains the same formation (Eq. (3.9)) as that of the standard NMF (Eq. (3.1)), therefore, its computational complexity is $O((M+N)MK)$ within a single iteration of updating factor matrices. Since for short text corpora, the number of keywords is usually less than the number of documents, i.e, $M < N$, we have $M+N < 2N$. Therefore, the computational complexity of SeaNMF for short texts is reduced to $O(NMK)$, which is the same as that of standard NMF [60]. However, due to data sparsity for short texts, this complexity can be further reduced. From Eqs. (3.10), (3.11) and (3.3), we can see the complexity is dominated by the calculations of AH, SW_c, A^TW. Without considering the sparsity, their computational costs are $O(MNK)$, $O(MMK)$, and $O(NMK)$, respectively. However, since A and S are sparse matrices, which can be seen in Table 3.2, we only need to multiply the non-zero elements with factor matrices. Suppose the numbers of non-zero elements in A and S are z_A and z_S, the complexity of calculating AH, SW_c, and A^TW will be $O(z_AK)$, $O(z_SK)$, and $O(z_AK)$, respectively. Therefore, the proposed SeaNMF model has the complexity of $O(\max(z_A,z_S)K)$, where $\max(z_A,z_S) \ll NM$ and $K \ll \min(N,M)$, which is much cheaper than the standard NMF.

3.2.3 The Sparse SeaNMF Model

In standard topic models, words are represented by dense vectors in a continuous real space. Specifically, in SeaNMF, we use the low-rank factor matrix W to encode the words. Introducing sparsity to W will reduce the active components of the word vectors, which will make it easy to interpret the topics.

Considering a better interpretability of the model, we introduce the Sparse SeaNMF (SSeaNMF) model, where we apply the sparsity constraint to W and express the penalty function as follows:

$$\psi(W, W_c, H) = \beta \|W\|_1^2, \tag{3.15}$$

where $\|\cdot\|_1$ represents the ℓ_1-norm. Since the sparsity is only applied to W, the BCD algorithm for updating W is modified to

$$W_{(:,k)} \leftarrow [W_{(:,k)} + \frac{(AH)_{(:,k)} + \alpha(SW_c)_{(:,k)} - (WH^TH)_{(:,k)} - \alpha(WW_c^TW_c)_{(:,k)} + \beta \cdot 1_K}{(H^TH)_{(k,k)} + \alpha(W_c^TW_c)_{(k,k)} + \beta}]_+ \tag{3.16}$$

where $1_K \in \mathbb{R}^{M\times 1}$ and $1_{K(i,:)} = -\sum_{k=1}^{K} W_{(i,k)}, \forall 1 \leq i \leq M$.

Updating procedures for W_c and H remain the same as in Eq. (3.11) and Eq. (3.3), respectively. Compared with standard SeaNMF, calculating 1_K will not significantly increase the computational complexity of the algorithm.

3.3 Experiments

In this section, we will demonstrate the promising performance of our models by conducting extensive experiments on different real-world datasets. We will introduce the datasets, evaluation metrics and baseline methods, and then explain different sets of results.

3.3.1 Datasets Used and Evaluation Metrics

Our experiments are carried out on four real-world short text datasets corresponding to four types of applications, i.e., News, Questions&Answers, Microblogs and Article headlines.

- **Tag.News**. This data set is a part of the TagMyNews dataset[3], which is composed of news, snippets and tweets. After removing the stopwords, we only keep the news with at most 25 keywords. The articles in the dataset belong to one of the following 7 categories: Business, Entertainment, Health, Sci&Tech, Sport, US and World.

[3] http://acube.di.unipi.it/datasets/

Table 3.2: Basic statistics of the datasets used in this study.

Data Set	#docs	#terms	density(A)	density(S)	doc-length	#cats
Tag.News	28658	11525	1.2861%	0.1369%	18.14	7
Yahoo.Ans	40754	4334	0.1997%	0.0973%	4.30	10
Tweets	43413	10279	0.2744%	0.0713%	7.73	15
DBLP	15001	2447	0.7693%	0.2677%	6.64	4
Yahoo.CA	30686	4334	5.0532%	0.7754%	42.61	-
ACM.IS	36392	2447	4.2667%	1.9494%	77.49	-

- **Yahoo.Ans**. This dataset is a subset extracted from the Yahoo! Answers Manner Questions, version 2.0[4]. In our dataset, we collect the subjects of the Questions from 10 different categories, including Financial Service, Diet&Fitness, etc.
- **Tweets**. The original Tweets dataset is collected and labeled by Zubiaga et al. [178]. We select 15 different categories from the dataset, i.e., Arts, Business, Computers, Games, Health, Home, News, Recreation, Reference, Regional, Science, Shopping, Society, Sports and World. For each category, we sample 2500∼3000 distinct tweets with at least two keywords.
- **DBLP**. The raw DBLP dataset is available at `http://dblp.uni-trier.de`. In our dataset, we collect the titles of the conference papers from the following 4 categories: Machine Learning, Data Mining, Information Retrieval and Database.

Some basic statistics of these datasets are shown in Table 3.2. In this table, '#docs' represents the number of documents in each dataset. '#terms' is the number of keywords in the vocabulary. 'density' is defined as $\frac{\#\text{non-zero}}{\#\text{docs}\cdot\#\text{terms}}$, where #non-zero is the number of non-zero elements in the matrix. The 'density(A)' and 'density(S)' represent the density of term-document matrix (A) and semantic correlation matrix (S), respectively. 'doc-length' represents the average length of the documents. '#cats' denotes the number of distinct categories.

In our experiments, we also leverage the following two datasets as external sources in the evaluations. It should be noted that they are NOT used to train the models.

- **Yahoo.CA**. From the Yahoo! Answers Manner Questions, version 2.0, we collect the content and best answer for each question, and construct a new regular-sized document set, namely, Yahoo.CA.
- **ACM.IS**. This dataset is part of the ACM IS abstract dataset[5], which contains the abstracts of ACM information system papers published between 2002 and 2011.

In order to train GPUDMM [72], we also obtain **GoogleNews(300d)** from `https://github.com/mmihaltz/word2vec-GoogleNews-vectors`. It contains 3 million English words which

[4] `https://webscope.sandbox.yahoo.com/catalog.php?datatype=l`

[5] `https://dataverse.harvard.edu/dataset.xhtml?persistentId=doi:10.7910/DVN/27695`

are embedded into 300 dimensional latent space by performing the word2vec model [96] on the Google News corpus which consists of 3 billion running words.

In this study, we will use the topic coherence and document classification accuracy for our evaluation.
Topic Coherence. Given a topic k, the PMI score is calculated by the following equation:

$$C_k = \frac{2}{\mathcal{N}(\mathcal{N}-1)} \sum_{1 \leq i < j \leq \mathcal{N}} \log \frac{p(w_i, w_j)}{p(w_i)p(w_j)} \tag{3.17}$$

where $\mathcal{N}$ is the number of most probable words in this topic.
$p(w_i, w_j) = \#(w_i, w_j)/\mathcal{D}$ is the probability of the words w_i and w_j co-occurring in the same document. $p(w_i) = \#(w_i)/\mathcal{D}$ and $p(w_j) = \#(w_j)/\mathcal{D}$ are the marginal probabilities. The average PMI score over all the topics will be used to evaluate the quality of the topic models. However, Quan et al. [112] have shown that the average PMI score, that works well for regular-sized documents, is still problematic for short texts, which means a gold-standard topic may be assigned with a low PMI score.

In this study, we leverage the following strategy to overcome this problem. First, we calculate the PMI score based on the four short text datasets as usual. Second, for the Yahoo.Ans and DBLP datasets, we calculate the PMI score based on the external corpora, i.e., Yahoo.CA and ACM.IS, which are composed of regular documents. The results in both experiments will be used to demonstrate the effectiveness of our models. We emphasize that Yahoo.CA and ACM.IS do not participate in the training of our models.

In our experiments, we set $\mathcal{N} = 10$. It also should be noted that the difference between Eq. (3.17) and the PMI score used in [179] is that we do not consider the co-occurrence of the same word.
Document Classification. Another popular way to evaluate the effectiveness of the topic models is to leverage the latent document representations for external tasks. In our experiments, we will conduct short text classification on all the datasets whose documents have been labeled. A five-fold cross validation is used to evaluate the performance of the classification, where each corpus is randomly split into training and testing sets with a ratio of 4 : 1. Then, the documents are classified by the LIBLINEAR package[6] [27].

Finally, the quality of the classification is measured by average precision, recall and F-score.

3.3.2 Comparison Methods

We compare the performance of our models with the following state-of-the-art methods.

- **Latent Dirichlet Allocation (LDA)**. LDA [6] is a well-known baseline method in the topic modeling which performs well on the regular-sized documents. In this study, we use

[6] `https://www.csie.ntu.edu.tw/~cjlin/liblinear/`

a Python implementation[7] of LDA with a collapsed Gibbs sampling.

- **Non-negative Matrix Factorization (NMF)**. NMF [60] is an unsupervised method that can perform dimension reduction and clustering simultaneously. It has found applications in a range of areas, including topic modeling. In our experiments, the NMF[8] is implemented in Python with a block coordinate descent algorithm.
- **Pseudo-document-based Topic Model (PTM)**. PTM [179] introduces *pseudo-documents* into the topic model, which implicitly aggregates short texts without auxiliary information. It is one of the most recent methods for discovering topics from short text corpora.
- **GPUDMM**. The GPUDMM [72] for short-text topic modeling is based on the Dirichlet Multinomial Mixture model. During the sampling process using the generalized Pólya urn model, it promotes the semantically related words in each topic by leveraging the external word semantic knowledge, i.e., word vectors, from very large corpora. In this study, we will use the Google News (300d) dataset as the external resource.

In our experiments, the default number of topics is set to $K = 100$. For LDA, we set parameters $\alpha = 0.1$ and $\beta = 0.01$, since weak prior can give a better performance for short texts [179]. For PTM and GPUDMM, we use the default hyper-parameter settings. Specifically, we set parameters $\alpha = 0.1$, $\lambda = 0.1$ and $\beta = 0.01$ for PTM. For GPUDMM, we set parameters $\beta = 0.1$. In LDA, PTM and GPUDMM, Gibbs sampling is run for 2000 iterations. For SeaNMF, we set $\alpha = 1.0$ for Tag.News and Tweets and $\alpha = 0.1$ for Yahoo.Ans and DBLP. To calculate S, we set $\kappa = 1.0$ and $\gamma = 1.0$. In SSeaNMF, we set $\beta = 0.1$. We also set the seed for the random number generator to 0 for NMF, SeaNMF and SSeaNMF to make sure the results are consistent and independent of random initial states. The codes for SeaNMF has been publicly available at `https://github.com/tshi04/SeaNMF`.

3.3.3 Results

Topic Coherence Results

We first present the topic coherence results of our models and other comparison methods in Tables 3.3 and 3.4. We use the bold font to show the best performance values and the underlining to highlight the second best values.

From Table 3.3, we observe that our models outperform the standard NMF, which indicates that SeaNMF is effective for learning topics from short texts. Compared with LDA and recent PTM, SeaNMF shows significant improvements, which implies that our models discover more coherent topics. To better understand the poor performance of GPUDMM in all cases, we visualize the top keywords in each topic, where we find that many top keywords (e.g. ‘extraction’, ‘extracting’ and ‘extract’) are semantically correlated, but they do not tend to

[7] `https://github.com/shuyo/iir/tree/master/lda`

[8] `https://github.com/kimjingu/nonnegfac-python`

Table 3.3: Topic coherence results in terms of PMI.

	Tag.News	Yahoo.Ans	Tweets	DBLP
LDA	1.5048	1.2957	1.1637	0.9346
NMF	1.6414	1.1394	1.8045	0.9184
PTM	1.6628	1.1311	1.3745	0.8505
GPUDMM	0.9751	0.5798	0.9213	0.2815
SeaNMF	**3.6318**	**1.7553**	4.1477	1.6137
SSeaNMF	3.6053	1.6081	**4.1979**	**1.6239**

Table 3.4: Topic coherence results with Yahoo.CA and ACM.IS.

	Yahoo.Ans/Yahoo.CA	DBLP/ACM.IS
LDA	0.6540	0.4282
NMF	0.5261	0.3626
PTM	0.6504	0.4431
GPUDMM	0.3302	-0.0159
SeaNMF	**1.1094**	0.6641
SSeaNMF	1.0188	**0.6447**

appear in the same document. Another possible reason is that the word semantic relationships in Google News and other datasets are different, so that the general semantics knowledge from Google News may not work well on discovering topics from these datasets.

As discussed in the topic coherence section, since the PMI scores are problematic for short texts, we also evaluate topic coherence based on external corpora which are composed of long documents. After training different models on Yahoo.Ans, we extract the top keywords from each topic, and then calculate the PMI scores based on the Yahoo.CA corpus. Similarly, for DBLP, the PMI scores are calculated based on the ACM.IS dataset. The results obtained on these external corpora are presented in Table 3.4. From the table, we find that SeaNMF outperforms the other baseline methods. Therefore, from our topic coherence results, we demonstrate that by leveraging the word semantic correlations, SeaNMF can capture more coherent topics from short texts.

Document Classification Results

In addition to the topic coherence, we also compared the document classification performance of different methods. As we can see from Tables 3.5 and 3.6, both the best and the second best results are achieved by our models on Tag.News, Yahoo.Ans, and Tweets. This demonstrates that our models are effective in the document classification for short texts. Compared with the conventional topic models, such as LDA and NMF, SeaNMF has a significant improvement in terms of different classification measures. The SeaNMF models also perform better than

Table 3.5: Performance comparison of various methods on document classification.

	Tag.News			Yahoo.Ans		
	Precision	Recall	F-score	Precision	Recall	F-score
LDA	0.7323	0.7184	0.7239	0.5929	0.5738	0.5659
NMF	0.6763	0.6371	0.6507	0.6303	0.5470	0.5706
PTM	0.7525	0.7396	0.7444	0.6390	0.6038	0.6026
GPUDMM	0.7843	0.7712	0.7760	0.5954	0.6308	0.5995
SeaNMF	0.7868	0.7786	0.7821	0.6566	0.6338	0.6366
SSeaNMF	**0.7894**	**0.7801**	**0.7841**	**0.6603**	**0.6369**	**0.6401**

Table 3.6: Performance comparison of various methods on document classification.

	Tweets			DBLP		
	Precision	Recall	F-score	Precision	Recall	F-score
LDA	0.3827	0.3867	0.3758	0.6081	0.5973	0.5994
NMF	0.3677	0.3517	0.3506	0.6393	0.6226	0.6273
PTM	0.3941	0.3838	0.3786	0.6424	0.6367	0.6379
GPUDMM	0.3985	0.4066	0.3903	0.6670	0.6573	0.6586
SeaNMF	**0.4648**	0.4555	**0.4527**	0.6648	0.6552	0.6575
SSeaNMF	0.4592	**0.4568**	0.4516	**0.6700**	**0.6613**	**0.6636**

PTM, which attempts to capture the cross-document word correlations by aggregating similar short texts into pseudo documents. This comparison demonstrates that the word correlations obtained from the skip-gram view of a corpus play an important role in capturing high quality semantics, given the performance of standard NMF is not as good as that of LDA. In Tables 3.5 and 3.6, we also observe that the GPUDMM model performs better than the other baseline methods. The difference between GPUDMM and SeaNMF is that GPUDMM explicitly makes use of the term correlations obtained from the pre-trained word representations on external large corpora, while SeaNMF is only based on the short text corpus itself. Thus, given an external resource, like Google News, the performance of GPUDMM cannot be guaranteed across different short texts. In summary, the classification results have shown that SeaNMF is a superior topic model for short texts, even without using the auxiliary information or external sources, or aggregating the short texts.

It should be noted that the results based on the Tweets dataset are more reliable because the number of tweets in different categories is almost the same, which avoids the problems caused by the so-called ‘imbalanced classes’. As we can see in Tables 3.5 and 3.6, SeaNMF has on an average more than 12% improvement over the other baseline methods with respect to precision, recall, and F-score.

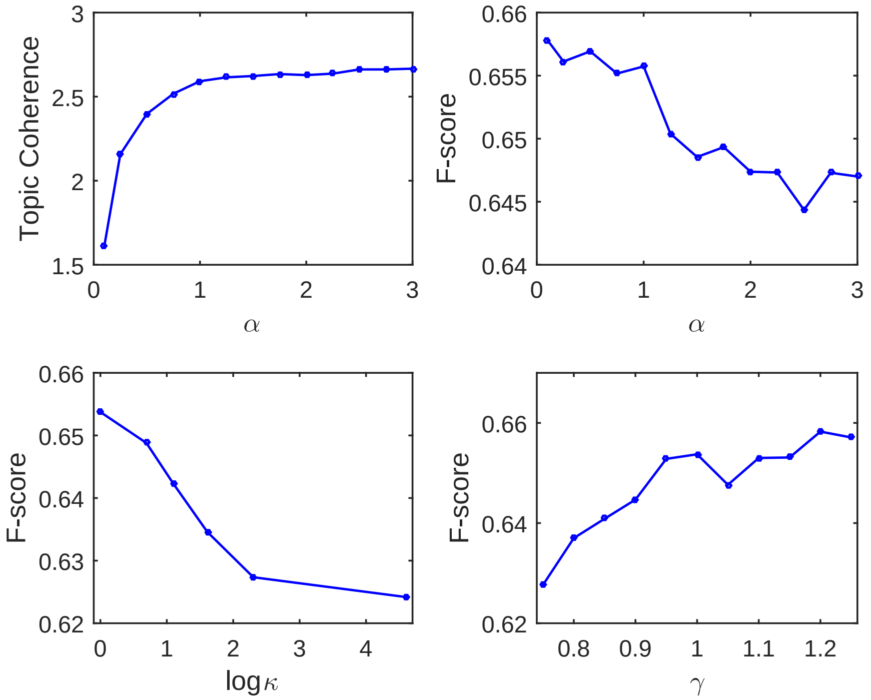


Figure 3.2: Topic coherence and classification performance by varying α, κ, and γ.

3.3.4 Parameter Sensitivity

In this section, we will demonstrate the stability and consistency of SeaNMF by varying the parameters α, κ, and γ.

The parameter α is the weight for factorizing the word semantic correlation matrix. Here, we study the effects of α on the topic coherence and classification accuracy on DBLP. It can be seen from Fig. 3.2 that the topic coherence increases rapidly as we increase the weight when $\alpha \in (0, 1]$. However, it stays almost constant after $\alpha > 1$. This clearly shows that SeaNMF is effective for short texts just because it leverages the word semantic correlations.

We also observe that a better topic coherence does not imply better document classification performance. As we can see in Fig. 3.2, the F-score decreases as α increases. Therefore, for a short text collection, a highly coherent topic is not the same as a high quality topic which is consistent with the findings of others in the literature [112]. We also notice that the F-score does not significantly change with α, i.e., the change is less than 0.02. Hence, SeaNMF is a stable topic model for short texts.

The parameters κ and α play an important role in constructing the semantic correlation matrix S. κ affects the sparsity of S. Large κ leads to very sparse S, which implies that the words are less correlated. As shown in Fig. 3.2, the F-score is reduced when we increase κ. γ

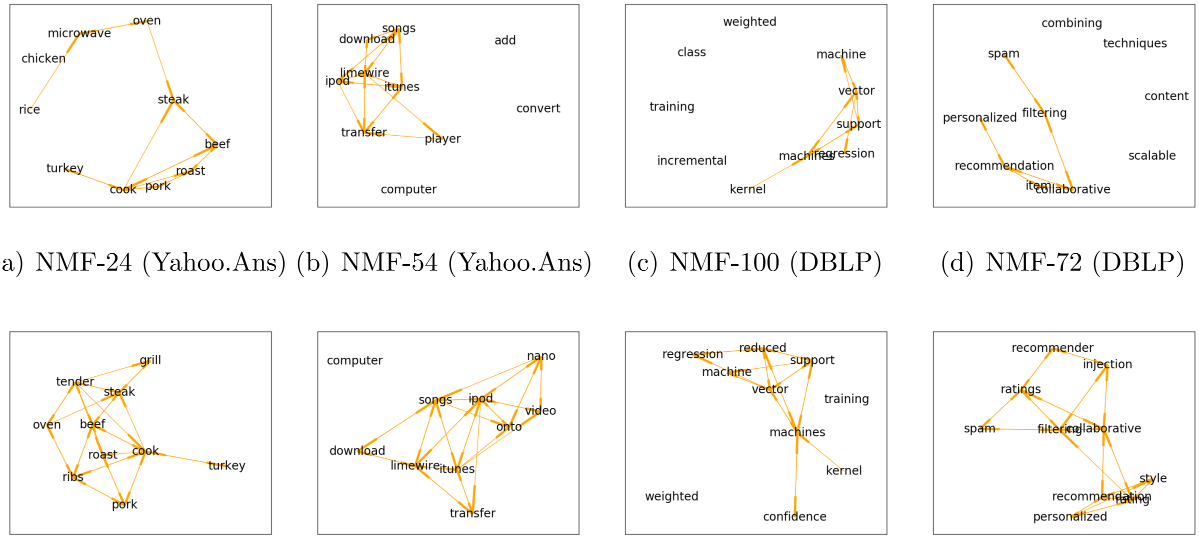
Table 3.7: Discovered topics by the proposed method. The word is colored in red if its degree is less than 2. The numbers in the parentheses represent the frequency of the word in the corpus. NMF-k corresponds to the k-th topic discovered by the NMF model.

	Yahoo.Ans			
Category	Cooking and Recipes		Blues	
	NMF-24	SeaNMF-47	NMF-54	SeaNMF-50
PMI	2.7291	3.1713	2.6674	3.3517
Top-10 keywords	cook(381)	cook(381)	songs(257)	songs(257)
	chicken(168)	roast(54)	ipod(143)	ipod(143)
	turkey(72)	oven(67)	download(179)	computer(216)
	roast(54)	pork(40)	computer(216)	download(179)
	rice(80)	beef(56)	itunes(54)	transfer(75)
	oven(67)	grill(50)	player(94)	onto(51)
	beef(56)	turkey(72)	limewire(70)	itunes(54)
	pork(40)	steak(50)	transfer(75)	limewire(70)
	steak(50)	tender(11)	add(138)	video(71)
	microwave(51)	ribs(16)	convert(118)	nano(31)

Table 3.8: Discovered topics by the proposed method. The word is colored in red if its degree is less than 2. The numbers in the parentheses represent the frequency of the word in the corpus. NMF-k corresponds to the k-th topic discovered by the NMF model.

	DBLP			
Category	Machine Learning		Data Mining	
	NMF-100	SeaNMF-45	NMF-72	SeaNMF-98
PMI	1.4570	1.7215	1.2636	1.9810
Top-10 keywords	support(228)	support(228)	filtering(147)	filtering(147)
	vector(150)	vector(150)	collaborative(122)	collaborative(122)
	machines(95)	machines(95)	content(166)	recommendation(47)
	machine(116)	machine(116)	scalable(130)	personalized(62)
	regression(127)	regression(127)	combining(118)	spam(37)
	class(104)	kernel(151)	spam(37)	recommender(27)
	training(79)	training(79)	recommendation(47)	injection(5)
	kernel(151)	confidence(19)	personalized(62)	style(15)
	incremental(105)	reduced(5)	item(29)	rating(8)
	weighted(67)	weighted(67)	techniques(115)	ratings(6)

is a smoothing factor for the probability of sampling a context. From the figure, the F-score is slightly improved when γ is increased. To summarize, both parameters affect the quality of topics by changing the semantic correlation matrix. It implies that the word semantic correlations are critical to SeaNMF.



(a) NMF-24 (Yahoo.Ans) (b) NMF-54 (Yahoo.Ans) (c) NMF-100 (DBLP) (d) NMF-72 (DBLP)

(e) SeaNMF-47 (f) SeaNMF-50 (g) SeaNMF-45 (h) SeaNMF-98

Figure 3.3: Network visualizations of the keywords obtained by the NMF and SeaNMF models on Yahoo.Ans and DBLP datasets.

3.3.5 Semantic Analysis of Topics

In this section, we show that the topics discovered by SeaNMF are meaningful by visualizing the top keywords. They will be compared with the top keywords given by the standard NMF method.

After training the NMF model on the Yahoo.Ans and DBLP datasets, we select the topics with high PMI scores. Then, we find the most similar topic obtained from SeaNMF for each of them based on the top keywords. The lists of the top keywords in the selected topics obtained are shown in Tables 3.7 and 3.8. As we can see, two topics for Yahoo.Ans are about cooking and the technical problems on downloading or transferring songs. The two topics selected from DBLP are on publications related with machine learning and data mining.

To demonstrate the topics discovered by SeaNMF are more semantically correlated, we use the selected top keywords in each topic to construct the word networks. More specifically, suppose the top keyword list is denoted as $\{w_i\}_{i=1}^{10}$, we first find the 30 most correlated words $\{v_j\}_{j=1}^{30}$ for each keyword w_{i_0} based on the positive PMI matrix. If a keyword $w_{i_1} \in \{w_i\} \cap \{v_j\}$, $i_1 \neq i_0$, we draw an edge from w_{i_0} to w_{i_1}.

As we can see from Fig. 3.3, all the graphs for the standard NMF model are very sparse. Some keywords with higher frequency in the corpus have lower degree which means that they are less correlated with the other words. For example, the frequency of ‘chicken’ is high, however, its most correlated words do not contain the other keywords and it is not in the most correlated word lists of the other keywords. In the standard topic modeling, these keywords might be viewed as noise. In Tables 3.7 and 3.8, keywords with degree less than two are colored in red. We can see that the topics obtained from the standard NMF model are noisy. On the other hand, we conduct the same experiments on our SeaNMF model. From Table 3.7, Table 3.8 and Fig. 3.3, we can see that topics discovered by our SeaNMF model have less noisy words and the top keywords are more correlated. Therefore, these semantic analysis results demonstrate that the SeaNMF model can discover meaningful and consistent topics for short texts.

3.4 Summary

In this study, we introduce a semantics-assisted NMF (SeaNMF) model to discover topics for short text corpora. The proposed model leverages the word-context semantic correlations in the training, which potentially overcomes the problem of lacking context that arises due to the data sparsity. The semantic correlations between the words and their contexts are learned from the skip-gram view of corpora, which was demonstrated to be effective for revealing word semantic relationships. We use a block coordinate descent algorithm to solve our SeaNMF model. To achieve a better model interpretability, a sparse SeaNMF model is also developed. We compared the performance of our models with several other state-of-the-art methods on four real-world short text datasets. The quantitative evaluations demonstrate that our models outperform other methods with respect to widely used metrics such as the topic coherence and document classification accuracy. The parameter sensitivity results demonstrate the stability and consistency of the performance of our SeaNMF model. The qualitative results show that the topics discovered by SeaNMF are meaningful and their top keywords are more semantically correlated. Hence, we conclude that the proposed SeaNMF is an effective topic model for short texts.

Chapter 4

Multi-Aspect Sentiment Analysis for Online Reviews of Medical Experts

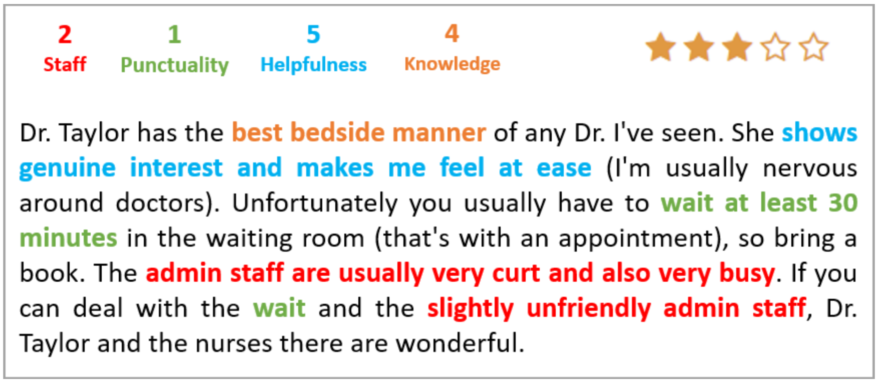
This chapter presents a new dataset in the healthcare domain, i.e., RateMDs, for the document-level multi-aspect sentiment analysis. Based on this dataset, we conduct a comprehensive statistical analysis, explore aspect related keywords, and develop a multi-task learning framework to predict aspect-level ratings. First, the introduction of this chapter is presented in Section 4.1. Section 4.2 provides detailed analysis for the RateMDs dataset. The proposed multi-task learning model is presented in Section 4.3. In Section 4.4, we introduce the datasets used in our experiments, baseline methods, and implementation details, as well as analyze experimental results. Section 4.5 concludes this study.

4.1 Background and Motivation

Healthcare systems are evolving rapidly due to advancements in recent artificial intelligence techniques, especially deep learning frameworks [98, 129]. A number of automated tools and ML driven micro-services in healthcare, e.g., medical imaging diagnosis for diabetic eye disease [41] and cancer [82], have gained attention from both industry and academia. Online doctor review systems, such as ratemds[1] and zocdoc[2], establish a unique environment for patients to give feedback to their doctors. These reviews are evolving into an important source for evaluating performance of doctors in medical practices as a supplement to their professional knowledge. For example, ratemds is one such review platform for doctors and facilities (e.g., hospitals or clinics), which has more than two million healthcare providers (i.e., doctors and facilities) and three million reviews. On their website, a patient can anonymously post a review along with an overall rating and ratings from four different aspects to their doctors,

[1] https://www.ratemds.com/

[2] https://www.zocdoc.com/



2	1	5	4	
Staff	Punctuality	Helpfulness	Knowledge	★★★☆☆

Dr. Taylor has the **best bedside manner** of any Dr. I've seen. She **shows genuine interest and makes me feel at ease** (I'm usually nervous around doctors). Unfortunately you usually have to **wait at least 30 minutes** in the waiting room (that's with an appointment), so bring a book. The **admin staff are usually very curt and also very busy**. If you can deal with the **wait** and the **slightly unfriendly admin staff**, Dr. Taylor and the nurses there are wonderful.

Figure 4.1: An example of ratemds reviews. Keywords corresponding to different aspects are highlighted with different colors.

i.e., staff, punctuality, helpfulness and knowledge. Similarly, patients can also review and rate facilities. Fig. 4.1 shows an example of doctor reviews. In this figure, there is a plain-text review with four aspect-level ratings, in which staff and punctuality refer to front-desks and appointments, respectively, while helpfulness and knowledge are about bedside manners of doctors and medical procedures. Generally speaking, these reviews sketch more detailed profiles of doctors in medical practices, so they can not only help other patients to find better options, but also help doctors to improve their service quality.

Nowadays, different knowledge discovery and opinion mining techniques allow us to find out general needs of patients and existing problems in clinics from a large number of online reviews, which helps to improve current healthcare systems. Many of these techniques, including graphical models [94], regression approaches [147] and deep learning methods [168, 69, 163], have been successfully applied to similar online review systems in other domains, such as BeerAdvocate[3], and TripAdvisor[4]. However, online doctor review systems, which are primary platforms for patients to give feedback, have not been sufficiently investigated before [39, 52, 11], especially for those systems which evaluate doctors' medical practices from different aspects. There are many tasks associated with this type of data. For example, many patients are less motivated to give aspect-level ratings and some ratings are inconsistent with reviews. Can we predict rating scores based on plain-text reviews to recover missing values and correct inconsistent ratings? On the other hand, given aspect categories and aspect-level ratings, can we use these ratings as a form of weak supervision to obtain keywords corresponding to different categories? Alternately, can we use unsupervised methods to discover cluster structures in latent space for keywords in reviews and associate them with different aspects? Although sophisticated models have been proposed for these tasks, they have only been applied to other types of datasets [74] and some of them have only been tested on small-scale datasets [163, 69]. In this study, we first thoroughly explore the ratemds

[3] https://www.beeradvocate.com/
[4] https://www.tripadvisor.com/

dataset, and then, due to the strong correlations of multi-aspect ratings, we formulate a multi-task learning model to predict ratings and detect aspect-keywords in each review with attention mechanism [4, 88]. Our contributions can be summarized as follows:

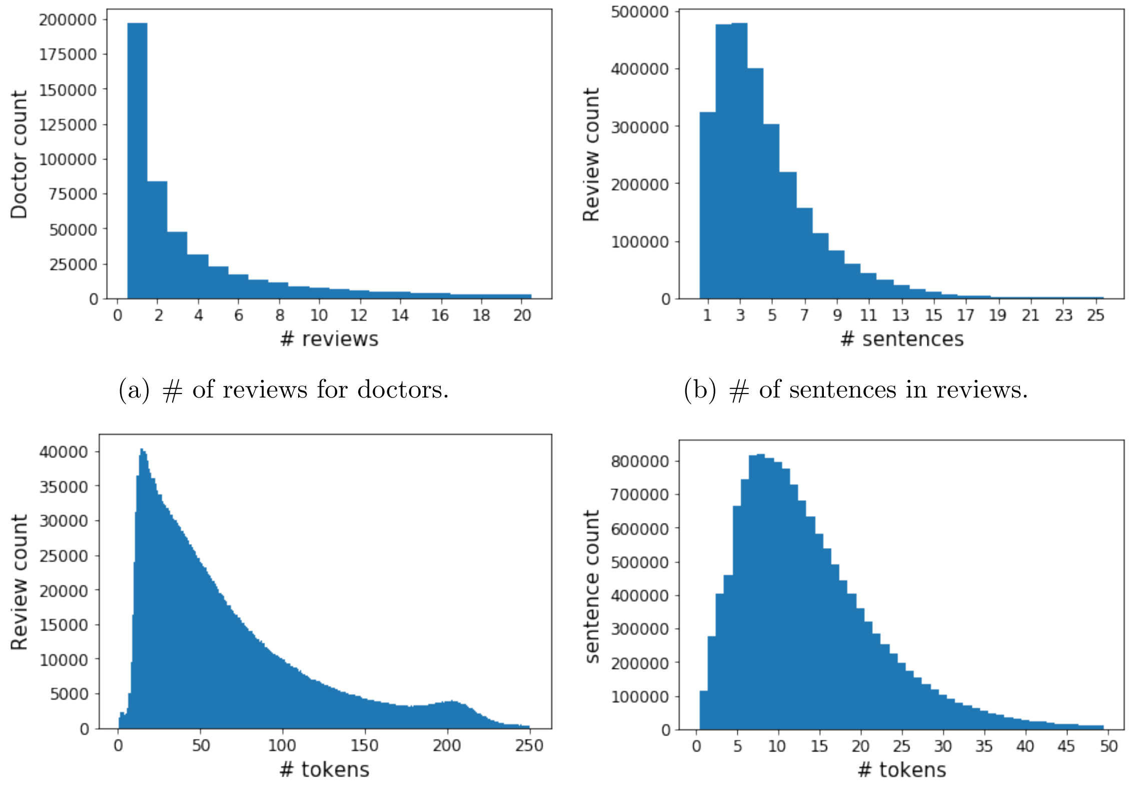
- Propose a multi-task learning framework, which takes features of doctors and aspect-keywords discovered by the topic model into consideration, for the document-level multi-aspect sentiment classification task and conduct extensive experiments on two subsets of the ratemds dataset.
- Introduce a new dataset which consists of more than two million reviews with multi-aspect ratings. Different from datasets for commercial products and entertainment (like BeerAdvocate and TripAdvisor), this dataset is healthcare related and an important source for studying general concerns of patients and existing problems in clinics.
- Conduct a comprehensive statistical analysis on this dataset, including statistics of reviews, ratings and doctors. We also explore aspect-keywords of reviews with a topic model [6].

4.2 Preliminary Data Analysis

In this section, we first conduct data analysis of reviews, ratings and doctors to get a comprehensive understanding of key features that can be useful for document-level multi-aspect sentiment classification. To gain deeper insights into the content of reviews, we also use topic models to discover aspect-keywords from latent topics.

4.2.1 Overview

The ratemds dataset was obtained from the `ratemds.com` website, which has records (e.g., specialties, insurance plans, etc.) of more than two million doctors world-wide, and over three million reviews along with numeric ratings of four aspects. The original ratemds dataset has many missing values for multi-aspect ratings which reflects the fact that patients are less motivated to provide ratings from different aspects even if their comments are about multiple things. This problem shows the importance of the multi-aspect rating prediction/sentiment classification task (see Section 4.3). Due to the missing value problem, we first removed reviews with missing aspect-level ratings and eliminated records of doctors without reviews before investigating statistics of the dataset. Then, we obtain a refined ratemds dataset, in which distributions of doctors and reviews are shown in Fig. 4.2. In this dataset, there are more than $500K$ doctors and 2.7 million reviews, and the average number of reviews for each doctor is 4.6. From Fig. 4.2 (a), we observed that the distribution of doctors over review counts follows the power law distribution and almost 40% of doctors have only one review. Thus, it is difficult to apply collaborative filtering based methods to predict multi-aspect ratings.



(a) # of reviews for doctors.

(b) # of sentences in reviews.

(c) # of tokens in reviews.

(d) # of tokens in sentences.

Figure 4.2: Statistics of reviews in ratemds dataset.

Alternately, we can make use of textual reviews for the rating prediction task, which is the same as the sentiment classification task in this study. Therefore, we further studied the quality of textual reviews based on lengths of texts in order to make sure that they are not composed of short texts, since short texts may cause several problems in this task. First, short reviews cannot contain information of four aspects, which can probably confuse the classifier with respect to the aspect-keywords. Second, due to lack of semantic relationships [157, 122], it is difficult to use traditional knowledge discovery methods such as topic models [6] to automatically uncover the hidden thematic information from them. As a result, we cannot incorporate external knowledge discovered by these models into the sentiment classifiers for better classification performance. Figures 4.2(b) and 4.2(c) show the distribution of reviews over numbers of tokens and sentences, respectively. Fig. 4.2(d) is the distribution of sentences over the number of tokens. From these figures, we observed that most reviews have at least 2 sentences and over 12 tokens, and most sentences have more than 10 tokens, which indicates that reviews in this dataset are not dominated by short texts. Moreover, the average length of reviews are more than 4 sentences and 72 tokens, which implies that there are a number of reviews whose content covers all four aspects in this dataset.

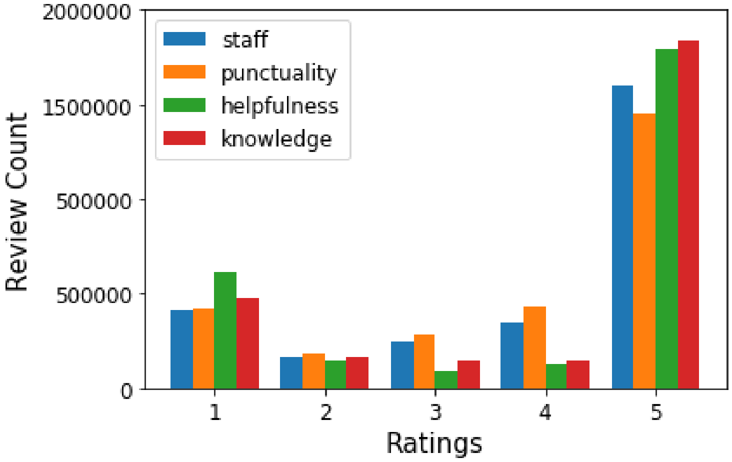


Figure 4.3: Statistics of reviews over aspect-level ratings.

Ratings

Each review comes with an overall rating and ratings for four different aspects, i.e., staff, punctuality, helpfulness and knowledge. The overall rating is the average of aspect-level ratings, which are integer numbers ranging from 1 to 5, where 1 and 5 represent extremely unsatisfied and satisfied, respectively. We show the distribution of reviews over rating scores in Fig. 4.3. From this figure, we observed that more than 60% of reviews have all aspect rating scores 5, which indicates that most patients are satisfied with their visits. About 17% of them are 1. It seems that patients with negative experience with their doctors tend to give extremely unsatisfied scores to express their sentiment, especially when their doctors are not helpful. Many patients are slightly unsatisfied with staff and punctuality even if they are satisfied with their doctors, which may be because of their appointments and waiting time.

Doctors

Apart from the basic statistics of reviews and ratings, it is also important to investigate demographic features of doctors, since they may affect the visit experience of patients. For example, doctors who work in urban hospitals may receive lower punctuality scores in general than those who work in suburban clinics. Each doctor has a certain specialty (e.g., dentist). In Fig. 4.4, we show the average ratings for doctors with different specialties. It can be seen that dentists have much higher rating scores than other types of doctors. General practitioners and family practitioners (family-gp) have lower punctuality scores than others, due to the fact that patients with nearly any issue can visit them and get referrals when they have complicated health issues. Therefore, incorporating these demographic features into sentiment prediction models may increase the accuracy of results. In the ratemds dataset, the key features of doctors includes gender, facility categories, specialties, locations and insurance plans.

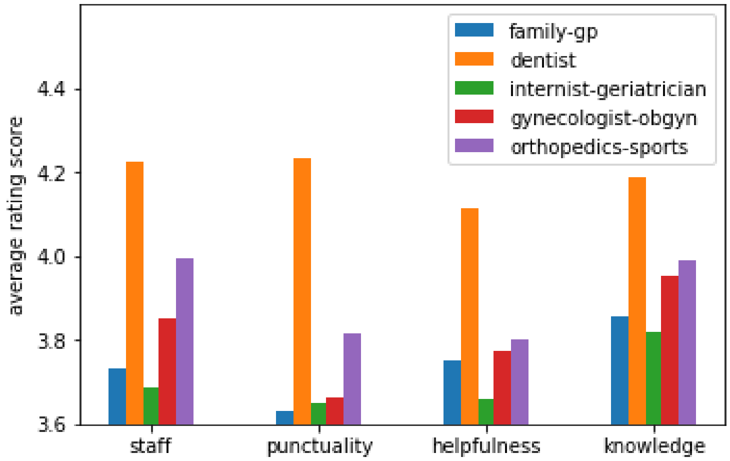


Figure 4.4: Ratings for doctors with different specialties.

We first observed that doctors in this dataset are from six different countries, i.e., United States (US), Canada (CA), Australia, India, United Kingdom and South Africa, where around 77% and 19% of them are located in the US and CA, respectively. There are three categories of facilities, i.e., hospital, clinic and urgent-care. About 90% of doctors work in clinics, 10% of them are in hospitals, and very few are in urgent-care. Many doctors work in different facilities and some of them work in two different countries. In this work, we remove those doctors who work in more than one country, because different countries have different healthcare systems. For the feature gender, we observed that around 32% of doctors are female. For the feature specialty, which has been briefly mentioned in the beginning of this section, each doctor is assigned one specialty and there are 57 different specialties. Almost 20% of doctors are family-gp. Dentists and obstetrician-gynecologists get relatively more reviews than doctors with other types of specialties.

4.2.2 Discover Aspect-Keywords

We further investigated reviews by extracting aspect-keywords using topic models [6]. Topic modeling approaches were considered because they can automatically uncover thematic information from a corpus in an unsupervised manner. In addition, keywords in each topic usually have strong semantic correlations and well-defined cluster structures. In the ratemds dataset, reviews are assumed to be written from different aspects (different topics), whose keywords are expected to be less correlated.

Table 4.1: Aspect-keywords extracted with the topic model.

Specialty	Aspect	Keyword Examples
family-gp	staff	staff, office, rude, nurse, service, charge, call, visit, contact, insurance, follow, phone.
	punctuality	wait, hour, long, time, late, appointment, minute.
	helpfulness	care, see, listen, regard, consider, refer, show, understanding.
	knowledge	lab, symptom, treatment, professional, medicine, knowledge, drug, skill, prescription, diagnosis.
dentist	staff	insurance, charge, service, receive, nice, kind, smile, front-desk, polite, sweet, respect, assistant, staff.
	punctuality	rush, drive, late, time, appointment, wait, day, long.
	helpfulness	help, make, feel, comfortable, ease, care, ask, follow.
	knowledge	knowledgeable, procedure, explain, treatment, implant, review, replace, perform, extraction, experience, professional, tooth.
gynecologist-obgyn	staff	call, tell, ask, nurse, rude, staff, office, nice, friendly, service.
	punctuality	time, wait, appointment, hour, long, minute, day, week, rush.
	helpfulness	care, concern, understanding, warm, ease, helpful, think, save, offer, answer, consider, refuse, suggest.
	knowledge	knowledgeable, test, exam, review, explain, complication, pregnancy, deliver, experience, baby, surgery, pain, hysterectomy, surgeon, medication, bleed, cry, fibroid, treatment, diagnosis, scar.

Datasets

We first separated the ratemds dataset based on countries and chose reviews for doctors in the US. Then, we divided selected reviews into sub-categories according to specialties. We tokenized all reviews with the SpaCy[5] package and removed stop-words, punctuation and rare words. Among all 57 specialties, we chose only three of them, i.e., family-gp, dentist, and gynecologist-obgyn, to illustrate our experiments and results.

Experiments and Results

Using the gensim[6] package, we apply the Latent Dirichlet Allocation (LDA) [6, 48] model to each dataset. The number of topics was set to 10 considering the fact that topics which are different from the four aspects may also be discovered. For each topic, we extracted top-20 keywords based on their weights. Finally, we empirically assigned these keywords to different aspects which have been shown in Table 4.1.

It can be seen from the table that *staff* usually represents *front-desk* or *nurse*. Their duties include receptions, contacting patients, managing insurance plans and bills, and so on. *Punctuality* is associated with *appointment* and *waiting time* in offices. From Fig. 4.3, we

[5] https://spacy.io/

[6] https://radimrehurek.com/gensim/

have found that fewer patients are satisfied with punctuality. This may be explained as it is hard to make an appointment, waiting time is too long, or doctors rush to see other patients. *Helpfulness* can be understood as bedside manner of doctors. For example, a good doctor can carefully listen to complaints of patients, answer their questions and make them feel comfortable. Finally, *knowledge* in general is related with *diagnosis*, *exam*, *treatment*, and so on. From the table, we also observed that keywords of *staff*, *punctuality* or *helpfulness* are similar to each other for doctors with different specialties. However, since they are experts in different fields, the *knowledge* for different specialties has different keywords. For example, *surgery*, *hysterectomy*, *fibroid*, and *pregnancy* are related with doctors specialized in gynecologist-obgyn.

4.3 Proposed Methods

In Section 4.2, we had a comprehensive understanding of statistics and key features of ratemds dataset, and also extracted aspect-keywords with the topic model. In this section, we perform document-level multi-aspect sentiment classifications for reviews in ratemds dataset.

4.3.1 Preliminaries

In the document-level multi-aspect sentiment classification problem, multi-aspect rating predictions can be viewed as tasks. Due to the strong correlations between different tasks, this problem can be naturally formulated as a multi-task learning problem. Hence, we propose a multi-task deep learning framework which takes plain-text reviews, aspect-keywords from topic models and features of doctors into consideration. Formally, this document-level multi-aspect sentiment classification problem can be described as follows: Given a textual review $X = (x_1, x_2, ..., x_T)$, keywords associated with different aspects $G = (G^1, G^2, ..., G^K)$ and a set of features ξ, our goal is to predict class labels, i.e., integer ratings, $y = (y^1, y^2, ..., y^K)$, where T and K are the number of tokens in the review and the number of aspects, respectively. x_t represents the one-hot encoding of word t. $G^K = (g_1^k, g_2^k, ..., g_M^k)$ is a list of keywords of aspect k, where g_m^k is the one-hot encoding of keyword m. y^k is an one-hot vector of the class label of aspect k. Specific to the ratemds dataset, there are four aspects, so $K = 4$, and each aspect has 5 classes corresponding to rating scores from 1 to 5. The proposed framework (see Fig. 4.5) has a *review encoder* to encode textual reviews, a *multi-aspect self-attention* layer to selectively focus on parts of the review for a given aspect, an *aspect-keywords guided-attention* layer to focus on parts of the review that are related to aspect-keywords, and an *aspect-specific feature encoder* to incorporate features of doctors into the sentiment classification.

We first use a word embedding [96] to map one-hot representations of tokens to a continuous vector space, thus, a review is represented as $(E_{x_1}, E_{x_2}, ..., E_{x_T})$, where E_{x_t} is the word vector of x_t. Then, a bi-directional GRU [18] encoder takes these word vectors as input and turns the review into a sequence of hidden states $H = (h_1, h_2, ..., h_M)$.

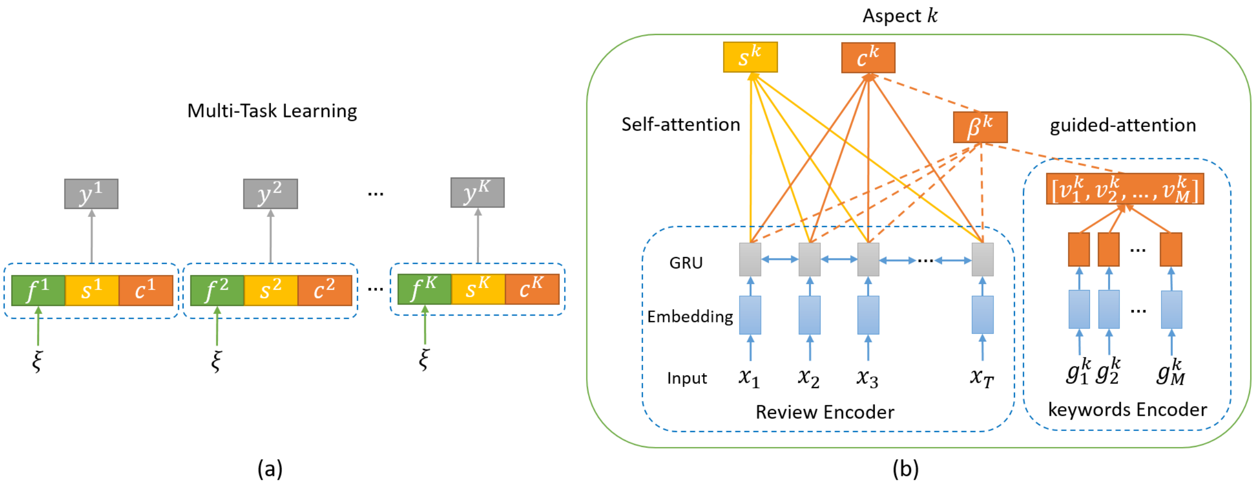


Figure 4.5: An illustration of the model architecture. (a) The proposed multi-task learning model. (b) Self-attention and guided attention for aspect k. Different aspects share the review encoder and word embedding.

4.3.2 Multi-Aspect Self-Attention

After encoding a review into a sequence of hidden states, our goal is to use these encoded vectors to predict rating scores (i.e., class labels) of different aspects. However, not all of them contribute equally to the predictions, especially for different aspects. Take the review in Fig. 4.1 as an example. A model might need to focus on "*wait at least 30 minutes*" to predict punctuality score, while for staff, we may put more attention to "*very curt and also very busy*". Therefore, we introduce a multi-aspect self-attention mechanism to capture important parts of each review.

Formally, for an aspect k, we first use a self-attention mechanism [161] to determine attention weights α_t^k of each token in the review

$$u_t^k = (r_{\text{self}}^k)^\top \tanh(W_{\text{self}}^k h_t + b_{\text{self}}^k), \quad \alpha_t^k = \frac{\exp(u_t^k)}{\sum_\tau \exp(u_\tau^k)} \tag{4.1}$$

where W_{self}^k, r_{self}^k and b_{self}^k are learnable parameters. Then, the representation of the review under self-attention can be calculated by taking the weighted sum of all hidden states,

$$s^k = \sum_{t=1}^{T} \alpha_t^k h_t \tag{4.2}$$

which will be used for the classification task.

4.3.3 Aspect-Keywords Guided-Attention

The multi-aspect self-attention mechanism relies on the model itself to discover relationships between class labels and keywords in a review. However, due to strong correlations of rating scores of different aspects, the model will be confused on ‘where to attend’, when aspect-level rating scores are the same. In this case, the model may make mistakes, like placing the same class label for all aspects for new reviews. This problem can be alleviated by bringing in external knowledge of keywords associated with different aspects (see Section 4.2.2).

Given a list of aspect-keywords for aspect k, we first obtain the word embedding for them, i.e., $(E_{g_1^k}, E_{g_2^k}, ..., E_{g_M^k})$. Then, each word vector is transformed into a hidden state with[7]

$$v_m^k = (1 - \sigma(W_0^k E_{g_m^k} + b_0^k)) \tanh(W_1^k E_{g_m^k} + b_1^k + b_3^k \sigma(W_2^k E_{g_m^k} + b_2^k)) \tag{4.3}$$

It is followed by concatenating all hidden states into a single vector $v^k = \left[v_1^k, v_2^k, ..., v_M^k\right]$. Here, the average of all vectors is not taken because we consider that averaging may neutralize some features. We then use the global attention mechanism [88, 4] to calculate alignment scores between encoded vectors of aspect-keywords and tokens in the review as

$$w_t^k = w(v^k, h_t) = (v^k)^\top W_{\text{guide}}^k h_t \tag{4.4}$$

where W_{guide}^k are learnable parameters. Thus, the guided-attention weights and vector representation of the review are obtained by

$$\beta_t^k = \frac{\exp(w_t^k)}{\sum_\tau \exp(w_\tau^k)}, \quad c^k = \sum_{t=1}^{T} \beta_t^k h_t \tag{4.5}$$

4.3.4 Aspect-Specific Feature Encoder

From the basic statistics given in Fig. 4.4, we can observe that some features of doctors (such as specialty and locations) also affect rating scores; therefore, we incorporate them into our model to improve the prediction accuracy. Formally, we embed one-hot representations of features of doctors into a continuous vector space for each aspect k as

$$f^k = W_f^k \xi + b_f^k \tag{4.6}$$

where W_f^k and b_f^k are model parameters.

4.3.5 Multi-Aspect Rating Prediction

So far, we have obtained aspect-specific representations of a review via self-attention and guided-attention mechanisms, and representations of features of doctors. These vectors will

[7] Here, we want to apply a single step GRU transformation for every keyword. But each aspect has only one GRU cell.

be concatenated and fed into a classifier, which is a single layer feed-forward network with a softmax activation function, to predict rating scores. The classifier yields a probability distribution of class labels of different aspects with

$$y^k = \text{softmax}(W^k_{\text{out}}[f^k, s^k, c^k] + b^k_{\text{out}}) \tag{4.7}$$

where W^k_{out} and b^k_{out} are parameters.

Given predicted labels y^k and ground-truth labels $\hat{y}^k$, we train our model in an end-to-end manner using back-propagation, where the loss function is defined as the cross-entropy loss. The goal of the training is to minimize average cross-entropy error between y^k and $\hat{y}^k$ for all aspects. Formally, it is given as

$$\mathcal{L}_\theta = -\sum_{k=1}^{K}\sum_{i=1}^{N} \hat{y}^k_i \log(y^k_i) + \lambda\Omega(\theta) \tag{4.8}$$

where $\Omega(\theta)$ and λ are a regularizer and a scalar, respectively. θ is a parameter set including all weight matrices and bias vectors. N represents the number of classes.

4.4 Experiments

In this section, we describe an extensive set of experiments on the ratemds dataset for document-level multi-aspect sentiment classification and explain different experimental results. We start with introducing two subsets of the ratemds dataset, baseline methods, and implementation details of the proposed model and evaluation metrics. Then, we will show the classification performance of different models along with some qualitative results.

4.4.1 Datasets Used

We created two subsets from the ratemds dataset, i.e., ratemds-us and ratemds-ca, based on countries that doctors work in. We chose the US and CA, because 90% of reviews are from these two countries (see Fig. 4.4 (a)). The ratemds-us consists of 1,414,235 reviews for 385,407 doctors, while ratemds-ca has 1,252,941 reviews for 99,719 doctors. We first tokenized texts with SpaCy[8]. Since features of doctors are used as additional input, we also extracted attributes of doctors, including specialties, insurance plans, locations, genders and facilities, and transformed them into one-hot representations. In addition, aspect-keywords were selected from latent topics. Finally, we randomly split each dataset into training, development and testing sets with the ratio of 80:10:10.

[8] `https://spacy.io/`

4.4.2 Compared Methods and Implementation Details

We compare the proposed model with different baseline methods, including conventional classification and deep learning models.

- **MAJOR**. This method simply uses the majority label of each aspect in the training set as the prediction label.
- **GLVL**. In this model, we first calculate the vector representation of each review by taking the average of vectors of all keywords in the review. Word vectors were pre-trained on the Twitter datasets with 2 billion tweets by GloVe [106]. Then, we use the LIBLINEAR [27] package[9] for the classification task.
- **BOWL**. This model feeds Bag-Of-Words (BOW) representations of reviews into the LIBLINEAR package for the sentiment classification. In the experiment, we have removed stop-words and punctuation in textual reviews to make the model capture keywords efficiently.
- **CNN**. We adopt the convolutional neural network (CNN) structure proposed in [61, 164] for the rating prediction of reviews. In our experiments, 1-directional convolutions with different filter sizes along the sequence time-step dimension are first applied to the word embedding of a review. Then, a max-over-time pooling operation [23] is built upon each feature map. By selecting the maximum value, we obtain the key-feature of each filter. Finally, the vector representation of the review is obtained by concatenating all features. This vector will be fed into a feed-forward network for classification (similar to other deep learning models.).
- **GRU**. We use GRU to refer to a bi-directional GRU with multiple hidden layers [18]. In this model, we concatenate output vectors of the last hidden states of the top hidden layer in both forward and backward directions[10] to represent a review.
- **GRU-ATN**. GRU-ATN first builds a self-attention layer [138, 75] on top of a recurrent neural network. With attention weights, we can compute a context vector for a review by taking the weighted sum of all hidden states (see Eq. (4.1)).
- **MT-BASE and MT-FEAT**. MT-BASE is a multi-task learning framework with only a review encoder, self-attention layers and classifiers (see Fig. 4.5) [163]. In this model, different tasks (i.e., aspects) share the same review encoder. MT-FEAT also takes features of doctors into consideration.

We implemented all deep learning models using PyTorch [105] and model parameters are selected based on the development set. For both ratemds-us and ratemds-ca, vocabulary sizes are set to 50,000. We do not use the pre-trained word embeddings [96, 106] and they are learned from scratch during the training. The dimension of word embeddings is set to 128. For CNN, filter sizes were chosen to be 3, 4, 5 and the number of filters are 100 for each size. For all GRU based models, the dimension of hidden states is set to 128 and the number

[9] `https://www.csie.ntu.edu.tw/~cjlin/liblinear/`

[10] Here, the first token in a sequence corresponds to the last hidden state in the backward direction.

Table 4.2: Performance comparison of different models on ratemds-us. For MSE, smaller is better.

	Staff		Punctuality		Helpfulness		Knowledge	
	F-score	MSE	F-score	MSE	F-score	MSE	F-score	MSE
MAJOR	0.1453	3.6394	0.1370	3.7749	0.1546	4.5445	0.1575	3.8039
GLVL	0.2893	1.9486	0.2777	2.0598	0.3341	1.4356	0.3140	1.6360
BOWL	0.3805	1.3691	0.3744	1.4440	0.4142	0.8564	0.4151	1.0056
CNN	0.3767	1.1588	0.3721	1.2375	0.4208	0.5355	0.4205	0.7079
GRU	0.4101	0.9717	0.3885	1.1000	0.4602	0.4617	0.4419	0.6326
GRU-ATN	0.4090	0.9638	0.3896	1.0938	0.4479	0.4817	0.4597	0.6078
MT-BASE	0.4093	0.9495	0.3997	1.0273	0.4554	0.4569	0.4528	0.5993
MT-FEAT	0.4187	0.9456	0.3976	1.0443	0.4684	0.4461	0.4721	0.5722
MT-FAKGA (our)	**0.4193**	**0.9061**	**0.4103**	**1.0018**	**0.4787**	**0.4437**	**0.4822**	**0.5681**

of layers is 2. All parameters are trained with the ADAM [63] optimizer with learning rate 0.0001. Gradient clipping has also been applied to prevent gradient explosion.

In this study, we adopt ‘macro’ averaged F-score and mean squared error (MSE) to evaluate performance of different models. Accuracy has been used in [163, 69], however their models are only tested on reviews with different aspect-level ratings, since those with identical aspect-level ratings can make it difficult for their models to distinguish keywords of different aspects. In our experiments, these reviews are still kept, because we assume that aspect-keywords guided-attention mechanism can alleviate this problem. However, based on distributions of aspect-level ratings and their correlations (see Fig. 4.3), the data is highly imbalanced, therefore, accuracy is not a suitable evaluation metric and we adopt F-score instead. Both accuracy and F-score are based on exact match of class labels, however, for sentiment analysis, we only need predicted rating scores close to the ground-truth. For example, if the ground truth score is 5, a model still performs reasonably well by predicting 4. Therefore, MSE is also a promising metric.

4.4.3 Rating Prediction Performance

We first present quantitative results of different models in Tables 4.2 and 4.3, where we use bold font to show the best performance values and underlining to highlight the second best values.

From these two tables, we can observe that MAJOR gets the lowest performance among all compared methods, since it simply classifies all reviews to the dominant labels without using textual reviews. GLVL achieves much better results than MAJOR, but is still not as good as other methods. Although it attempts to take advantage of semantic information of the word embedding, simply averaging all word vectors in a review can cause information offset,

Table 4.3: Performance comparison of different models on ratemds-ca.

	Staff		Punctuality		Helpfulness		Knowledge	
	F-score	MSE	F-score	MSE	F-score	MSE	F-score	MSE
MAJOR	0.1466	3.1578	0.1377	3.3958	0.1590	3.8706	0.1613	3.2678
GLVL	0.2665	2.1426	0.2645	2.1774	0.3209	1.6168	0.3028	1.6960
BOWL	0.3663	1.4573	0.3651	1.5007	0.4239	0.8667	0.4179	0.9554
CNN	0.3480	1.3431	0.3568	1.3520	0.4267	0.5871	0.4197	0.7042
GRU	0.3778	1.1466	0.3958	1.1282	0.4714	0.4742	0.4519	0.5977
GRU-ATN	0.3907	1.0910	0.3891	1.1457	0.4827	0.4743	0.4739	0.5714
MT-BASE	0.3894	1.0730	0.3905	1.1205	0.4806	0.4686	0.4759	0.5568
MT-FEAT	0.3965	1.0838	0.3916	1.1020	0.4856	0.4556	0.4833	0.5362
MT-FAKGA (our)	**0.4013**	**1.0403**	**0.3965**	**1.0781**	**0.5051**	**0.4432**	**0.5025**	**0.5203**

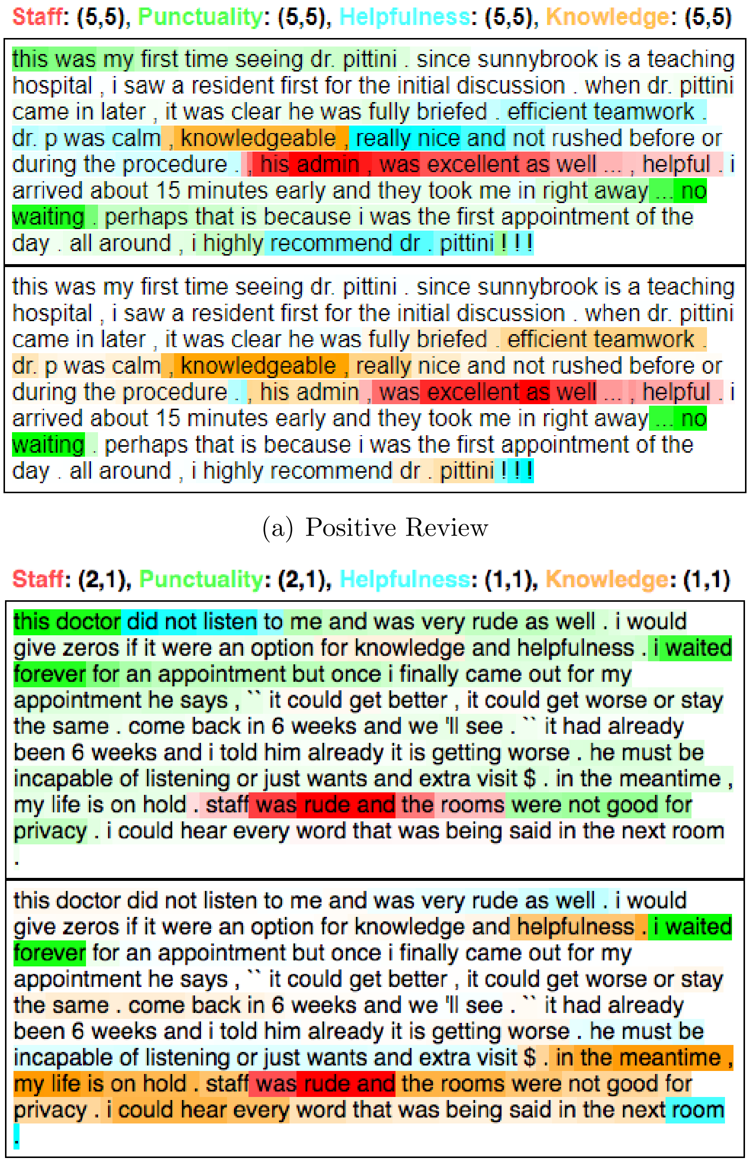
which results in poor review representation[11]. BOWL can also capture word-level semantic information via bag-of-words (BOW) representations of reviews. It performs significantly better than GLVL and as well as CNN. Compared to GLVL, the BOW representation encodes each review into a high-dimensional space, thus, BOWL requires more parameters to classify reviews which avoids under-fitting. On the other hand, by removing stop-words and punctuation in reviews, we only keep keywords relevant to classification and frequency of keywords in a review can partially reflect their importance. Therefore, representations of reviews by BOWL are better than those obtained by GLVL.

Compared to traditional methods and CNN, GRU based models have achieved significantly better results on both datasets. GRU and GRU-ATN are simple classification methods and trained separately for different aspects, while MT-BASE, MT-FEAT, MT-FAKGA are multi-task models and they share the word embedding and recurrent hidden layers. Since most model parameters are attributed to these layers, multi-task models require significantly fewer parameters. Moreover, GRU and GRU-ATN need K different training for K different aspects, while the multi-task learning framework can simultaneously learn different aspects, thus, they require much lesser training time. As to the performance of rating predictions, we first observe that multi-task learning models can perform as well as or even better than GRU and attention-based GRU models. MT-FEAT performs slightly better than MT-BASE in most cases, since it considers features of doctors. By incorporating knowledge from aspect-keywords, we further improve the performance of MT-FEAT. The proposed MT-FAKGA achieves the best results in terms of F-score and MSE on both datasets.

4.4.4 Attention Visualization

As the attention mechanism enables a model to selectively focus on important parts of reviews, visualization of attention weights has become a popular tool that helps to interpret models

[11] Before averaging word vectors, we have removed stop-words and punctuation from reviews. However, the performance has not improved significantly using this trick.



Staff: (5,5), Punctuality: (5,5), Helpfulness: (5,5), Knowledge: (5,5)

this was my first time seeing dr. pittini . since sunnybrook is a teaching hospital , i saw a resident first for the initial discussion . when dr. pittini came in later , it was clear he was fully briefed . efficient teamwork . dr. p was calm , knowledgeable , really nice and not rushed before or during the procedure . , his admin , was excellent as well ... , helpful . i arrived about 15 minutes early and they took me in right away ... no waiting . perhaps that is because i was the first appointment of the day . all around , i highly recommend dr . pittini ! ! !

this was my first time seeing dr. pittini . since sunnybrook is a teaching hospital , i saw a resident first for the initial discussion . when dr. pittini came in later , it was clear he was fully briefed . efficient teamwork . dr. p was calm , knowledgeable , really nice and not rushed before or during the procedure . , his admin , was excellent as well ... , helpful . i arrived about 15 minutes early and they took me in right away ... no waiting . perhaps that is because i was the first appointment of the day . all around , i highly recommend dr . pittini ! ! !

(a) Positive Review

Staff: (2,1), Punctuality: (2,1), Helpfulness: (1,1), Knowledge: (1,1)

this doctor did not listen to me and was very rude as well . i would give zeros if it were an option for knowledge and helpfulness . i waited forever for an appointment but once i finally came out for my appointment he says , `` it could get better , it could get worse or stay the same . come back in 6 weeks and we 'll see . `` it had already been 6 weeks and i told him already it is getting worse . he must be incapable of listening or just wants and extra visit $. in the meantime , my life is on hold . staff was rude and the rooms were not good for privacy . i could hear every word that was being said in the next room .

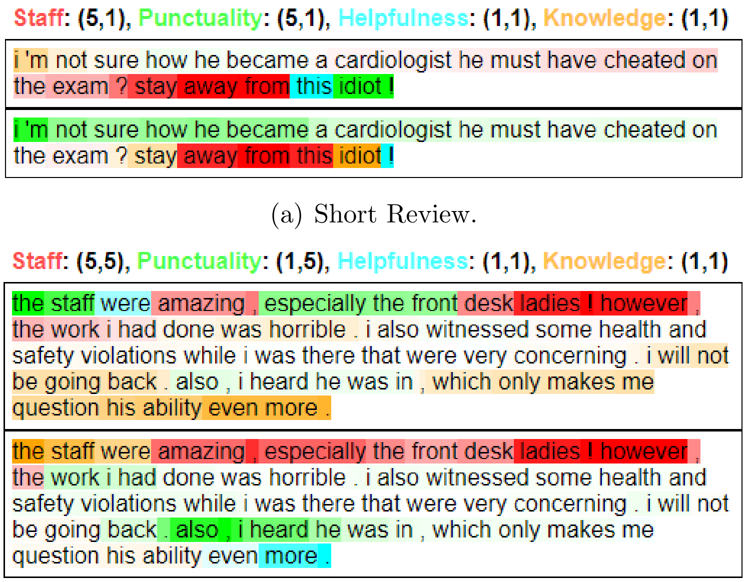
this doctor did not listen to me and was very rude as well . i would give zeros if it were an option for knowledge and helpfulness . i waited forever for an appointment but once i finally came out for my appointment he says , `` it could get better , it could get worse or stay the same . come back in 6 weeks and we 'll see . `` it had already been 6 weeks and i told him already it is getting worse . he must be incapable of listening or just wants and extra visit $. in the meantime , my life is on hold . staff was rude and the rooms were not good for privacy . i could hear every word that was being said in the next room .

(b) Negative Review

Figure 4.6: Visualization of attention weights. In parentheses, first and second numbers represent ground-truth and predicted ratings, respectively. For each sub-figure, the first and second rows represent self-attention and guided-attention weights, respectively. Different aspects are labeled with different colors, therefore, this figure is best viewed in color.

and analyze experimental results [163, 155]. Specific to our multi-aspect classification task, our goal is to investigate if models accurately attend keywords of different aspects or not.

In Fig. 4.6, we first show one example with positive ratings and one with negative ratings. In these examples, the proposed model makes correct predictions of sentiment, and reviews



Staff: (5,1), Punctuality: (5,1), Helpfulness: (1,1), Knowledge: (1,1)

i 'm not sure how he became a cardiologist he must have cheated on the exam ? stay away from this idiot !

i 'm not sure how he became a cardiologist he must have cheated on the exam ? stay away from this idiot !

(a) Short Review.

Staff: (5,5), Punctuality: (1,5), Helpfulness: (1,1), Knowledge: (1,1)

the staff were amazing , especially the front desk ladies ! however , the work i had done was horrible . i also witnessed some health and safety violations while i was there that were very concerning . i will not be going back . also , i heard he was in , which only makes me question his ability even more .

the staff were amazing , especially the front desk ladies ! however , the work i had done was horrible . i also witnessed some health and safety violations while i was there that were very concerning . i will not be going back . also , i heard he was in , which only makes me question his ability even more .

(b) Review does not cover punctuality.

Figure 4.7: Visualization of attention weights. This figure will be best viewed in color.

contain keywords of all four different aspects, therefore, we only need to check if the model can successfully detect these keywords. Take Fig. 4.6(a) as an example, both self-attention and guided-attention focus on "*excellent, helpful*" for staff. As to punctuality, both of them capture "*no waiting*". However, self-attention also highlights *"this was my first time ..."* which is not quite relevant. Helpfulness and knowledge are often difficult to be distinguished in many examples. Here, self-attention focuses on *"efficient teamwork, calm, really nice and not rush"* for helpfulness, while guided-attention does not successfully detect these keywords, which might be because the extracted aspect-keywords do not align well with *"calm, nice, rush"*. Finally, for knowledge, both mechanisms capture *"knowledgeable"*. The guided-attention also treats *"efficient teamwork"* as knowledge aspect keywords, which is reasonable. For the negative review (see Fig. 4.6 (b)), both self-attention and guided-attention highlight *"rude"* for staff, and *"i waited forever"* for punctuality. Therefore, the model predicts a rating score of 1 for both aspects, which is consistent with ground-truth in sentiment sense. As to helpfulness, guided-attention incorrectly attends *"room"*. However, it also focuses on *"he must be incapable of listening or just wants an extra visit"* which reflects the fact that the doctor does not help. On the other hand, self-attention focuses on *"did not listen"*, which is also good. Finally, we observe that self-attention fails to capture knowledge aspect keywords, while guided-attention highlights *"helpfulness, my life is on hold, rooms were not good for privacy"*, which can partially indicate that the patient is not happy with the knowledge of this doctor.

As we can see from the above examples, an attention mechanism cannot always build accurate connections between rating and keywords of the same aspect. In practice, we found that failure of attention may be caused by several reasons: 1) A review is very short and only discusses a certain issue. For example, in Fig. 4.7 (a), the patient first questioned the knowledge of the doctor and then suggested others to stay away from him/her. However, it does not mention anything about staff and punctuality. Therefore, both self-attention and guided-attention make mistakes in finding aspect-keywords, which will then result in incorrect predictions. 2) A review is long enough, but does not cover all aspects. Fig. 4.7 (b) shows an example in which the patient did not mention anything about punctuality. Thus, *"the staff, also, i heard"* are highlighted for this aspect, which lead to the opposite sentiment. 3) We may need some reasoning for a review to make predictions. For example, some reviews start with *"dr. started out being an excellent doctor for us."*, then the patients begin to complain about different issues. 4) Many keywords and phrases are ambiguous in different contexts, such as *"long"* in *"wait very long"* and *"he has been my doctor very long"*.

4.4.5 Practical Implications

In this section, we describe the practical applications of our tool. Similar to the example shown in Fig. 4.1, our tool can highlight keywords corresponding to different aspects, so that both patients and doctors can get the important information from these reviews more efficiently. For doctors, they can find out their problems by just visualizing keywords of the aspects with negative ratings. For example, if the punctuality is a problem in a clinic, then, "wait very long" may appear in many reviews. Coloring these keywords can help doctors to find out this problem in seconds. On the other hand, patients may need to read the reviews of many doctors, which takes a long time, before they can find their primary care physicians or specialists. However, if they are trying to find a doctor who is caring and helpful, they can use this tool, which can also highlight the keywords of positive and negative sentiment with different colors for aspect "helpfulness", to see the experience of other patients instead of browsing all reviews.

4.5 Summary

Online doctor review systems provide a platform for patients to give feedback to their doctors. These reviews not only help other patients to learn more about a doctor before they visit, but also help doctors to improve their service quality. From these reviews, we can also discover common concerns of patients and existing problems in clinics. In this study, we systematically investigated the dataset from one such review system, i.e., ratemds.com, where each review comes with an overall rating and ratings for four different aspects. We first studied statistics of reviews, ratings and doctors. Then, we attempted to explore the content of reviews by extracting aspect-keywords with topic modeling. We proposed a multi-task learning framework for the document-level multi-aspect sentiment classification, which can help us to not only recover missing aspect-level ratings and detect inconsistent rating scores, but also identify aspect-keywords in a given review based on ratings. The proposed model takes both features of doctors and aspect-keywords into consideration. Extensive experiments have been conducted on two subsets of the ratemds dataset to demonstrate the effectiveness of the proposed model. Qualitative results show the power of attention mechanisms. In the future, we will work on solving these problems and applying fine-grained aspect-based sentiment classification techniques to study these reviews.

Chapter 5

Corpus-level and Concept-based Explanation Methods for Model Interpretation and Review Understanding

This chapter introduces a corpus-level explanation approach, which aims to capture causal relationships between keywords and model predictions via learning importance of keywords for predicted labels across a training corpus based on attention weights, to interpret attention-based deep document classification models. A concept-based explanation method, which can automatically learn higher level concepts and their importance to the model prediction task, has also been proposed. The rest of this chapter is organized as follows: The introduction of this chapter is first presented in Section 5.1. In Section 5.2, we first present details of our proposed abstraction-aggregation network (AAN), and then discuss corpus-level and concept-based explanation methods. In Section 5.3, we evaluate different self-attention and AAN based models on three different datasets. We also show how corpus-level and concept-based explanations can help us interpret attention-based classification models and understand training corpora. Our discussion concludes in Section 5.4.

5.1 Background and Motivation

Attention Mechanisms [4] have boosted performance of deep learning models in a variety of natural language processing (NLP) tasks, such as sentiment analysis [151, 108], semantic parsing [145], machine translation [88], reading comprehension [45, 26] and others. Attention-based deep learning models have been widely investigated not only because they achieve state-of-the-art performance, but also because they can be interpreted by identifying important input information via visualizing heat-maps of attention weights [135, 144, 35], namely attention visualization. Therefore, attention mechanisms help end-users to understand models and diagnose trustworthiness of their decision making.

However, the attention visualization approach still suffers from several drawbacks: 1) The fragility of attention weights can easily make end-users find contradicting examples, especially for noisy data and cross-domain applications. For example, a model may attend on punctuation or stop-words. 2) Attention visualization cannot automatically extract high-level concepts that are important for model predictions. For example, when a model assigns news articles to *Sports*, relevant keywords may be *player*, *basketball*, *coach*, *nhl*, *golf*, and *nba*. Obviously, we can build three concepts/clusters for this example, i.e., roles (*player*, *coach*), games (*basketball*, *soccer*), and leagues (*nba*, *nhl*). 3) Attention visualization still relies on human experts to decide if keywords attended by models are important to model predictions.

There have been some studies that attempt to solve these problems. For example, Jain *et al.* [53] and Serrano *et al.* [121] focused on studying if attention can be used to interpret a model, however, there are still problems in their experimental designs [53]. Yeh *et al.* [162] tried to apply a generic concept-based explanation method to interpret BERT models in the text classification task, however, they did not obtain semantically meaningful concepts for model predictions. Antognini *et al.* [2] introduced a concept explanation method that first extracts a set of text snippets as concepts and infers which ones are described in the document, and then it explained the predictions of sentiment with a linear aggregation of concepts. In this study, we propose a general-purpose corpus-level explanation method and a concept-based explanation method based on a novel Abstraction-Aggregation Network (AAN) to tackle the aforementioned drawbacks of attention visualization. We summarize the primary contributions of this study as follows:

- To solve the first problem, we propose a *corpus-level explanation* method, which aims to discover causal relationships between keywords and model predictions. The importance of keywords is learned across a training corpus based on attention weights. Thus, it can provide more robust explanations compared to attention visualization case studies. The discovered keywords are semantically meaningful for model predictions.
- To solve the second problem, we propose a *concept-based explanation method* (case-level and corpus-level) that can automatically learn semantically meaningful concepts and their importance to model predictions. The concept-based explanation method is based on an AAN that can automatically cluster keywords, which are important to model predictions, during the end-to-end training for the main task. Compared to the basic attention mechanisms, the models with AAN do not compromise on classification performance or introduce any significant number of new parameters.
- To solve the third problem, we build a *Naïve Bayes Classifier* (NBC), which is based on an *attention-based bag-of-words document representation* technique and the causal relationships discovered by the corpus-level explanation method. By matching predictions from the model and NBC, i.e., consistency analysis, we can verify if the discovered keywords are

important to model predictions. This provides an automatic verification pipeline for the results from the corpus-level explanation and concept-based explanation methods.

5.2 Proposed Methods

In this section, we first introduce the classification framework and our Abstraction-Aggregation Network (AAN). Then, we systematically discuss the *corpus-level explanation*, *concept-based explanation*, and attention-based *Naïve Bayes Classifier*.

5.2.1 The Proposed Model

Basic Framework

A typical document classification model is equipped with three components, i.e., an encoder, an attention or pooling layer and a classifier. 1) **Encoder**: An encoder reads a document, denoted by $d = (w_1, w_2, ..., w_T)$, and transforms it to a sequence of hidden states $H = (h_1, h_2, ..., h_T)$. Here, w_t is the one-hot representation of token t in the document. h_t is also known as a word-in-context representation. Traditionally, the encoder consists of a word embedding layer followed by a LSTM [47] sequence encoder. Recently, pre-trained language models [26, 160, 107] have emerged as an important component for achieving superior performance on a variety of NLP tasks including text classification. Our model is adaptable to any of these encoders. 2) **Attention/Pooling**: The attention or pooling (average- or max-pooling) layer is used to construct a high-level document representation, denoted by v^{doc}. In attention networks, the attention weights show the contributions of words to the representations [161, 75]. Compared with pooling, attention operations can be well interpreted by visualizing attention weights [161]. 3) **Classifier**: The document representation is passed into a classifier to get the probability distribution over different class labels. The classifier can be a multi-layer feed-forward network with activation layer followed by a softmax layer, i.e., $y = \text{softmax}(W_2 \cdot \text{ReLU}(W_1 \cdot v^{\text{doc}} + b_1) + b_2)$, where W_1, W_2, b_1 and b_2 are model parameters.

To infer parameters, we can minimize the averaged cross-entropy error between predicted and ground-truth labels. Here, loss function is defined as $\mathcal{L}_\theta = -\sum_{l=1}^{L} \hat{y} \log(y)$, where $\hat{y}$ represents the ground-truth label and L is the number of class labels. The model is trained in an end-to-end manner using back-propagation.

Abstraction-Aggregation Network

In order to use different explanation methods, especially concept-based explanation, to interpret deep neural networks, we propose a novel AAN for the Attention/Pooling layer,

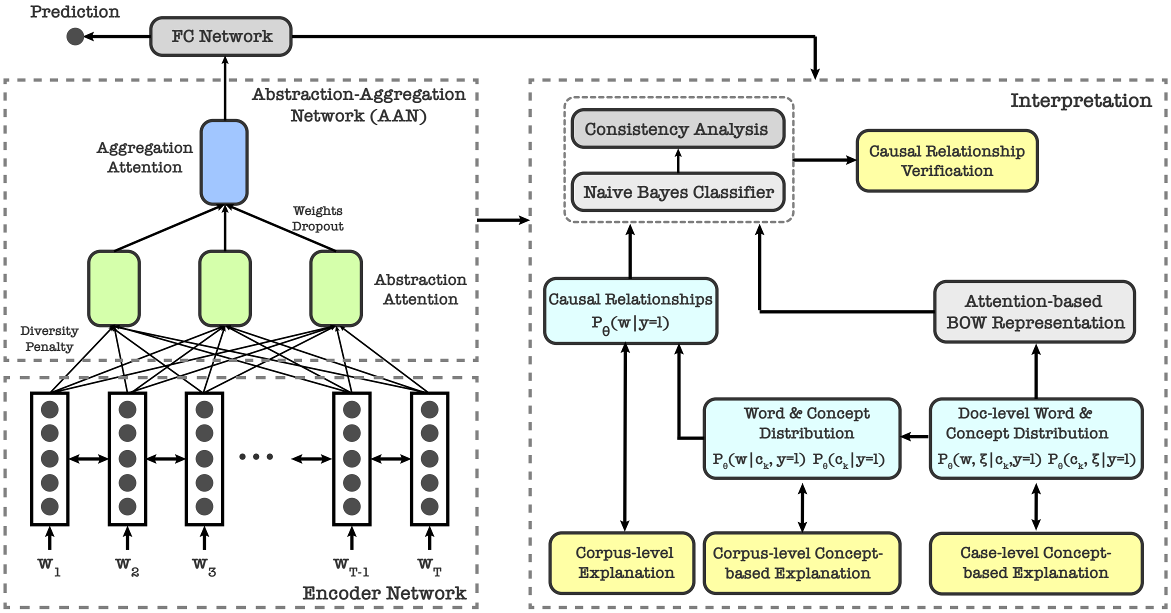


Figure 5.1: The proposed Abstraction-Aggregation Network and different interpretation methods.

which first captures keywords for different concepts from a document, and then aggregates all concepts to construct the document representation (see Fig. 5.1).

AAN has two stacked attention layers, namely, *abstraction-attention (abs)* and *aggregation-attention (agg)* layers. In the *abs* layer, for each attention unit k, we calculate the alignment score $u_{k,t}^{\text{abs}}$ and attention weight $\alpha_{k,t}^{\text{abs}}$ as follows:

$$\begin{aligned} u_{k,t}^{\text{abs}} &= (g_k^{\text{abs}})^\top h_t, \\ \alpha_{k,t}^{\text{abs}} &= \frac{\exp(u_{k,t}^{\text{abs}})}{\sum_{\tau=1}^{T}\exp(u_{k,\tau}^{\text{abs}})}, \end{aligned} \tag{5.1}$$

where g_k^{abs} are model parameters. Here, we do not apply linear transformation and tanh activation when calculating alignment scores for two reasons: 1) **Better intuition**: Calculating attention between g_k^{abs} and h_t in Eq. (5.1) is the same as calculating a normalized similarity between them. Therefore, abstraction-attention can also be viewed as a clustering process, where g_k^{abs} determines the centroid of each cluster. In our model, concepts are related to the clusters discovered by AAN. 2) **Fewer parameters**: Without the linear transformation layer, the abstraction-attention layer only introduces $K \times |h_t|$ new parameters, where $|h_t|$ is the dimension of h_t and $K \ll |h_t|$. The k^{th} representation is obtained by $v_k^{\text{abs}} = \sum_{t=1}^{T}\alpha_{k,t}^{\text{abs}}h_t$. We use K to denote the total number of attention units.

In the *agg* layer, there is only one attention unit. The alignment score u_k^{agg} and attention weight α_k^{agg} are obtained by

$$u_k^{\text{agg}} = (g^{\text{agg}})^\top \tanh(W_{\text{agg}} v_k^{\text{abs}} + b_{\text{agg}}),$$

and

$$\alpha_k^{\text{agg}} = \frac{\exp(u_t^{\text{agg}})}{\sum_{\kappa=k}^{K} \exp(u_\kappa^{\text{agg}})},$$

where $W_{\text{agg}}, b_{\text{agg}}$ and g^{agg} are model parameters. The final document representation is obtained by $v^{\text{doc}} = \sum_{k=1}^{K} \alpha_k^{\text{agg}} v_k^{\text{abs}}$. It should be noted that AAN is different from hierarchical attention [161], which aims to get a better representation. However, AAN is used to automatically capture concepts/clusters. We have also applied two important techniques to obtain semantically meaningful concepts.

1) Diversity penalty for abstraction-attention weights: To encourage the diversity of concepts, we introduce a new penalization term to abstraction-attention weights $A = [\overrightarrow{\alpha}_1^{\text{abs}}, \overrightarrow{\alpha}_2^{\text{abs}}, ..., \overrightarrow{\alpha}_K^{\text{abs}}] \in \mathbb{R}^{T \times K}$, where $\overrightarrow{\alpha}_k^{\text{abs}} = (\alpha_{k,1}^{\text{abs}}, \alpha_{k,2}^{\text{abs}}, ..., \alpha_{k,T}^{\text{abs}})^\top$. We define the penalty function as

$$\mathcal{L}_{\text{div}} = \frac{1}{K} \| A^\top A - I \|_F, \tag{5.2}$$

where $\| \cdot \|_F$ represents the Frobenius norm of a matrix. Hence, the overall loss function is expressed as $\mathcal{L} = \mathcal{L}_\theta + \mathcal{L}_{\text{div}}$.

2) Dropout of aggregation-attention weights: In the aggregation-attention layer, it is possible that $\alpha_k^{\text{agg}} \approx 1$ for some k, and other attention weights tend to be 0. To alleviate this problem, we apply dropout with a small dropout rate to aggregation-attention weights $(\alpha_1^{\text{agg}}, \alpha_2^{\text{agg}}, ..., \alpha_K^{\text{agg}})$, namely attention weights dropout. It should be noted that a large dropout rate has negative impact on the explanation, since it discourages the diversity of concepts. More specifically, the model will try to capture keywords in the dropped abstraction-attention units by the other units.

5.2.2 Explanation

In this section, we discuss corpus-level and concept-based explanations. Given a corpus $\mathcal{C}$ with $|\mathcal{C}|$ documents, we use d or ξ to represent a document. Let us also use θ to denote all parameters of a model and $\mathcal{V}$ to represent the vocabulary, where $|\mathcal{V}|$ is the size of $\mathcal{V}$. Throughout this chapter, we will assume that both prior document probability $p(d)$ and prior label probability $p_\theta(y = l)$ are constants. For example, in a label-balanced dataset, $p_\theta(y = l) \approx 1/L$.

We will first apply the attention weights visualization technique to the proposed AAN model.

Here, the document representation can be directly expressed by the hidden states, i.e.,

$$v^{\text{agg}} = \sum_{t=1}^{T} \left(\sum_{k=1}^{K} \alpha_k^{\text{agg}} \alpha_{k,t}^{\text{abs}} \right) h_t,$$

where

$$\alpha_t^d = \sum_{k=1}^{K} \alpha_k^{\text{agg}} \alpha_{k,t}^{\text{abs}} \tag{5.3}$$

gives the contribution of word w_t to the document representation. Therefore, we can interpret a single example via visualizing the combined weights α_t^d.

Corpus-Level Explanation

Corpus-level explanation aims to find causal relationships between keywords captured by the attention mechanism and model predictions, which can provide robust explanation for the model. To achieve this goal, we learn distributions of keywords for different predicted labels on a training corpus based on attention weights.

Formally, for a given word $w \in \mathcal{V}$ and a label l predicted by a model θ[1], the importance of the word to the label can be estimated by the probability $p_\theta(w|y = l)$ across the training corpus $\mathcal{C}_{\text{train}}$ since the model is trained on it. Therefore, $p_\theta(w|y = l)$ can be expanded as follows:

$$p_\theta(w|y = l) = \sum_{\xi \in \mathcal{C}_{\text{train}}^l} p_\theta(w, \xi|y = l), \tag{5.4}$$

where $\mathcal{C}_{\text{train}}^l \subset \mathcal{C}_{\text{train}}$ consists of documents with model predicted label l. For each document $\xi \in \mathcal{C}_{\text{train}}^l$, probability $p_\theta(w, \xi|y = l)$ represents the importance of word w to label l, which can be defined using attention weights, i.e.,

$$p_\theta(w, \xi|y = l) := \frac{\sum_{t=1}^{T} \alpha_t^\xi \cdot \delta(w_t, w)}{\sum_{\xi' \in \mathcal{C}_{\text{train}}} f_{\xi'}(w) + \gamma}, \tag{5.5}$$

where $f_{\xi'}(w_t)$ is frequency of w_t in document ξ' and γ is a smoothing factor. $\delta(w_t, w) = \begin{cases} 1 & \text{if } w_t = w \\ 0 & \text{otherwise} \end{cases}$ is a delta function. The denominator is applied to reduce noise from stop-words and punctuation. For the sake of simplicity, we will use $p_\theta(w, l, \mathcal{C})$ to denote $p_\theta(w_t|y = l)$, where $\mathcal{C}$ corresponds to the corpus in Eq. (5.4), and can be different from $\mathcal{C}_{\text{train}}$ in our applications. The denominator in Eq. (5.5) is always determined by the training corpus.

[1] Here, the label is the model's prediction, not the ground-truth label, because our goal is to explain the model.

As to applications: 1) Since Eq. (5.4) captures the importance of words to model predicted labels, we can use it as a criterion for finding their causal relationships. In experiments, we can collect top-ranked keywords for each label l for further analysis. 2) We can also use corpus-level explanation to measure the difference between two corpora (i.e., $\mathcal{C}_{\text{test1}}$ and $\mathcal{C}_{\text{test2}}$). Formally, we can compare $\frac{|\mathcal{C}_{\text{train}}|}{|\mathcal{C}_{\text{test1}}|} \cdot p_\theta(w, l, \mathcal{C}_{\text{test1}})$ with $\frac{|\mathcal{C}_{\text{train}}|}{|\mathcal{C}_{\text{test2}}|} \cdot p_\theta(w, l, \mathcal{C}_{\text{test2}})$ across different words and class labels. The difference can be evaluated by Kullback-Leibler divergence [67]. In addition, we can get mutual keywords shared across different domains based on these distributions.

It should be noted that the corpus-level explanation discussed in this section can be applied to interpret different attention-based networks.

Concept-Based Explanation

The corpus-level explanation still suffers from the drawback that it cannot automatically obtain higher-level concepts/clusters for those important keywords. To alleviate this problem, we propose concept-based explanation for our AAN model. In AAN, each abstraction-attention unit can capture one concept/cluster. Here, we will take distribution of concepts into consideration. Formally, we express $p_\theta(w_t|y = l)$ as follows:

$$p_\theta(w|y = l) = \sum_{k=1}^{K} p_\theta(w|c_k, y = l) p_\theta(c_k|y = l)$$

where $p_\theta(w|c_k, y = l)$ captures the distribution of w across $\mathcal{C}_{\text{train}}$ for the k^{th} concept and label l, while $p_\theta(c_k|y = l)$ captures the distribution of the concept c_k across $\mathcal{C}_{\text{train}}$ for label l. They can be computed using the following equations.

$$\begin{aligned} p_\theta(w|c_k, y = l) &= \sum_{\xi \in \mathcal{C}^l_{\text{train}}} p_\theta(w, \xi|c_k, y = l), \\ p_\theta(c_k|y = l) &= \sum_{\xi \in \mathcal{C}^l_{\text{train}}} p_\theta(c_k, \xi|y = l), \end{aligned} \tag{5.6}$$

where we define

$$p_\theta(w, \xi|c_k, y = l) := \frac{\sum_{t=1}^{T} \alpha_{k,t}^{\text{abs},\xi} \cdot \delta(w_t, w)}{\sum_{\xi' \in \mathcal{C}_{\text{train}}} f_{\xi'}(w) + \gamma} \tag{5.7}$$

and

$$p_\theta(c_k, \xi|y = l) := \frac{\alpha_k^{\text{agg},\xi}}{|\mathcal{C}_{\text{train}}|}, \tag{5.8}$$

where $\alpha_{k,t}^{\text{abs},\xi}$ represents $\alpha_{k,t}^{\text{abs}}$ for document ξ. Based on Eq. (5.6), we are able to obtain scores (importance) and most relevant keywords for different concepts for a given label l.

Consistency Analysis

In corpus-level and concept-based explanations, we have obtained causal relationships between keywords and predictions, i.e., $p_\theta(w|y = l)$. However, we have not verified if these keywords are really important to predictions. To achieve this goal, we build a Naïve Bayes classifier [30] (NBC) based on these causal relationships. Formally, for each testing document d, the probability of getting label l is approximated as follows:

$$\begin{aligned} p_\theta(y = l|d) &= \frac{p_\theta(d|y = l)p_\theta(y = l)}{p(d)} \\ &\propto p_\theta(d|y = l) = \prod_{t=1}^{T} p_\theta(w_t|y = l), \end{aligned} \tag{5.9}$$

where $p_\theta(w_t|y = l)$ is obtained by Eq. (5.4) or Eq. (5.6) on the training corpus. We further approximate Eq. (5.9) with

$$p_\theta(y = l|d) = \prod_{w \in d'} (p_\theta(w|y = l) + \lambda), \tag{5.10}$$

where $d' \subset d$ is an *attention-based bag-of-words representation* for document d. It consists of important keywords based on attention weights. λ is a smoothing factor. Here, we can conduct consistency analysis by comparing labels obtained by the model and NBC, which may also help estimate the uncertainty of a model [170].

5.3 Experiments

5.3.1 Datasets

We conducted experiments on three publicly available datasets. Newsroom is used for news categorization, while IMDB and Beauty are used for sentiment analysis. The details of the three datasets are as follows:

- **Newsroom** [40]: The original dataset, which consists of 1.3 million news articles, was proposed for text summarization. In our experiments, we first determined the category of each article based on the URL, and then, randomly sampled 10,000 articles for each of the five categories, including business, entertainment, sports, health, and technology [57, 123, 126].
- **IMDB** [89]: This dataset contains 50,000 movie reviews from the IMDB website with binary (positive or negative) labels.
- **Beauty** [44]: This dataset contains product reviews in the beauty category from Amazon. We converted the original ratings (1-5) to binary (positive or negative) labels and sampled 20,000 reviews for each label.

Table 5.1: Statistics of the datasets used.

Dataset	**#docs**	**Avg. Length**	**Scale**
Newsroom	50,000	827	1-5
IMDB	50,000	292	1-2
Beauty	40,000	91	1-2

For all three datasets, we tokenized reviews using the BERT tokenizer [153] and randomly split them into train/development/test sets with a proportion of 8:1:1. Statistics of the datasets are summarized in Table 5.1.

5.3.2 Models and Implementation Details

We compare different classification models including several baselines, variants of our AAN model, and Naïve Bayes classifiers driven by a basic self-attention network (SAN) [121] and AAN.

- **CNN** [61]: This model extracts key features from a review by applying convolution and max-over-time pooling operations [23] over the shared word embedding layer.
- **LSTM-SAN**, **BERT-SAN**, **DistilBERT-SAN**, **RoBERTa-SAN**, and **Longformer-SAN**: All these models are based on the SAN framework. In LSTM-SAN, the encoder consists of a word embedding layer and a Bi-LSTM encoding layer, where embeddings are pre-loaded with 300-dimensional GloVe vectors [106] and fixed during training. BERT [153], DistilBERT [119], RoBERTa [83], and Longformer [5] leverage different pre-trained language models, which have 110M, 66M, 125M, 125M parameters, respectively.
- **AAN + C(c) + Drop(r)**: These are variants of AAN. C(c) and Drop(r) represent the number of concepts and dropout rate, respectively.

We implemented all deep learning models using PyTorch [105] and the best set of parameters are selected based on the development set. For CNN based models, the filter sizes are chosen to be 3, 4, and 5 and the number of filters is set to 100 for each size. For LSTM based models, the dimension of hidden states is set to 300 and the number of layers is 2. All parameters are trained with the ADAM optimizer [63] with a learning rate of 0.0002. Dropout with a rate of 0.1 is also applied in the classification layer. For all explanation tasks, we set the number of concepts to 10 and dropout-rate to 0.02. Our codes and datasets are available at `https://github.com/tshi04/ACCE`.

Table 5.2: Averaged accuracy of different models on Newsroom, IMDB, and Beauty testing sets.

Model	**Newsroom**	**IMDB**	**Beauty**
CNN	90.18	88.56	88.42
LSTM-SAN	91.26	90.68	92.00
BERT-SAN	92.28	92.60	93.72
DistilBERT-SAN	**92.66**	92.52	92.82
RoBERTa-SAN	91.16	92.76	93.40
Longformer-SAN	92.04	**93.74**	**94.50**

Table 5.3: Averaged accuracy of BERT and Longformer-based AAN models on Newsroom, IMDB, and Beauty testing sets.

	Newsroom		IMDB		Beauty	
	BERT	Longformer	BERT	Longformer	BERT	Longformer
SAN Framework	92.28	92.04	92.60	93.74	93.72	94.50
AAN + C(10) + Drop(0.01)	92.54	91.72	92.22	92.96	93.38	93.42
AAN + C(10) + Drop(0.02)	92.14	91.64	92.14	92.86	93.58	93.75
AAN + C(10) + Drop(0.05)	92.14	91.60	91.82	92.66	93.05	93.80
AAN + C(10) + Drop(0.10)	92.30	91.48	91.50	92.12	93.25	93.60
AAN + C(20) + Drop(0.01)	92.02	91.98	91.64	92.78	93.70	93.48
AAN + C(20) + Drop(0.02)	92.44	91.84	91.80	93.04	93.55	93.88
AAN + C(20) + Drop(0.05)	92.54	91.86	91.92	93.14	93.68	93.42
AAN + C(20) + Drop(0.10)	92.52	91.98	92.10	92.96	93.72	93.88

5.3.3 Performance Results

We use accuracy as evaluation metric to measure the performance of different models. All quantitative results have been summarized in Tables 5.2 and 5.3, where we use bold font to highlight the highest accuracy on testing sets in Table 5.2. Comparing LSTM-SAN with BERT, DistilBERT, RoBERTa and Longformer, we first find that different pre-trained language model-based encoders are better than the conventional LSTM encoder with pre-trained word embeddings. In Table 5.3, we replace self-attention on top of pre-trained language models with the abstraction-aggregation network (AAN). We observe that different AAN models do not significantly lower the classification accuracy, which indicates we can use AAN for the concept-based explanation task without losing the overall performance. Here, the strategy of aggregation-attention weights dropout is necessary when training AAN models. In Table 5.9, we show that AAN models without randomly dropping aggregation-attention weights attain poor interpretability in concept-based explanation.

Table 5.4: Case-level concept-based explanation. Here, each ID is associated with a concept, i.e., abstraction-attention unit. Scores and weights (following each keyword) are calculated with Eq. (5.7) and (5.8). '-' represents special characters.

ID	Score	Keywords
8	0.180	com(0.27), boston(0.26), boston(0.16), boston(0.1), m(0.02)
6	0.162	marketing(0.28), ad(0.06), ##fs(0.05), investors(0.03), said(0.03)
1	0.148	campaign(0.14), firm(0.14), money(0.06), brand(0.04), economist(0.03)
2	0.122	economist(0.2), said(0.16), professional(0.16), investors(0.08), agency(0.06)
9	0.116	boston(0.89), boston(0.11)
7	0.108	bloomberg(0.16), cn(0.11), global(0.09), money(0.06), cable(0.05)
5	0.103	-(0.96), -(0.03), s(0.01)
4	0.047	investment(0.76), money(0.14), investment(0.06), investment(0.02), investors(0.01)
3	0.016	,(0.64), -(0.36)
10	0.000	.(0.93), -(0.07)

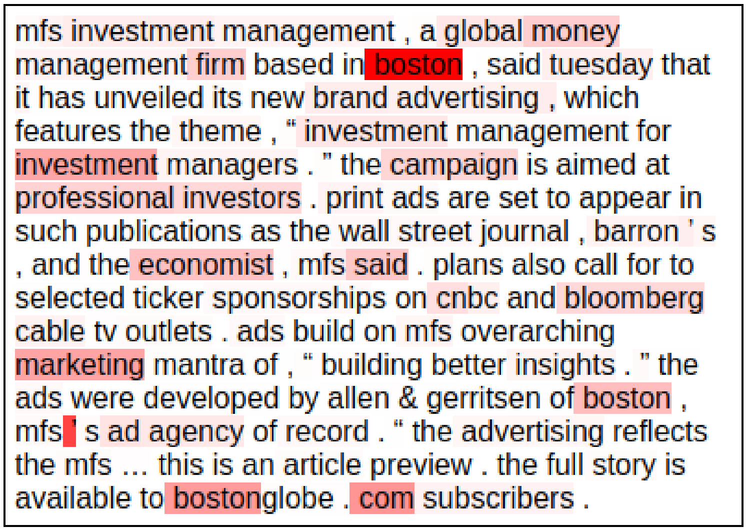

mfs investment management , a global money management firm based in boston , said tuesday that it has unveiled its new brand advertising , which features the theme , " investment management for investment managers . " the campaign is aimed at professional investors . print ads are set to appear in such publications as the wall street journal , barron ' s , and the economist , mfs said . plans also call for to selected ticker sponsorships on cnbc and bloomberg cable tv outlets . ads build on mfs overarching marketing mantra of , " building better insights . " the ads were developed by allen & gerritsen of boston , mfs ' s ad agency of record . " the advertising reflects the mfs ... this is an article preview . the full story is available to bostonglobe . com subscribers .

Figure 5.2: Attention-weight visualization for an interpretable attention-based classification model.

5.3.4 Heat-maps and Case-level Concept-based Explanation

First, we investigate if AAN attends to relevant keywords when it is making predictions, which can be accomplished by visualizing attention weights (see Fig. 5.2). This is a *Business* news article from Newsroom and we observe that the most relevant keyword that AAN detects is *boston*. Other important keywords include *investment*, *economist*, *marketing* and *com*. Compared with Fig. 5.2, our *case-level concept-based explanation* provides more informative results. From Table 5.4, we observe that AAN makes the prediction based on several different aspects, such as corporations (e.g., com), occupations (e.g., economist), terminology (e.g., marketing) and so on. Moreover, *boston* may be related with corporation (e.g., bostonglobe or gerritsen of boston) or city, thus, it appears in both concepts 8 (corporations) and 9 (locations).

5.3.5 Corpus-Level Explanation

Corpus-level explanation aims to find the important keywords for the predictions. In Table 5.5, we show 20 most important keywords for each predicted label and we assume these keywords determine the predictions. In the last section, we will demonstrate this assumption by the consistency analysis. The scores of keywords have been shown in Fig. 5.3.

In addition to causal relationships, we can also use these keywords to check if our model and datasets have bias or not. For example, *boston* and *massachusetts* plays an important role in predicting business, which indicates the training set has bias. By checking our data, we find that many business news articles are from *The Boston Globe*. Another obvious bias example is that the numbers *8, 7,* and *9* are important keywords for IMDB sentiment analysis. This is because the original ratings scale from 1 to 10 and many reviews mention that "*rate this movie 8 out of 10*".

Moreover, from Fig. 5.3 (a) and (b), we find that for a randomly split corpus, distributions of keywords across training/development/test sets are similar to each other. This guarantees the model achieves outstanding performance on testing sets. If we apply a model trained on IMDB to Beauty (see Fig. 5.3 (c)), it can only leverage the cross-domain common keywords (e.g., *disappointed* and *loved*) to make predictions. However, we achieve 71% accuracy, which is much better than random predictions. In Table 5.5, we use bold font to highlight these common keywords.

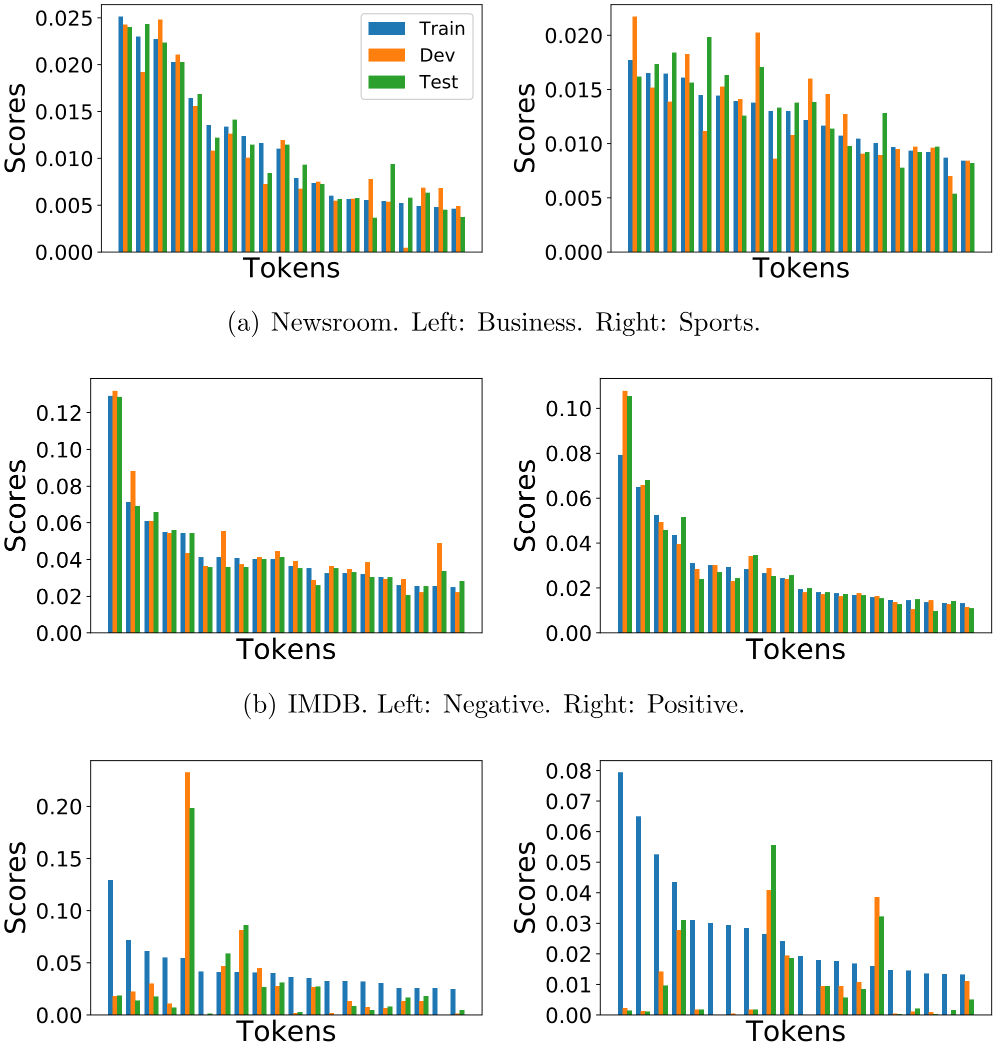
5.3.6 Corpus-level Concept-based Explanation

Corpus-level concept-based explanation further improves the corpus-level explanation by introducing clustering structures to keywords. In this section, we still use the AAN trained on Newsroom as an example for this task. Table 5.6 shows concepts and relevant keywords for AAN when it assigns an article to *Business*. Here, we observe that the first-tier salient concepts consist of concepts 8 (corporations) and 1 (business terminology in general). The second-tier concepts 7, 6, and 4 are related to economy, finance, mortgage, and banking, which are domain-specific terminology. They share many keywords. Concepts 9 and 2 are

Table 5.5: This table shows 20 most important keywords for model predictions on different training sets. Keywords are ordered by their scores. For Newsroom, we only show 2 out of 5 classes due to space limitations.

Dataset	**Label**	**Keywords**
IMDB	Negative	worst, awful, terrible, bad, **disappointed**, boring, **disappointing**, **waste**, **horrible**, sucks, fails, disappointment, lame, dull, poorly, poor, worse, mess, dreadful, pointless
	Positive	8, 7, **excellent**, **loved**, 9, enjoyable, superb, enjoyed, **highly**, **wonderful**, entertaining, best, beautifully, good, **great**, brilliant, terrific, funny, hilarious, fine
Beauty	Negative	disappointed, nothing, unfortunately, made, not, waste, disappointing, terrible, worst, horrible, makes, no, sadly, disappointment, t, awful, sad, bad, never, started
	Positive	great, love, highly, amazing, pleased, perfect, works, best, happy, awesome, makes, recommend, excellent, wonderful, definitely, good, glad, well, fantastic, very
Newsroom	Business	inc, corp, boston, massachusetts, economic, cambridge, financial, economy, banking, auto, automotive, startup, company, mr, finance, biotechnology, somerville, retailer, business, airline
	Entertainment	singer, actress, actor, star, fox, comedian, hollywood, sunday, rapper, fashion, celebrity, contestant, filmmaker, bachelor, insider, porn, oscar, rocker, host, monday
	Sports	quarterback, coach, basketball, baseball, soccer, nba, sports, striker, tennis, hockey, nfl, nhl, football, olympic, midfielder, golf, player, manager, outfielder, nascar
	Health	dr, health, pediatric, obesity, cardiovascular, scientists, researcher, medicine, psychologist, diabetes, medical, psychiatry, aids, fitness, healthcare, autism, psychology, neuroscience, fox, tobacco
	Technology	tech, cyber, electronics, wireless, lifestyle, silicon, gaming, culture, telecommunications, scientist, company, google, smartphone, technology, francisco, broadband, privacy, internet, twitter

associated with locations and occupations, respectively, which receive relatively lower scores. Concepts 5, 3 and 10 are not quite meaningful. We have also shown results for Newsroom sports in Table 5.7, where we find that 1 (sports terminology) and 7 (leagues and teams) are the first-tier salient concepts. The second-tier salient concepts 6 and 4 are about games and campaigns. Concepts 7, 6, 4 also share many keywords. Concepts 8 (corporations and channels), 2 (occupations and roles) and 9 (locations) are the third-tier salient concepts. Concepts 5, 3, 10 are also meaningless. From these tables, we summarize some commonalities: 1) Domain-specific terminologies (i.e., concepts 1, 7, 6 and 4) play an important role in predictions. 2) Locations (i.e., concept 9) and Occupations/Roles (i.e., concept 2) are less important. 3) Meaningless concepts (i.e., concepts 5, 3, and 10), such as punctuation, have the least influence.



(a) Newsroom. Left: Business. Right: Sports.

(b) IMDB. Left: Negative. Right: Positive.

(c) IMDB-to-Beauty. Left: Negative. Right: Positive.

Figure 5.3: Distribution of keywords on training, development and testing sets. Scores are calculated by $p_\theta(w, l, \mathcal{C})$. The orders of tokens are the same as those in Table 5.5.

5.3.7 Consistency Analysis

In this section, we leverage the method proposed in Section 5.2.2 to respectively build a NBC for BERT-SAN and BERT-AAN on the training set. Then, we apply them to the testing set to compare if NBC predictions and the model predictions are consistent with each other. We approximate the numerator of Eq. (5.5) with five words (can repeat) with highest attention weights in each document. In Eq. (5.4), γ is set to be 1000. In Eq. (5.10), we set $\lambda = 1.2$ for text categorization and $\lambda = 1.0$ for sentiment analysis. d' consists of five words with highest attention weights.

Table 5.6: Concept-based explanation (Business). Scores are calculated using Eq. (5.6).

ID	**Score**	**Keywords**
8	0.173	inc, corp, massachusetts, boston, mr, ms, jr, ltd, mit, q
1	0.168	economy, retailer, company, startup, ##maker, airline, chain, bank, utility, billionaire
7	0.151	biotechnology, banking, tech, startup, pharmaceuticals, mortgage, financial, auto, commerce, economic
6	0.124	economic, health, banking, finance, insurance, healthcare, economy, housing, safety, commerce
4	0.107	financial, economic, banking, auto, automotive, securities, housing, finance, monetary, biotechnology
9	0.086	boston, massachusetts, cambridge, washington, detroit, frankfurt, harvard, tokyo, providence, paris
5	0.056	-, ##as, -, -, itunes, inc, corp, northeast, -, llc
2	0.054	economist, executive, spokesman, analyst, economists, ##gist, ceo, director, analysts, president
3	0.026	-, -, -,), ##tem, ##sp, the, =, t, ob
10	0.000	-, comment,), insurance, search, ’, tesla, graphic, guitarist, ,

Table 5.7: Concept-based explanation (Sports). Scores are calculated using Eq. (5.6).

ID	**Score**	**Keywords**
1	0.176	quarterback, player, striker, champion, pitcher, midfielder, outfielder, athlete, goaltender, forward
7	0.165	nhl, mets, soccer, nets, yankees, nascar, mls, reuters, doping, twitter
6	0.147	tennis, sports, soccer, golf, doping, hockey, athletic, athletics, injuries, basketball
4	0.139	baseball, basketball, nba, nfl, sports, football, tennis, olympic, hockey, golf
8	0.119	jr, ”, n, j, fox, espn, nl, u, boston, ca
2	0.100	coach, manager, commissioner, boss, gm, trainer, spokesman, umpire, coordinator, referee
9	0.060	philadelphia, indianapolis, boston, tampa, louisville, buffalo, melbourne, manchester, baltimore, atlanta
5	0.055	’, ’, ##as, –, ##a, ‘, sides, newcomers, chelsea, jaguars
3	0.022	’, ’,), ,, ##kus, ##gre, the, whole, lever, ##wa
10	0.000	.,), finishes, bel, gymnastics, ’, ##ditional, becomes, tu, united

We use the accuracy (consistency score) between labels predicted by NBC and the original model to evaluate the consistency. Table 5.8 shows that around 85% of predictions are consistent. This demonstrates that keywords obtained by the corpus-level and concept-based explanation methods are important to predictions. They can be used to interpret attention based models. Moreover, from CP and NCP scores, we observe a significantly higher probability that the model makes an incorrect prediction if it is inconsistent with NBC prediction. This finding suggests us to use consistency score as one criterion for *uncertainty estimation*.

Table 5.8: Consistency between the model and NBC. CS represents consistency score, CP/NCP denote percentage of incorrect predictions when NBC predictions are consistent/not consistent with model predictions.

Model	**Newsroom**			**IMDB**			**Beauty**		
	CS	NCP	CP	CS	NCP	CP	CS	NCP	CP
BERT-SAN	83.96	21.59	4.72	86.02	17.17	5.81	85.45	16.30	4.56
BERT-AAN	84.36	20.20	5.57	85.46	21.18	5.05	84.72	16.04	4.51

Table 5.9: Concept-based explanation (Sports) for AAN without applying dropout to attention weights.

CID	**Weight**	**Keywords**
1	0.8195	quarterback, athletic, olympic, basketball, athletics, qb, hockey, outfielder, sports
7	0.0865	nascar, celtics, motorsports, nba, boston, augusta, nhl, tennis, leafs, zurich
4	0.0370	mets, knicks, yankees, players, pitchers, lakers, hosts, coaches, forwards, swimmers
3	0.0164	offensive, eli, bird, doping, nba, jay, rod, hurdle, afc, peyton
2	0.0098	premier, american, mets, nl, field, yankee, national, aaron, nba, olympic
10	0.0083	games, seasons, tries, defeats, baskets, players, season, contests, points, throws
5	0.0015	dustin, antonio, rookie, dante, dale, dylan, lineman, ty, launch, luther
8	0.0010	2016, 2014, college, tribune, card, press, s, -, this, leadership
9	0.0004	men, -, grand, 9, s, usa, state, west, world, major
6	0.0000	the, -, -, year, whole, vie, very, tr, too, to

5.3.8 Dropout of Aggregation-Attention Weights

For AAN, we apply dropout to aggregation-attention weights during training. In Table 5.9, we show an example without using the attention weight dropout mechanism. We observed that the weight for concept 1 is much higher than the other concepts. In addition, keywords for each concept are not semantically coherent.

5.4 Summary

In this study, we proposed a general-purpose *corpus-level explanation* approach to interpret attention-based networks. It can capture causal relationships between keywords and model predictions via learning importance of keywords for predicted labels across the training corpus based on attention weights. Experimental results show that the keywords are semantically meaningful for predicted labels. We further propose a *concept-based explanation* method to identify important concepts for model predictions. This method is based on a novel

Abstraction-Aggregation Network (AAN), which can automatically extract concepts, i.e., clusters of keywords, during the end-to-end training for the main task. Our experimental results also demonstrate that this method effectively captures semantically meaningful concepts/clusters. It also provides relative importance of each concept to model predictions. To verify our results, we also built a *Naïve Bayes Classifier* based on an *attention-based bag-of-words document representation* technique and the causal relationships. Consistency analysis results demonstrate that the discovered keywords are important to the predictions.

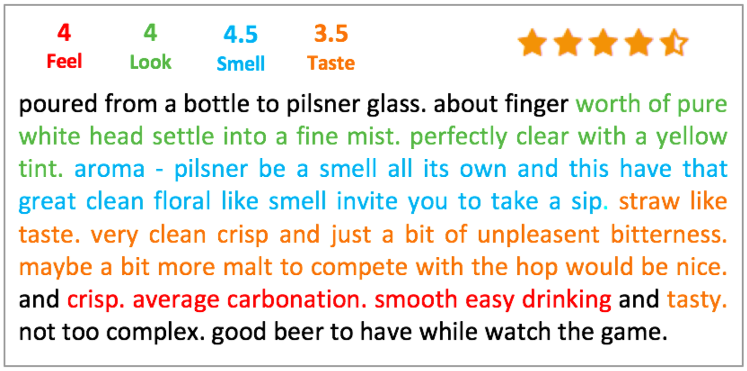
Chapter 6

An Interpretable and Uncertainty Aware Multi-Task Framework for Multi-Aspect Sentiment Analysis

This chapter introduces a deliberate self-attention based deep neural network model for the document-level multi-aspect sentiment analysis problem. An attention-driven keywords ranking method has also been proposed to automatically discover aspect keywords and aspect-level opinion keywords from review corpora based on the attention weights. In addition, we develop a lecture-audience method to estimate model uncertainty in the context of multi-task learning. The rest of this chapter is organized as follows: The introduction of this chapter is presented in Section 6.1. In Section 6.2, we present details of our proposed model, attention-driven keywords extraction method and lecture-audience uncertainty estimation approach. In Section 6.3, we introduce different benchmark datasets, baseline methods and implementation details, as well as analyze experimental results. Our discussion concludes in Section 6.4.

6.1 Background and Motivation

Sentiment analysis plays an important role in many business applications [103]. It is used to identify customers' opinions and emotions toward a particular product/service via identifying polarity (i.e., positive, neutral or negative) of given textual reviews [77, 104]. In the past few years, with the rapid growth of online reviews, the topic of fine-grained aspect-based sentiment analysis (ABSA) [108] has attracted significant attention since it allows models to predict opinion polarities with respect to aspect-specific terms in a sentence. Different from sentence-level ABSA, document-level multi-aspect sentiment classification (DMSC) aims to predict the sentiment polarity of documents, which are composed of several sentences, with respect to a given aspect [163, 74, 167]. DMSC has become a significant challenge since



4 Feel | 4 Look | 4.5 Smell | 3.5 Taste ★★★★⯪

poured from a bottle to pilsner glass. about finger worth of pure white head settle into a fine mist. perfectly clear with a yellow tint. aroma - pilsner be a smell all its own and this have that great clean floral like smell invite you to take a sip. straw like taste. very clean crisp and just a bit of unpleasent bitterness. maybe a bit more malt to compete with the hop would be nice. and crisp. average carbonation. smooth easy drinking and tasty. not too complex. good beer to have while watch the game.

Figure 6.1: An example of an online review from the BeerAdvocate platform. Keywords corresponding to different aspects are highlighted with different colors.

many websites provide platforms for users to give aspect-level feedback and ratings, such as TripAdvisor[1] and BeerAdvocate[2]. Fig. 6.1 shows a review example from the BeerAdvocate website. In this example, a beer is rated with four different aspects, i.e., feel, look, smell and taste. The review also describes the beer with four different aspects. There is an overall rating associated with this review. Recent studies have found that users are less motivated to give aspect-level ratings [163, 167], which makes it difficult to analyze their preference, and it takes a lot of time and effort for human experts to manually annotate them.

There are several recent studies that aim to predict the aspect ratings or opinion polarities using deep neural network based models with a multi-task learning framework [163, 74, 169, 167]. In this setting, rating predictions for different aspects, which are highly correlated and can share the same review encoder, are treated as different tasks. However, these models rely on hand-crafted aspect keywords to aid in rating/sentiment predictions [163, 74, 169]. Thus, their results, especially case studies of reviews, are biased towards pre-defined aspect keywords. In addition, these models only focus on improving the prediction accuracy, however, knowledge discovery (such as aspect and opinion related keywords) from review corpora still relies on unsupervised [94] and rule-based methods [167], which limits applications of current DMSC models [163, 74, 169]. In the past few years, model uncertainty of deep neural network classifiers has received increasing attention [32, 31], because it can identify low-confidence regions of input space and give more reliable predictions. Uncertainty models have also been applied to deep neural networks for text classification [170]. However, few existing uncertainty methods have been used to improve the overall prediction accuracy of multi-task learning models when crowd-sourcing annotation is involved in the DMSC task. In this study, we attempt to tackle the above mentioned issues. The primary contributions of this study are as follows:

- Develop a FEDAR model that achieves competitive results on five benchmark datasets

[1] https://www.tripadvisor.com
[2] https://www.beeradvocate.com

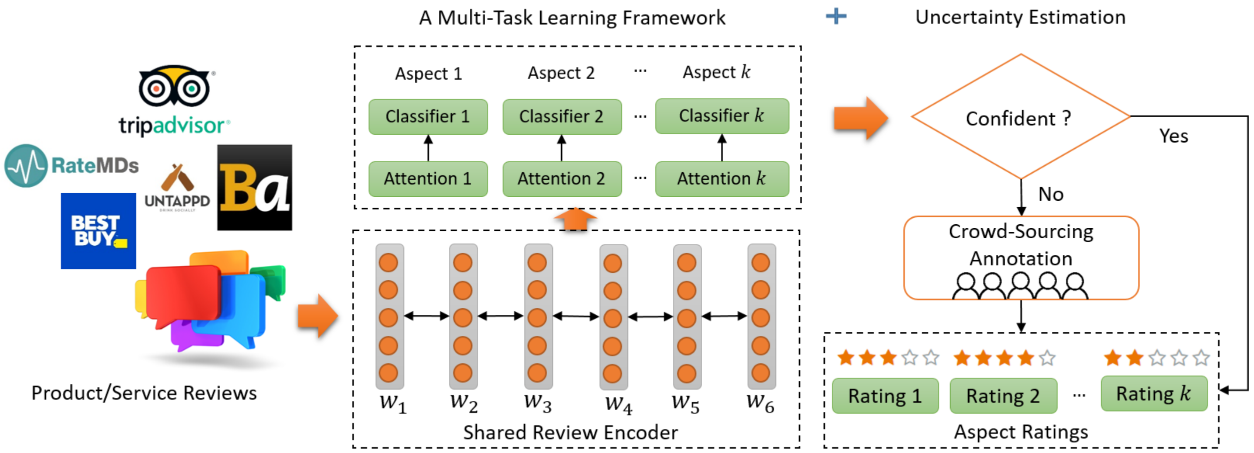


Figure 6.2: An overview of our multi-task learning framework with uncertainty estimation for accurate and reliable sentiment classification in DMSC task. Here, sentiment classification for each aspect is treated as a task and different tasks share the same review encoder.

without using hand-crafted aspect keywords. The proposed model is equipped with a highway word embedding layer, a sequential encoder layer whose output features are enriched by pooling and factorization techniques, and a deliberate self-attention layer. The deliberate self-attention layer can boost performance as well as provide interpretability for our FEDAR model. Here, FEDAR represents some key components of our model, including Feature Enrichment, Deliberate self-Attention, and overall Rating.

- Introduce two new datasets obtained from the RateMDs website `https://www.ratemds.com`, which is a platform for patients to review the performance of their doctors. We benchmark different models on them.
- Propose an Attention-driven Keywords Ranking (AKR) method to automatically discover aspect and opinion keywords from review corpora based on attention weights, which also provides a new research direction for interpreting self-attention mechanism. The extracted keywords are significant to ratings/polarities predicted by FEDAR.
- Propose a LEcture-Audience (LEAD) method to measure the uncertainty of our FEDAR model for given reviews. This method can also be generally applied to other deep neural networks.

6.2 Proposed Methods

In this section, we first introduce our FEDAR model (see Fig. 6.3) for the DMSC task. Then, we describe our AKR method to automatically discover aspect and aspect-level sentiment terms based on the FEDAR model. Finally, we discuss our LEAD method (see Fig. 6.4) for measuring the uncertainty of the FEDAR model.

6.2.1 The Proposed FEDAR Model

Problem Formulation

The DMSC problem can be formulated as a multi-task classification problem, where the sentiment classification for each aspect is viewed as a task (see Fig. 6.2). More formally, the DMSC problem is described as follows: Given a textual review $X = (x_1, x_2, ..., x_T)$, our goal is to predict class labels, i.e., integer ratings/sentiment polarity of the review $y = (y^1, y^2, ..., y^K)$, where T and K are the number of tokens in the review and the number of aspects/tasks, respectively. x_t and y^k are the one-hot vector representations of word t and the class label of aspect k, respectively.

The challenge in this problem is to build a model that can achieve competitive accuracy without losing model interpretability or obtaining biased results. Therefore, we propose improving word embedding, review encoder and self-attention layers to accomplish this goal. We will now introduce our model and provide more details of our architecture in a layer-by-layer manner.

Highway Word Embedding Layer

This layer aims to learn word vectors based on pre-trained word embeddings. We first use a word embedding technique [96] to map one-hot representations of tokens $x_1, x_2, ..., x_T$ to a continuous vector space, thus, they are represented as $E_{x_1}, E_{x_2}, ..., E_{x_T}$, where E_{x_t} is the word vector of x_t, pre-trained on a large corpus and fixed during parameter inference. In our experiments, we adopted GloVe word vectors [106], so that they do not need to be trained from random states, which may result in poor embeddings due to the lack of word co-occurrence.

Then, a single layer highway network [134] is used to adapt the knowledge, i.e., semantic information from pre-trained word embeddings, to target DMSC datasets. Formally, the highway network is defined as follows:

$$E'_{x_t} = f(E_{x_t}) \odot g(E_{x_t}) + E_{x_t} \odot (1 - g(E_{x_t})) \tag{6.1}$$

where $f(\cdot)$ and $g(\cdot)$ are affine transformations with ReLU and Sigmoid activation functions, respectively. $\odot$ represents element-wise product. $g(\cdot)$ is also known as gate, which is used to control the information that is being carried to the next layer. Intuitively, the highway network aims at transferring knowledge from pre-trained word embeddings to the target review corpus. E'_{x_t} can be viewed as a perturbation of E_{x_t}, and $f(\cdot)$ and $g(\cdot)$ have significantly fewer parameters than E_{x_t}. Therefore, training a highway network is more efficient than training a word embedding layer from random parameters.

Review Encoder Layer

This layer describes the review encoder and feature enrichment techniques proposed in our model.

Sequential Encoder Layer: The output of the highway word embedding layer $(E'_{x_1}, E'_{x_2}, ..., E'_{x_T})$ is fed into a sequential encoder layer. Here, we adopt a multi-layer bi-directional LSTM encoder [47], which encodes a review into a sequence of hidden states in forward direction $\overrightarrow{H} = (\overrightarrow{h_1}, \overrightarrow{h_2}, ..., \overrightarrow{h_T})$ and backward direction $\overleftarrow{H} = (\overleftarrow{h_1}, \overleftarrow{h_2}, ..., \overleftarrow{h_T})$.

Representative Features: For each hidden state $\overrightarrow{h_t}$ (or $\overleftarrow{h_t}$), we generate three representative features, which will be later used to assist the attention mechanism to learn the overall review representation.

The first and second features, denoted by $\overrightarrow{h_t^{\max}}$ and $\overrightarrow{h_t^{\text{avg}}}$, are the max-pooling and average-pooling of $\overrightarrow{h_t}$, respectively. The third one is obtained using a factorization machine [116], where the factorization operation is defined as

$$\mathcal{F}(z) = w_0 + \sum_{i=1}^{N} w_i z_i + \sum_{i=1}^{N} \sum_{j=i+1}^{N} \langle V_i, V_j \rangle \, z_i z_j. \tag{6.2}$$

Here, the model parameters are $w_i \in \mathbb{R}$ and $V \in \mathbb{R}^{N \times F}$. N and F are the dimensions of the input vector z and factorization, respectively. $\langle \cdot, \cdot \rangle$ is the dot product between two vectors. w_0 in Eq. (6.2) is a global bias, w_i is the strength of the i-th variable, and $\langle V_i, V_j \rangle$ captures the pairwise interaction between z_i and z_j.

Intuitively, the max-pooling and avg-pooling provide the approximated location (bound and mean) of the hidden state $\overrightarrow{h_t}$ in the N dimensional space, while the factorization captures all single and pairwise interactions. Together they provide the high-level knowledge of that hidden state.

Feature Augmentation: Finally, the aggregated hidden state h_t at time step t is obtained by concatenating hidden states in both directions and all representative features, i.e.,

$$\begin{aligned} \overrightarrow{h_t} &= \overrightarrow{h_t} \oplus \overrightarrow{h_t^{\max}} \oplus \overrightarrow{h_t^{\text{avg}}} \oplus \mathcal{F}(\overrightarrow{h_t}), \\ \overleftarrow{h_t} &= \overleftarrow{h_t} \oplus \overleftarrow{h_t^{\max}} \oplus \overleftarrow{h_t^{\text{avg}}} \oplus \mathcal{F}(\overleftarrow{h_t}), \\ h_t &= \overrightarrow{h_t} \oplus \overleftarrow{h_t}. \end{aligned} \tag{6.3}$$

Thus, the review is encoded into a sequence of aggregated hidden states $H = (h_1, h_2, \ldots, h_T)$.

Deliberate Self-Attention Layer

Once the aggregated hidden states for each review are obtained, we apply a self-attention layer for each task to learn an overall review representation for that task. Compared with

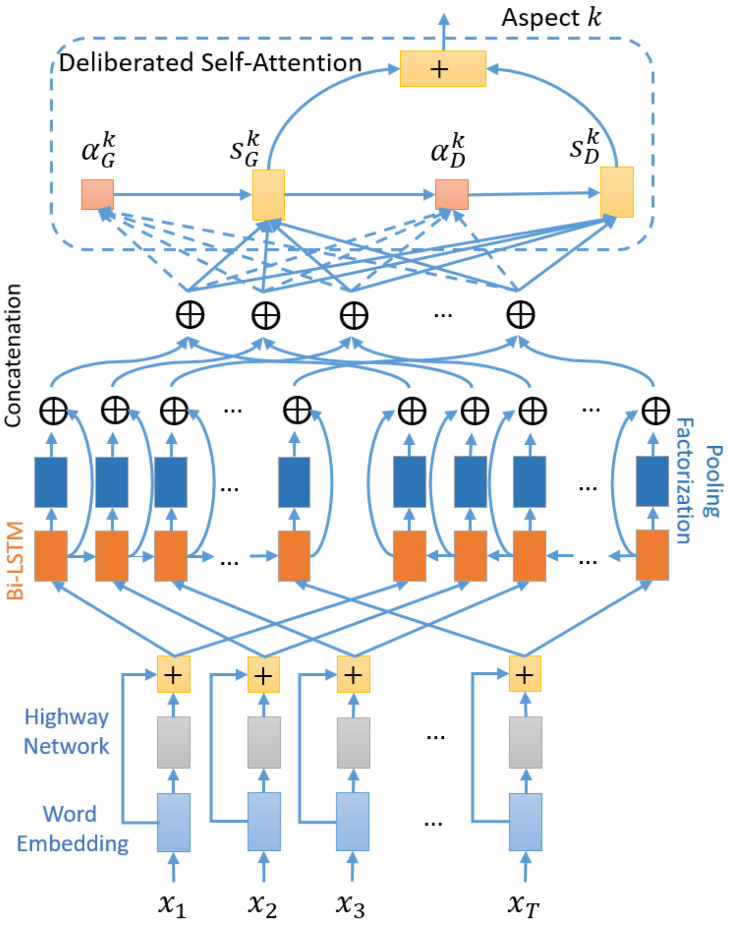


Figure 6.3: Review encoder and deliberate self-attention for aspect k. Each hidden state is enriched by three features, i.e., max-pooling, average-pooling, and factorization.

pooling and convolution operations, the self-attention mechanism is more interpretable, since it can capture relatively important words for a given task. However, a standard self-attention layer merely relies on a single global alignment vector across different reviews, which results in sub-optimal representations. Therefore, we propose a deliberate self-attention alignment method to refine the review representations while maintaining the network interpretability. In this section, we will first introduce the self-attention mechanism, and then provide the details of the deliberation counterpart.

Global Self-Attention: For each aspect k, the self-attention mechanism [161] is used to learn the relative importance of tokens in a review to the sentiment classification task. Formally, given the aggregated hidden states H for a review, the alignment score $u_{t,G}^k$ and attention weight $\alpha_{t,G}^k$ are calculated as follows:

$$u_{t,G}^k = (v_G^k)^\top \tanh(W_G^k h_t + b_G^k),\ \alpha_{t,G}^k = \frac{\exp(u_{t,G}^k)}{\sum_{\tau=1}^T \exp(u_{\tau,G}^k)}, \tag{6.4}$$

where W_G^k, v_G^k and b_G^k are model parameters. G represents global, as the above attention mechanism is also known as global attention [88]. v_{G}^k is viewed as a global aspect-specific base-vector in this study, since it has been used in calculating the alignment with different hidden states across different reviews. It can also be viewed as a global aspect-specific filter that is designed to capture important information for a certain aspect from different reviews. Therefore, we also call the regular self-attention layer as the global self-attention layer. With attention weights, the global review representation is calculated by taking the weighted sum of all aggregated hidden states, i.e., $s_G^k = \sum_{t=1}^T \alpha_{t,G}^k h_t$. Traditionally, s_G^k is used for the sentiment classification task.

Deliberate Attention: As we can see from Eq. (6.4), the importance of a token t is measured by the similarity between $\tanh(W_G^k h_t + b_G^k)$ and the base-vector v_G^k. However, a single base-vector v_G^k is difficult to capture the variability in the reviews, and hence, such alignment results in sub-optimal representations of reviews. In this study, we attempt to alleviate this problem by reusing the output of the global self-attention, i.e., s_G^k, as a document-level aspect-specific base-vector to produce better review representations. Notably, s_G^k already incorporates the knowledge of the review content and aspect k. We refer this step as deliberation.

Given the hidden states H and review representation s_G^k, we first calculate the alignment scores and attention weights as follows:

$$u_{t,D}^k = (s_G^k)^\top \tanh(W_D^k h_t + b_D^k),\ \alpha_{t,D}^k = \frac{\exp(u_{t,D}^k)}{\sum_{\tau=1}^T \exp(u_{\tau,D}^k)}, \tag{6.5}$$

where W_D^k and b_D^k are parameters. D represents deliberation. Similarly, we can calculate the aspect-specific review representation by deliberation as $s_D^k = \sum_{t=1}^T \alpha_{t,D}^k h_t$.

Review Representation: Finally, the review representation for aspect k can be obtained as follows[3]:

$$s^k = s_G^k + s_D^k = \sum_{t=1}^T \big(\alpha_{t,G}^k + \alpha_{t,D}^k\big) h_t. \tag{6.6}$$

From the above equation, we not only get refined review representations but also maintain the interpretability of our model. Here, we did not use the concatenation of two vectors since we would like to maintain the interpretability as well. Notably, we can use the accumulated attention weights, i.e., $\frac{1}{2}(\alpha_{t,G}^k + \alpha_{t,D}^k)$, to interpret our experimental results.

[3] In this study, we also consider models that repeat the deliberation multiple times. However, we did not observe significant performance improvement.

Sentiment Classification Layer

Finally, we pass the representation of each review for aspect k into an aspect-specific classifier to get the probability distribution over different class labels. Here, the classifier is defined as a two layer feed-forward network with a ReLU activation followed by a softmax layer, i.e.,

$$\begin{aligned} y^k_{\text{out}} &= \text{ReLU}(W^k_{\text{out}} s^k + b^k_{\text{out}}), \\ y^k_{\text{pred}} &= \text{softmax}(W^k_{\text{pred}} y^k_{\text{out}} + b^k_{\text{pred}}), \end{aligned} \tag{6.7}$$

where W^k_{out}, W^k_{pred}, b^k_{out}, and b^k_{pred} are learnable parameters.

Given the ground-truth labels $\hat{y}^k$, which is a one-hot vector, our goal is to minimize the averaged cross-entropy error between y^k_{pred} and $\hat{y}^k$ across all aspects, i.e.,

$$\mathcal{L}_\theta = -\sum_{k=1}^{K} \sum_{i=1}^{N} \hat{y}^k_i \log(y^k_{\text{pred},i}), \tag{6.8}$$

where K and N represents the number of aspects and class labels, respectively. The model is trained in an end-to-end manner using back-propagation.

6.2.2 Aspect and Sentiment Keywords

Traditionally, aspect and sentiment keywords are obtained using unsupervised clustering methods, such as topic models [94, 125]. However, these methods cannot automatically build correlations between keywords and aspects or sentiment due to the lack of supervision. Aspect and opinion term extractions in fine-grained aspect-based sentiment analysis tasks [110, 108, 28, 150] focus on extracting terms and phrases from sentences. However, they require a number of labeled reviews to train deep learning models. In this study, we propose a fully automatic Attention-driven Keywords Ranking (AKR) method to discover aspect and opinion keywords, which are important to predicted ratings, from a review corpus based on a self-attention (or deliberate self-attention) mechanism in the context of DMSC.

Aspect Keywords Ranking

The significance of a word w to an aspect k can be described by a conditional probability $p_{\mathcal{C}}(w|k)$ on a review corpus $\mathcal{C}$. Intuitively, given an aspect k, if a word w_1 is more frequent than w_2 across the corpus, then, w_1 is more significant to aspect k. We can further expand this probability as follows:

$$p_{\mathcal{C}}(w|k) = \sum_{\xi \in \mathcal{C}} p_{\mathcal{C}}(w, \xi|k), \tag{6.9}$$

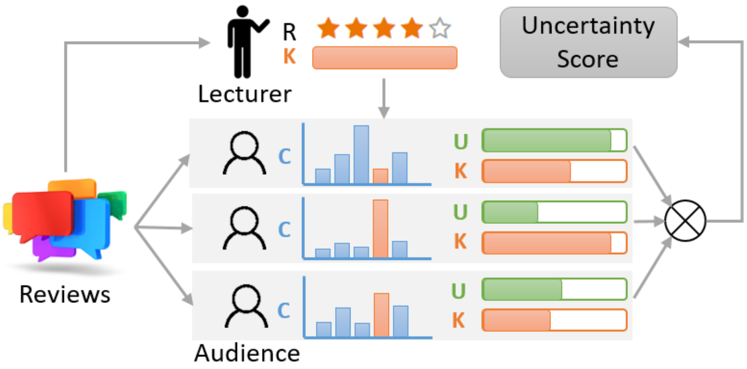


Figure 6.4: The LEcture-AuDience (LEAD) model for uncertainty estimation. 'R', 'K', 'C', 'U' represent rating, knowledge, probability distribution of different class labels (see Eq. (6.7)), and uncertainty score, respectively.

where ξ is a review in corpus $\mathcal{C}$. For each $\xi \in \mathcal{C}$, probability $p_{\mathcal{C}}(w, \xi|k)$ indicates the importance of word w to the aspect k, which can be defined using attention weights, i.e.,

$$p_{\mathcal{C}}(w, \xi|k) = \frac{\sum_{t=1}^{T} \alpha_t^{\xi} \cdot \delta(w_t, w)}{\sum_{\xi' \in \mathcal{C}} f_{\xi'}(w) + \gamma}, \tag{6.10}$$

where $f_{\xi'}(w)$ is frequency of w in document ξ' and γ is a smoothing factor. $\delta(w_t, w) = \begin{cases} 1 & \text{if } w_t = w \\ 0 & \text{otherwise} \end{cases}$ is a delta function. Attention weight α_t^{ξ} is defined as $\alpha_t^{\xi} = \frac{1}{2}(\alpha_{t,G}^{k} + \alpha_{t,D}^{k})$ for the deliberation self-attention mechanism. In Eq. (6.10), the denominator is applied to reduce the noise from stop-words and punctuation. After obtaining the score $p_{\mathcal{C}}(w|k)$ for every member in the vocabulary, we collect top-ranked words (with part-of-speech tags: NOUN and PROPN) as aspect keywords.

Aspect-level Opinion Keywords

Similarly, we can estimate the significance of a word w to an aspect-level opinion label/rating $\hat{y}^k$ by a conditional probability $p_{\mathcal{C}}(w|\hat{y}^k)$. Let us use $\mathcal{C}_{\hat{y}^k}$ to denote reviews with rating $\hat{y}^k$ for aspect k, then, the following equivalence holds, i.e.,

$$p_{\mathcal{C}}(w|\hat{y}^k) = p_{\mathcal{C}_{\hat{y}^k}}(w|k), \tag{6.11}$$

which can be further calculated by Eqs. (6.9) and (6.10). Intuitively, we first construct a subset $\mathcal{C}_{\hat{y}^k} \subset \mathcal{C}$ of the review corpus, then, we use attention weights of aspect k to calculate the significance of word w to that aspect. Finally, we collect top-ranked words (with part-of-speech tags: ADJ, ADV and VERB) as aspect-level opinion keywords.

6.2.3 The Proposed Uncertainty Model

Although our FEDAR model has achieved competitive prediction accuracy and our AKR method allows us to explore aspect and sentiment keywords, it is still difficult to deploy such a model in real-world applications. In DMSC datasets, we find that there are many typos and abbreviations in reviews and many reviews describe the product or service from only one aspect. However, deep learning models cannot capture these problems in the datasets, therefore, the predictions are not reliable. One way to tackle this challenge is by estimating the uncertainty of model predictions. If a model returns ratings with high uncertainty, we can pass the review to human experts for annotation. In this section, we propose a LEcture-AuDience (LEAD) method (see Fig.6.4) to measure the uncertainty of our FEDAR model in the context of multi-task learning.

Lecturer and Audiences

We use a lecturer (denoted by $\mathcal{M}^L$) to represent any well-trained deep learning model, e.g., FEDAR model. Audiences are models (denoted by $\mathcal{M}^A$) with partial knowledge of the lecturer, where *knowledge can be interpreted as relationships between an input review and output ratings* which are inferred by $\mathcal{M}^L$. Here, $\mathcal{M}^A = \{\mathcal{M}^{A_1}, \mathcal{M}^{A_2}, ..., \mathcal{M}^{A_{|A|}}\}$, where $|A|$ is the number of audiences. *Partial knowledge determines the eligibility of audiences to provide uncertainty scores.* For example, eligible audiences can be: (1) Models obtained by pruning some edges (e.g., dropout with small dropout rate) of the lecturer model. (2) Models obtained by continuing training of the lecturer model with very small learning rate for a few batches. Ineligible audiences include: (1) Random models trained on the same or a different review corpus. (2) Models with the same or similar structure as lecturer but initialized with different parameters and trained on a different corpus.

Uncertainty Scores

Given a review, suppose the lecturer $\mathcal{M}^L$ predicts the class label as $\tilde{y}^{L,k}$ for aspect k, where $\tilde{y}^{L,k}$ is an one-hot vector. An audience $\mathcal{M}^{A_\mu}$ obtains the probability distribution over different class labels as $y^{A_\mu,k}_{\text{pred}}$ (see Eq. (6.7)). Then, the uncertainty score is defined as the cross entropy between $\tilde{y}^{L,k}$ and $y^{A_\mu,k}_{\text{pred}}$, which is calculated by

$$\psi^{A_\mu,k} = -\sum_{i=1}^{N} \tilde{y}_i^{L,k} \log(y^{A_\mu,k}_{\text{pred},i}). \tag{6.12}$$

Intuitively, the audience is more uncertain about the lecturer's prediction if it gets lower probability for that prediction. For example, in Fig. 6.4, the lecturer model predicts rating/label as 4. Three audiences obtain probability 0.1, 0.8, 0.5 for that label, respectively. Then, their uncertainty scores are $\psi^{A_1,k} = 2.30$, $\psi^{A_2,k} = 0.22$, and $\psi^{A_3,k} = 0.69$.

With the uncertainty score from a single audience and for a single aspect, we can calculate the final uncertainty score as

$$\psi = \exp \sum_{\mu=1}^{|A|} \zeta \log \left(\exp \sum_{k=1}^{k} \log \left(\psi^{A_{\mu},k} + \lambda \right) + \eta \right), \tag{6.13}$$

where λ and $\eta \geq 1$ are smoothing factors that are set to 1 in our experiments. ζ is an empirical factor for knowledge. If audience networks are obtained by applying dropout to the lecturer network, the higher the dropout rate, the lower the factor ζ. In this case, the audiences have less knowledge of the lecturer.

After obtaining uncertainty scores for all reviews in the testing set, we can select either a certain percent of reviews with higher scores or reviews with scores over a threshold for crowdsourcing annotation. Human experts are expected to analyze the reviews and decide the aspect ratings for them.

6.3 Experiments

In this section, we present the results from an extensive set of experiments and demonstrate the effectiveness of our proposed FEDAR model, AKR, and LEAD methods.

6.3.1 Research Questions

Our empirical analysis aims at the following Research Questions (RQs):

- **RQ1**: What is the overall performance of FEDAR? Does it outperform state-of-the-art baselines?
- **RQ2**: What is the overall performance of the LEAD method compared with uncertainty estimation baselines?
- **RQ3**: How does each component in FEDAR contribute to the overall performance?
- **RQ4**: Is the deliberate self-attention module interpretable? Does it learn meaningful aspect and opinion terms from a review corpus?

6.3.2 Datasets

We first conduct our experiments on five benchmark datasets, which are obtained from the TripAdvisor and BeerAdvocate review platforms. TripAdvisor based datasets have seven aspects (*value, room, location, cleanliness, check in/front desk, service, and business service*), while BeerAdvocate based datasets have four aspects (*feel, look, smell, and taste*).

Table 6.1: Statistics of different DMSC datasets. † indicates the datasets collected and prepared by us.

Dataset	**# docs**	**# aspects**	**Scale**
TripAdvisor-R	29,391	7	1-5
TripAdvisor-RU	58,632	7	1-5
TripAdvisor-B	28,543	7	1-2
BeerAdvocate-R	50,000	4	1-10
BeerAdvocate-B	27,583	4	1-2
RateMDs-R†	155,995	4	1-5
RateMDs-B†	120,303	4	1-2

TripAdvisor-R [163], TripAdvisor-U [74], and BeerAdvocate-R [163, 69] use the original rating scores as sentiment class labels. In TripAdvisor-B and BeerAdvocate-B [167], the original scale is converted to a binary scale, where 1 and 2 correspond to negative and positive sentiment, respectively. Neutral has been ignored in both datasets. All datasets have been tokenized and split into train/development/test sets with a proportion of 8:1:1. In our experiments, we use the same datasets that are provided by the previous studies in the literature [163, 74, 167]. Statistics of the datasets are summarized in Table 6.1.

In addition to the aforementioned five datasets, we also propose two new datasets, i.e., RateMDs-R and RateMDs-B, and benchmarked our models on them. The RateMDs dataset was collected from the `https://www.ratemds.com` website which has textual reviews along with numeric ratings for medical experts primarily in the North America region. Each review comes with ratings of four different aspects, i.e., *staff, punctuality, helpfulness, and knowledge*. The overall rating is the average of these aspect ratings. To obtain a more refined dataset for our experiments, we removed reviews with missing aspect ratings and selected the rest of the reviews whose lengths are between 72 and 250 tokens (i.e., not outliers [4]), since short reviews may not have information on all the four aspects. The original data has a rating-imbalance problem, i.e., 60% and 17% of reviews are rated as 5 and 1, respectively, and more than 50% of reviews have identical aspect ratings. Therefore, similar to [69], we chose reviews with different aspect ratings, i.e., at least three of the aspect ratings are different. The statistics of our dataset have been shown in Table 6.1. For RateMDs-R, we tokenized reviews with Stanford CoreNLP[5] and randomly split the dataset into training, development and testing by a proportion of 135,995:10,000:10,000. For RateMDs-B, we followed the process in [167] by converting original scales to binary and sampling data according to the overall polarities to avoid the imbalance issue. The statistics of the RateMDs-B dataset have also been shown in Table 6.1. Similarly, we split the dataset into training, development and testing by a proportion of 100,303:10,000:10,000.

[4] The average number of tokens for all reviews is 72 tokens and there are very few reviews with more than 250 tokens.

[5] `https://stanfordnlp.github.io/CoreNLP/`

6.3.3 Comparison Methods

To demonstrate the effectiveness of our methods, we compare the proposed models with the following baseline methods:

- **MAJOR** simply uses the majority sentiment labels or polarities in training data as predictions.
- **GLVL** first calculates the document representation by averaging the word vectors of all keywords in a review, where pre-trained word vectors are obtained from GloVe [106]. Then, a LIBLINEAR package [27] is used for the classification task.
- **BOWL** feeds the normalized Bag-of-Words (BOW) representation of reviews into the LIBLINEAR package for the sentiment classification. In our experiments, stop-words and punctuation are removed in order to enable the model to capture the keywords more efficiently.
- **MCNN** is an extension of the CNN model in the multi-task learning framework. For each task, CNN [61] extracts key features from a review by applying convolution and max-over-time pooling [23] operations over the shared word embeddings layer.
- **MLSTM** extends a multi-layer Bi-LSTM model [47], which captures both forward and backward semantic information, with the multi-task learning framework, where different tasks have their own classifiers and share the same Bi-LSTM encoder.
- **MBERT** is a multi-task version of the BERT classification model [26]. Different tasks share the same BERT encoder [153].
- **MATTN** is a multi-task version of self-attention based models. Similar to MLSTM, different tasks share the same Bi-LSTM encoder. For each task, we first apply a self-attention layer, and then pass the document representations to a sentiment classifier.
- **DMSCMC** [163] introduces a hierarchical iterative attention model to build aspect-specific document representations by frequent and repeated interactions between documents and aspect questions.
- **HRAN** [74] incorporates hand-crafted aspect keywords and the overall rating into a hierarchical network to build sentence and document representations.
- **AMN** [169] first uses attention-based memory networks to incorporate hand-crafted aspect keywords information into the aspect and sentence memories. Then, recurrent attention operation and multi-hop attention memory networks are employed to build document representations.
- **FEDAR** is the name of our model, where FE, DA and R represent Feature Enrichment, Deliberate self-Attention, and overall Rating, respectively.

We compare our LEAD method with the following uncertainty estimation approaches:

- **Max-Margin** is the maximal activation of the sentiment classification layer (after softmax normalization).
- **PL-Variance** (Penultimate Layer Variance) [166] uses the variance of the output of the

sentiment classification layer (before softmax normalization) as the uncertainty score.

- **Dropout** [32] applies dropout to deep neural networks during training and testing. The dropout can be used as an approximation of Bayesian inference in deep Gaussian processes, which aims to identify low-confidence regions of input space.

All methods are based on our FEDAR model.

6.3.4 Implementation Details

We implemented all deep learning models using PyTorch [105] and the best set of parameters are selected based on the development set. Word embeddings are pre-loaded with 300-dimensional GloVe embeddings [106] and fixed during training. For MCNN, filter sizes are chosen to be 3, 4, 5 and the number of filters are 400 for each size. For all LSTM based models, the dimension of hidden states is set to 600 and the number of layers is 4. All parameters are trained using the ADAM optimizer [63] with an initial learning rate of 0.0005. The learning rate decays by 0.8 every 2 epochs. Dropout with a dropout-rate 0.2 is applied to the classifiers. Gradient clipping with a threshold of 2 is also applied to prevent gradient explosion. For MBERT, we leveraged the pre-trained BERT encoder from HuggingFace's Transformers package [153] and fixed its weights during training. We also adopted the learning rate warmup heuristic [79] and set the warmup step to 2000. For dropout-based uncertainty estimation methods, we set the dropout-rate to 0.5. The number of samples for Dropout are 50. The number of audiences is 20 for our LEAD model. ζ is set to 1.0. Our codes and datasets are available at `https://github.com/tshi04/DMSC_FEDA`.

6.3.5 Prediction Performance

For research question **RQ1**, we use accuracy (ACC) and mean squared error (MSE) as our evaluation metrics to measure the prediction performance of different models. All results are shown in Tables 6.2 and 6.3, where we use bold font to highlight the best performance values and underlining to highlight the second best values.

For the DMSC problem, it has been demonstrated that deep neural network (DNN) based models perform much better than conventional machine learning methods that rely on n-gram or embedding features [163, 74]. In our experiments, we have also demonstrated this by comparing different DNN models with MAJOR, GLVL, and BOWL. Compared to simple DNN classification models, multi-task learning DNN models (MDNN) can achieve better results with fewer parameters and training time [163]. Therefore, we focused on comparing the performance of our model with different MDNN models. From Table 6.2, DMSCMC achieves better results on all five datasets compared with baselines MCNN, MLSTM, MBERT, and MATTN. HRAN and AMN leverage the power of overall rating and get significantly better results than other compared methods. From both tables, we observed our FEDAR

Table 6.2: Averaged Accuracy (ACC) and MSE of different models on TripAdvisor-R (Trip-R), TripAdvisor-U (Trip-U), TripAdvisor-B (Trip-B), BeerAdvocate-R (Beer-R), and BeerAdvocate-B (Beer-B) testing sets. For MSE, smaller is better. † indicates that results are obtained from previous published papers and NA indicates that results are not available in those papers. We use bold font to highlight the best performance values and underlining to highlight the second best values.

Method	**Trip-R**		**Trip-U**		**Trip-B**	**Beer-R**		**Beer-B**
	ACC	**MSE**	**ACC**	**MSE**	**ACC**	**ACC**	**MSE**	**ACC**
MAJOR	29.12	2.115	39.73	1.222	62.42	26.29	4.252	67.26
GLVL	38.94	1.795	48.04	0.879	78.15	30.59	2.774	79.73
BOWL	40.14	1.708	48.68	0.888	78.38	31.02	2.715	79.14
MCNN	41.75	1.458	51.21	0.714	81.31	34.11	2.016	82.37
MLSTM	42.74	1.401	48.64	0.791	80.56	34.48	2.167	82.07
MATTN	42.13	1.427	50.53	0.679	80.82	35.78	1.962	84.86
MBERT	44.41	1.250	54.50	0.617	82.84	35.94	1.963	84.73
DMSCMC†	46.56	1.083	55.49	0.583	83.34	38.06	1.755	86.35
HRAN†	47.43	1.169	58.15	0.528	NA	39.11	1.700	NA
AMN†	48.66	1.109	NA	NA	NA	40.19	1.686	NA
FEDAR (Ours)	**48.92**	**1.072**	**58.50**	**0.522**	**85.50**	**40.62**	**1.530**	**87.40**

Table 6.3: Averaged accuracy (ACC) and MSE of different models on RateMDs-R (RMD-R) and RateMDs-B (RMD-B) testing sets. For MSE, smaller is better.

Method	**RMD-R**		**RMD-B**
	ACC	**MSE**	**ACC**
MAJOR	31.42	3.393	57.18
GLVL	43.11	1.882	76.93
BOWL	44.78	1.704	78.68
MCNN	46.19	1.333	81.60
MLSTM	48.37	1.148	82.40
MATTN	49.08	1.157	82.66
MBERT	48.65	1.160	83.39
FEDAR (Ours)	**55.57**	**0.794**	**88.63**

model achieves the best performance on all seven datasets. These results demonstrate the effectiveness of our methods.

6.3.6 Uncertainty Performance

Uncertainty estimation can help users identify reviews for which the models are not confident of their predictions. More intuitively, prediction models are prone to mistakes on the reviews

Table 6.4: Performance of various uncertainty methods on different datasets.

TripAdvisor-R

Method	**top-5%**	**top-10%**	**top-15%**	**top-20%**	**top-25%**
Max-Margin	35.40	36.00	37.47	39.15	40.68
PL-Variance	40.20	42.40	43.00	44.25	44.84
Dropout	53.80	53.50	53.33	53.35	53.60
LEAD	**65.40**	**62.60**	**60.93**	**60.85**	**60.20**

BeerAdvocate-R

Method	**top-5%**	**top-10%**	**top-15%**	**top-20%**	**top-25%**
Max-Margin	38.80	43.80	46.53	48.30	49.44
PL-Variance	44.00	47.00	48.33	49.65	50.68
Dropout	57.00	57.90	58.60	58.70	59.28
LEAD	**71.60**	**69.50**	**67.93**	**67.55**	**67.20**

RateMDs-R

Method	**top-5%**	**top-10%**	**top-15%**	**top-20%**	**top-25%**
Max-Margin	20.20	23.80	26.80	28.15	29.40
PL-Variance	28.60	29.70	30.53	30.85	31.60
Dropout	51.00	50.70	50.60	49.60	48.88
LEAD	**66.00**	**62.70**	**60.40**	**59.05**	**58.32**

that they are uncertain about. In Table 6.4, we first selected the most uncertain predictions (denoted by **top-n%**) based uncertainty scores from the testing sets of the TripAdvisor-R, BeerAdvocate-R and RateMDs-R datasets. Then, we evaluated the uncertainty performance by comparing the mis-classification rate (i.e., error rate) of our FEDAR model for the selected reviews. The more incorrect predictions that can be captured, the better the uncertainty method will be. From these results, we can observe that the Dropout method achieves significantly better results than Max-Margin and PL-Variance. Our LEAD method outperforms all these baseline methods on three datasets, which shows our method is superior in identifying less confident predictions and answers research question **RQ2**.

6.3.7 Ablation Study of FEDAR

For research question **RQ3**, we attribute the performance improvement of our FEDAR model to: 1) Better review encoder, including a highway word embedding layer and a feature enriched encoder. 2) Deliberate self-attention mechanism. 3) Overall rating.

Therefore, we systematically conducted ablation studies to demonstrate the effectiveness of these components, and provided the results in Table 6.5, Table 6.6 and Fig. 6.5. We first observe that FEDAR significantly outperforms model-OR (FEDAR w/o OR), which indicates that overall rating can help the model make better predictions. Secondly, we compare model-OR with model-ORFE (FEDAR w/o OR, FE), which is equipped with a regular

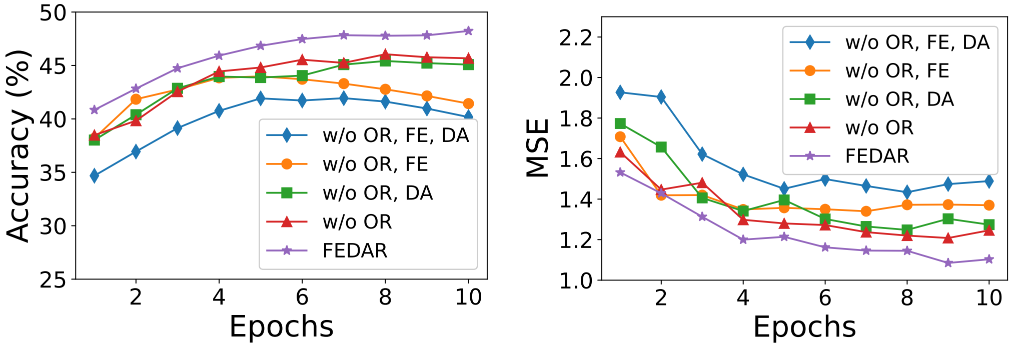
Table 6.5: Ablation study results. Different models are evaluated by Averaged Accuracy (ACC) and MSE metrics on five public DMSC testing sets. For MSE, smaller is better. **FE**, **DA** and **OR** represent Feature Enrichment, Deliberated self-Attention, Overall Rating, respectively.

Method	**Trip-R**		**Trip-U**		**Trip-B**	**Beer-R**		**Beer-B**
	ACC	**MSE**	**ACC**	**MSE**	**ACC**	**ACC**	**MSE**	**ACC**
FEDAR	**48.92**	**1.072**	**58.50**	**0.522**	**85.50**	**40.62**	**1.530**	**87.40**
w/o OR	46.72	1.178	55.82	0.574	84.23	39.66	1.617	86.52
w/o OR, DA	45.70	1.224	55.39	0.584	83.43	38.85	1.633	85.99
w/o OR, FE	44.50	1.300	53.41	0.632	82.39	38.92	1.714	84.99
w/o OR, DA, FE	42.13	1.427	50.53	0.679	80.82	35.78	1.962	84.86

Table 6.6: Ablation study results. Different models are evaluated by Averaged Accuracy (ACC) and MSE metrics on RateMDs-R (RMD-R) and RateMDs-B (RMD-B) testing sets.

Method	**RMD-R**		**RMD-B**
	ACC	**MSE**	**ACC**
FEDAR	**55.82**	**0.786**	**88.63**
w/o OR	49.80	1.106	83.89
w/o OR, DA	49.68	1.108	83.62
w/o OR, FE	49.28	1.123	83.47
w/o OR, DA, FE	49.08	1.157	82.66

word embedding layer and a multi-layer Bi-LSTM encoder. Obviously, model-OR obtained better results than model-ORFE. Similarly, we also compare model-ORDA (FEDAR w/o OR, DA) with model-BASE (FEDAR w/o OR, DA, FE), since model-ORDA adopts the same self-attention mechanism as model-BASE. It can be observed that model-ORDA performs significantly better than model-BASE on all the datasets. This experiment shows that we can improve the performance by using the highway word embedding layer and feature enrichment technique. Furthermore, we compared model-OR with model-ORDA, which does not have a deliberate self-attention layer. It can be seen that model-OR outperforms model-ORDA in all the experiments. In addition, we have also compared the results of model-ORFE and model-BASE, which are equipped with a deliberate self-attention layer and a regular self-attention layer. We observed that model-ORFE has a better performance compared to model-BASE. This experiment indicates the effectiveness of the deliberate self-attention mechanism. In Fig. 6.5, we show the accuracy and MSE of different models during training in order to demonstrate that FEDAR can get consistently higher accuracy and lower MSE after training for several epochs than its basic variants.



(a) Accuracy vs. Epochs

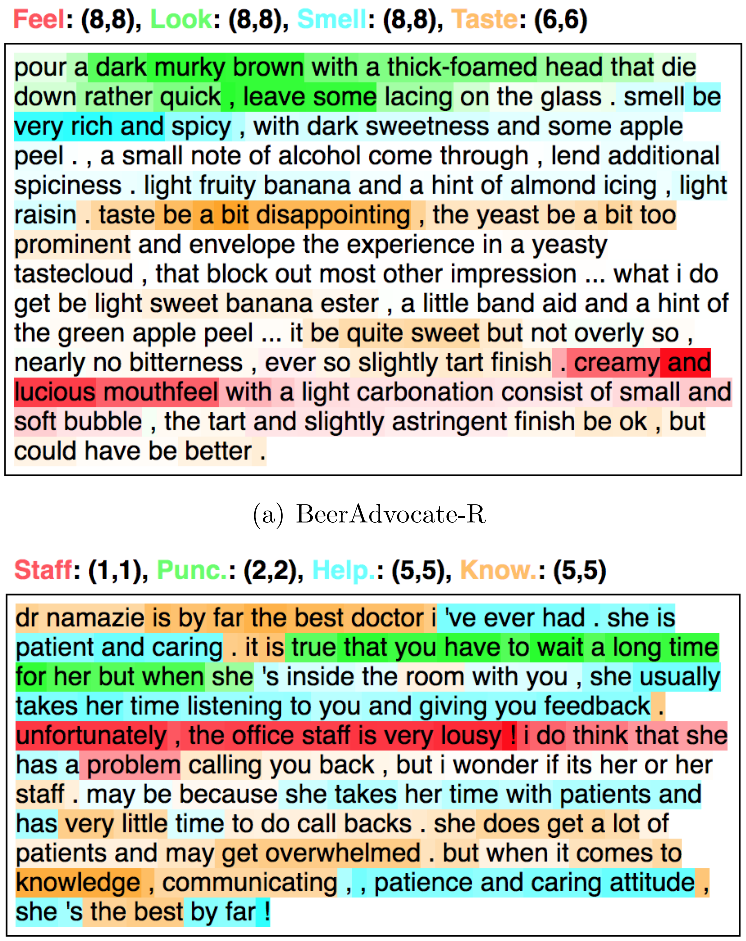
(b) MSE vs. Epochs

Figure 6.5: This figure shows (a) Averaged Accuracy and (b) MSE for FEDAR and its variants on the TripAdvisor-R dataset during the training process.

6.3.8 Attention Visualization

The attention mechanism enables a model to selectively focus on important parts of the reviews, and hence, visualization of the attention weights can help in interpreting our model and analyzing the experimental results [163, 155]. To answer research question **RQ4**, we need to investigate whether our model attends to relevant keywords when it is making aspect-specific rating predictions for the DMSC problem.

In Fig. 6.6 (a), we show a review example from the BeerAdvocate-R testing set, for which our model has successfully predicted all aspect-specific ratings. In this figure, we highlighted the review with deliberate attention weights. The review contains keywords of all four aspects, thus, we only need to verify whether our model can successfully detect those aspect-specific keywords. We observed that deliberate self-attention attends to "*creamy and luscious mouthfeel*" for **feel**. For the **look** aspect, it captures "*dark murky brown with a ..., leave some lacing on the glass*", which is quite relevant to the appearance of the beer. Our model also successfully detects "*very rich and spicy*" for **smell**. For **taste**, it attends to "*taste is a bit disappointing, ... too prominent*", which yields a slightly lower rating. Similarly, we show an example from the RateMDs-R testing set in Fig. 6.6 (b). Our model detects "*unfortunately, the office staff is very lousy! I do think ...*" for **staff**, which expresses negative opinion on the office staff. For **punctuality**, it captures "*true that you have to wait a long time for her*", which is also negative. Finally, it attends to "*is by far the best doctor, she does get a lot of patient and may get overwhelmed. but when it comes to knowledge, communicating, the best*" for the **knowledge**, and "*she is patient and caring, patience and caring attitude*" for the **helpfulness** of the doctor. Both aspects have positive sentiment. Therefore, these two examples show good interpretability of our model.



Feel: (8,8), Look: (8,8), Smell: (8,8), Taste: (6,6)

pour a dark murky brown with a thick-foamed head that die down rather quick , leave some lacing on the glass . smell be very rich and spicy , with dark sweetness and some apple peel . , a small note of alcohol come through , lend additional spiciness . light fruity banana and a hint of almond icing , light raisin . taste be a bit disappointing , the yeast be a bit too prominent and envelope the experience in a yeasty tastecloud , that block out most other impression ... what i do get be light sweet banana ester , a little band aid and a hint of the green apple peel ... it be quite sweet but not overly so , nearly no bitterness , ever so slightly tart finish . creamy and lucious mouthfeel with a light carbonation consist of small and soft bubble , the tart and slightly astringent finish be ok , but could have be better .

(a) BeerAdvocate-R

Staff: (1,1), Punc.: (2,2), Help.: (5,5), Know.: (5,5)

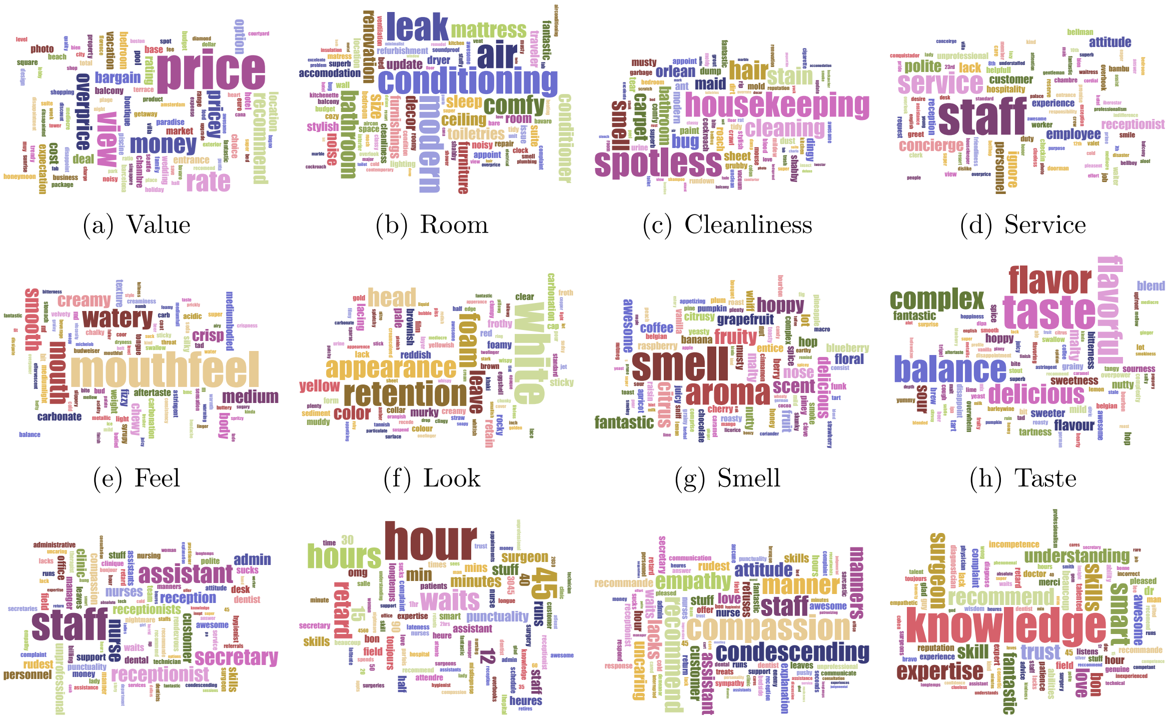
dr namazie is by far the best doctor i 've ever had . she is patient and caring . it is true that you have to wait a long time for her but when she 's inside the room with you , she usually takes her time listening to you and giving you feedback . unfortunately , the office staff is very lousy ! i do think that she has a problem calling you back , but i wonder if its her or her staff . may be because she takes her time with patients and has very little time to do call backs . she does get a lot of patients and may get overwhelmed . but when it comes to knowledge , communicating , , patience and caring attitude , she 's the best by far !

(b) RateMDs-R

Figure 6.6: Visualization of attention weights. In parentheses, the first and second numbers represent ground-truth and predicted ratings, respectively. Different aspects are labeled with different colors. The figure is best viewed in color.

6.3.9 Aspect and Opinion Keywords

In Fig. 6.7, we first show aspect keywords detected by our AKR method for the TripAdvisor-B, BeerAdvocate-B, and RateMDs-B corpora. From Fig. 6.7 (top row), we observe that **value** related keywords include "*price, money, rate, overprice*". Keywords related to a **room** are "*air conditioning, comfy, leak, mattress, bathroom, modern, ceiling*" and others. For **cleanliness**, people are interested in "*housekeeping, spotless, cleaning, hair, stain, smell*" and so on. **Service** is related with "*staff, service, employee, receptionist, personnel*". From Fig. 6.7 (middle row), we observe that **feel** is usually related with keywords, like "*mouthfeel, mouth, smooth, watery*", which describe feel of beers in mouth. **Look** is the appearance of beers, thus, the model captures "*appearance, retention, white, head, foam, color*" and



(a) Value (b) Room (c) Cleanliness (d) Service

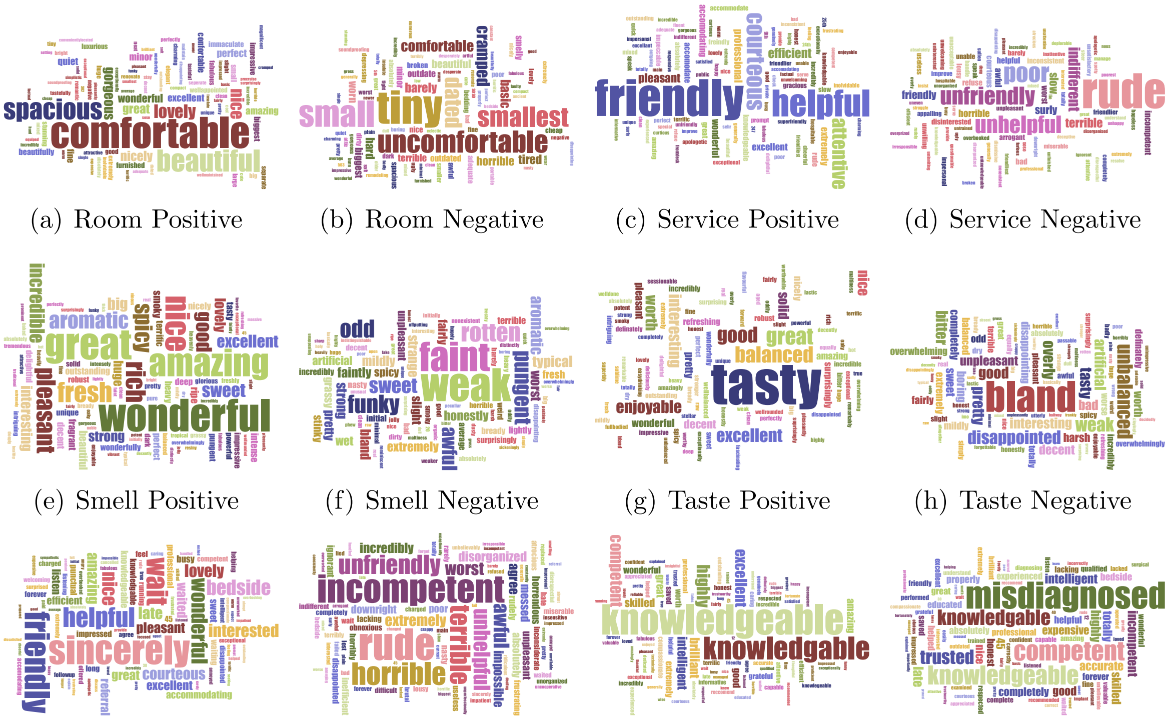
(e) Feel (f) Look (g) Smell (h) Taste

(i) Staff (j) Punctuality (k) Helpfulness (l) Knowledge

Figure 6.7: Word-cloud visualization of aspect keywords for TripAdvisor-B (Top row), BeerAdvocate-B (middle row) and RateMDs-B datasets (bottom row).

others. **Smell** related aspect keywords include "*smell, aroma, scent, fruity*" and more. Finally, representative keywords for **taste** are "*taste, balance, complex, flavor*" and so on. From Fig. 6.7 (bottom row), we observe that **staff** related keywords are "*staff, assistant, secretary, receptionist*" and so on. For **punctuality**, people usually concern "*waits, hour, hours, retard*". The **helpfulness** of a doctor is related to "*compassion, manner, empathy, attitude, condescending*" and so on. Finally, **knowledge** related keywords are "*knowledge, expertise, surgeon, skill*" and others.

We also obtain aspect-specific opinion keywords from the Trip-B, Beer-B, and RMD-B datasets, and show them in Fig. 6.8. From this figure (top row), we observe that reviewers with positive experience usually live in "*comfortable, beautiful, spacious, lovely and gorgeous*" rooms, and the staff are "*helpful, friendly, courteous and attentive*", while reviewers with negative experience may live in "*uncomfortable, small, cramped and tiny*" rooms. Something may "*leak*" and there are also problems with "*air conditioning*". The staff are "*rude, unhelpful and unfriendly*" and the service is "*poor*". From Fig. 6.8 (middle row), we learn that good beers should have "*great, amazing, wonderful, pleasant, aromatic, fresh, rich, and incredible*" smell, and the taste may be "*tasty, great, balanced, enjoyable, and flavorful*". The smell of low-rated beers is "*faint, weak, pungent, odd, funky, and rotten*", and the taste may be "*bland, unbalanced, disappointed, and sour*". From Fig. 6.8 (bottom row), we find that good



(a) Room Positive (b) Room Negative (c) Service Positive (d) Service Negative

(e) Smell Positive (f) Smell Negative (g) Taste Positive (h) Taste Negative

(i) Staff Positive (j) Staff Negative (k) Knowledge Positive (l) Knowledge Negative

Figure 6.8: Word-cloud visualization of aspect-level opinion keywords for TripAdvisor-B (top row), BeerAdvocate-B (middle row), and RateMDs-B datasets (bottom row).

doctors usually have "*sincerely, friendly, helpful, and wonderful*" staff and are "*knowledgeable, competent, intelligent, and excellent*". In a low-rated clinic, staff may be "*incompetent, rude, horrible, terrible, and unfriendly*", and doctors may "*misdiagnose*" conditions of patients and can be not "*competent, knowledgeable, or trusted*".

From these figures, we can conclude that our deliberate self-attention mechanism is interpretable, and by leveraging our AKR method, it is a powerful knowledge discovery tool for online multi-aspect reviews, which answers research question **RQ4**.

6.4 Summary

In this study, we proposed a multi-task deep learning model, namely FEDAR, for the problem of document-level multi-aspect sentiment classification. Different from previous studies, our model does not require hand-crafted aspect-specific keywords to guide the attention and boost model performance for the task of sentiment classification. Instead, our model relies on (a) a highway word embedding layer to transfer knowledge from pre-trained word vectors on a large corpus, (b) a sequential encoder layer whose output features are enriched by pooling and feature factorization techniques, and (c) a deliberate self-attention layer which maintains the interpretability of our model. Experiments on various DMSC datasets have demonstrated the superior performance of our model. In addition, we also developed an Attention-driven Keywords Ranking (AKR) method, which can automatically discover aspect and opinion keywords from the review corpus based on attention weights. Attention weights visualization and aspect/opinion keywords word-cloud visualization results have demonstrated the interpretability of our model and effectiveness of our AKR method. Finally, we also proposed a LEcture-AuDience (LEAD) method to measure the uncertainty of deep neural networks, including our FEDAR model, in the context of multi-task learning. Our experimental results on multiple real-world datasets demonstrate the effectiveness of the proposed work.

Chapter 7

Self-Supervised Contrastive Learning for Aspect Detection

This chapter introduces a self-supervised contrastive learning framework and an attention-based model equipped with a novel smooth self-attention module for the unsupervised aspect detection task. We also introduce a high-resolution selective mapping method to efficiently assign aspects discovered by the model to the aspects of interest. In addition, we propose using a knowledge distillation technique to further improve the aspect detection performance. The rest of this chapter is organized as follows: The introduction of this chapter is presented in Section 7.1. In Section 7.2, we present details of our self-supervised contrastive learning framework, high-resolution select mapping method and knowledge distillation approach. In Section 7.3, we introduce different aspect detection datasets, baseline methods and implementation details, as well as analyze experimental results. Our discussion concludes in Section 7.4.

7.1 Background and Motivation

Aspect detection, which is a vital component of aspect-based sentiment analysis [110, 109], aims at identifying predefined aspect categories (e.g., *Price*, *Quality*) discussed in segments (e.g., sentences) of online reviews. Table 7.1 shows an example review about a television from several different aspects, such as *Image*, *Sound*, and *Ease of Use*. With a large number of reviews, automatic aspect detection allows people to efficiently retrieve review segments of aspects they are interested in. It also benefits many downstream tasks, such as review summarization [1] and recommendation justification [102].

There are several research directions for aspect detection. *Supervised approaches* [168] can leverage annotated labels of aspect categories but suffer from domain adaptation problems [118]. Another research direction consists of *unsupervised approaches* and has gained a lot

Table 7.1: An example from Amazon product reviews about a television and aspect annotations for every sentence.

Sentence	**Aspect**
Replaced my 27” jvc clunker with this one.	General
It fits perfectly inside our armoire.	General
Good picture.	Image
Easy to set up and program.	Ease of Use
Descent sound, not great...	Sound
We have the 42” version of this set downstairs.	General
Also a solid set.	General

of attention in recent years. Early unsupervised systems are dominated by Latent Dirichlet Allocation (LDA) based topic models [10, 100, 34, 113, 171]. However, several recent studies have revealed that LDA-based approaches do not perform well for aspect detection and the extracted aspects are of poor quality (incoherent and noisy) [43]. Compared to LDA-based approaches, deep learning models, such as aspect-based autoencoder (ABAE) [43, 87], have shown excellent performance in extracting coherent aspects and identifying aspect categories for review segments. However, these models require some human effort to manually map model discovered aspects to aspects of interest, which may lead to inaccuracies in mapping especially when model discovered aspects are noisy. Another research direction is based on *weakly supervised approaches* that leverage a small number of aspect representative words (namely, *seed words*) for the fine-grained aspect detection [1, 55]. Although these models outperform unsupervised approaches, they do make use of human annotated data to extract high-quality aspect seed words, which may limit their application. In addition, they are not able to automatically discover new aspects from review corpora.

We focus on the problem of unsupervised aspect detection (UAD) since a large number of reviews are generated every day and many of them are for newer products. It is difficult for humans to efficiently capture new aspects and manually annotate segments for them at scale. Motivated by ABAE, we learn interpretable aspects by mapping aspect embeddings into word embedding space, so that aspects can be interpreted by the nearest words. To learn better representations for both aspects and review segments, we formulate UAD as a self-supervised representation learning problem and solve it using a contrastive learning algorithm, which is inspired by the success of self-supervised contrastive learning in visual representations [14, 42]. In addition to the learning algorithm, we also resolve two problems that deteriorate the performance of ABAE, including its self-attention mechanism for segment representations and aspect mapping strategy (i.e., many-to-one mapping from aspects discovered by the model to aspects of interest). Finally, we discover that the quality of aspect detection can be further improved by knowledge distillation [46]. The contributions of this study are summarized as follows:

- Propose a self-supervised contrastive learning framework for the unsupervised aspect detection task.
- Introduce a high-resolution selective mapping strategy to map model discovered aspects to the aspects of interest.
- Utilize knowledge distillation to further improve the performance of aspect detection.
- Conduct systematic experiments on seven benchmark datasets and demonstrate the effectiveness of our models both quantitatively and qualitatively.

7.2 Proposed Methods

In this section, we describe our self-supervised contrastive learning framework for aspect detection shown in Fig. 7.1. The goal is to first learn a set of interpretable aspects (named as *model-inferred aspects*), and then extract aspect-specific segments from reviews so that they can be used in downstream tasks.

Problem Statement

The *aspect detection problem* is defined as follows: given a review segment $x = \{x_1, x_2, ..., x_T\}$ such as a sentence or an elementary discourse unit (EDU) [91], the goal is to predict an aspect category $y_k \in \{y_1, y_2, ..., y_K\}$, where x_t is the index of a word in the vocabulary, T is the total length of the segment, y_k is an aspect among all aspects that are of interest (named as *gold-standard aspects*), and K is the total number of gold-standard aspects. For instance, when reviewing restaurants, we may be interested in the following gold-standard aspects: *Food*, *Service*, *Ambience*, etc. Given a review segment, it most likely relates to one of the above aspects.

The first challenge in this problem is to learn model-inferred aspects from unlabeled review segments and map them to a set of gold-standard aspects. Another challenge is to accurately assign each segment in a review to an appropriate gold-standard aspect y_k. For example, in restaurants reviews, "*The food is very good, but not outstanding.*"→*Food.* Therefore, we propose a series of modules in our framework, including segment representations, contrastive learning, aspect interpretation and mapping, and knowledge distillation, to overcome both challenges and achieve our goal.

7.2.1 Self-Supervised Contrastive Learning

To automatically extract interpretable aspects from a review corpus, a widely used strategy is to learn aspect embeddings in the word embedding space so that the aspects can be interpreted using their nearest words [43, 1]. Here, we formulate this learning process as a *self-supervised representation learning* problem.

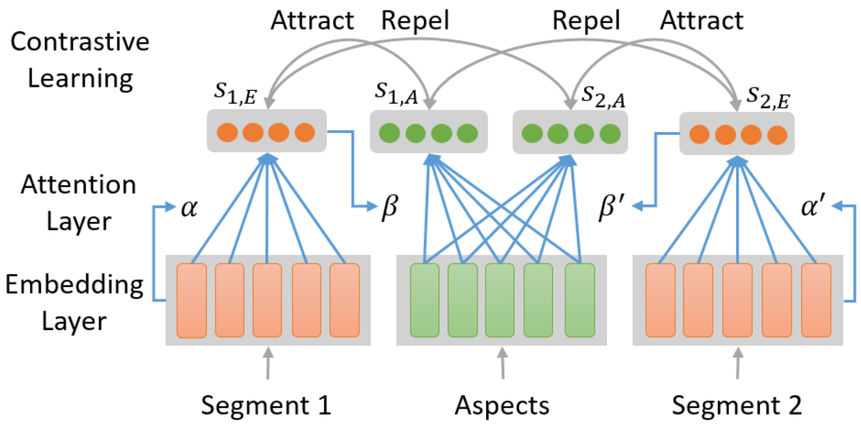


Figure 7.1: The proposed self-supervised contrastive learning framework. Attract and Repel represent positive and negative pairs, respectively.

Segment Representations

For every review segment in a corpus, we construct two representations directly based on (i) word embeddings and (ii) aspect embeddings. Then, we develop a *contrastive learning mechanism* to map aspect embeddings to the word embedding space. Let us denote a word embedding matrix as $E \in \mathbb{R}^{V \times M}$, where V is the vocabulary size and M is the dimension of word vectors. The aspect embedding matrix is represented by $A \in \mathbb{R}^{N \times M}$, where N is the number of model-inferred aspects.

Given a review segment $x = \{x_1, x_2, ..., x_T\}$, we construct a vector representation $s_{x,E}$ based on its word embeddings $\{E_{x_1}, E_{x_2}, ..., E_{x_T}\}$, along with a novel self-attention mechanism, i.e.,

$$s_{x,E} = \sum_{t=1}^{T} \alpha_t E_{x_t}, \tag{7.1}$$

where α_t is an attention weight and is calculated as follows:

$$\begin{aligned} \alpha_t &= \frac{\exp(u_t)}{\sum_{\tau=1}^{T} \exp(u_\tau)} \\ u_t &= \lambda \cdot \tanh\left(q^\top (W_E E_{x_t} + b_E)\right) \end{aligned} \tag{7.2}$$

Here, u_t is an alignment score and $q = \frac{1}{T}\sum_{t=1}^{T} E_{x_t}$ is a query vector. $W_E \in \mathbb{R}^{M \times M}$, $b_E \in \mathbb{R}^M$ are trainable parameters, and the smooth factor λ is a hyperparameter. More specifically, we call this attention mechanism as **Smooth Self-Attention (SSA)**. It applies an activation function tanh to prevent the model from using a single word to represent the segment, thus increasing the robustness of our model. For example, for the segment "*plenty of ports and settings*", SSA will attend on both "*ports*" and "*settings*", while regular self-attention may

Algorithm 2: The SSCL Algorithm

```
Input: Batch size $X$; constants $\lambda$ and $\tau$; network structures;
Output: Aspect embedding matrix $A$; model parameters $W_E$, $b_E$, $v_A$, $b_A$;
Initialize Matrix E with pre-trained word vectors; matrix A with k-means centroids;
for sampled mini-batch of size $X$ do
    for i=1,X do
        Calculate $s_{i,E}$ with Eq. (7.1);
        Calculate $s_{i,A}$ with Eq. (7.3);
    end
    for i=1,X; j=1,X do
        Calculate sim$(s_{j,E}, s_{i,A})$ with Eq. (7.6);
    end
    for i=1,X do
        Calculate $l_i$ with Eq. (7.5);
    end
    Calculate regularization term $\Omega$ using Eq. (7.7);
    Define Loss function $\mathcal{L} = \frac{1}{X}\sum_{i=1}^{X} l_i + \Omega$;
    Update learnable parameters to minimize $\mathcal{L}$.
end
```

only concentrate on "*settings*". Hereafter, we will use **RSA** to represent regular self-attention adopted in [1]. In our experiments, we discover that RSA without smoothness gets worse performance compared to a simple average pooling mechanism.

Further, we also construct a vector representation $s_{x,A}$ for the segment x with global aspect embeddings $\{A_1, A_2, ..., A_N\}$ through another attention mechanism, i.e.,

$$s_{x,A} = \sum_{n=1}^{N} \beta_n A_n \tag{7.3}$$

The attention weight β_n is obtained by

$$\beta_n = \frac{\exp\left(v_{n,A}^{\top} s_{x,E} + b_{n,A}\right)}{\sum_{\eta=1}^{N} \exp\left(v_{\eta,A}^{\top} s_{x,E} + b_{\eta,A}\right)}, \tag{7.4}$$

where $v_{n,A} \in \mathbb{R}^M$ and $b_{n,A} \in \mathbb{R}$ are learnable parameters. $\beta = \{\beta_1, \beta_2, ..., \beta_N\}$ can be also interpreted as **soft-labels (probability distribution) over model-inferred aspects** for a review segment.

Contrastive Learning

Inspired by recent contrastive learning algorithms [14], SSCL learns aspect embeddings by introducing a contrastive loss to maximize the agreement between two representations of the same review segment. During training, we randomly sample a mini-batch of X examples and define the contrastive prediction task on pairs of segment representations from the mini-batch, which is denoted by $\{(s_{1,E}, s_{1,A}), (s_{2,E}, s_{2,A}), ...(s_{X,E}, s_{X,A})\}$. Similar to [15], we treat $(s_{i,E}, s_{i,A})$ as a positive pair and $\{(s_{j,E}, s_{i,A})\}_{j \neq i}$ as negative pairs within the mini-batch. The contrastive loss function for a positive pair of examples is defined as

$$l_i = -\log \frac{\exp\left(\text{sim}(s_{i,E}, s_{i,A})/\mu\right)}{\sum_{j=1}^{X} \mathbb{I}_{[j \neq i]} \exp\left(\text{sim}(s_{j,E}, s_{i,A})/\mu\right)}, \tag{7.5}$$

where $\mathbb{I}_{[j \neq i]} \in \{0, 1\}$ is an indicator function that equals 1 iff $j \neq i$ and μ represents a temperature hyperparameter. We utilize cosine similarity to measure the similarity between $s_{j,E}$ and $s_{i,A}$, which is calculated as follows:

$$\text{sim}(s_{j,E}, s_{i,A}) = \frac{(s_{j,E})^\top s_{i,A}}{\|s_{j,E}\| \|s_{i,A}\|}, \tag{7.6}$$

where $\|\cdot\|$ denotes L_2-norm.

We summarize our SSCL framework in Algorithm 2. Specifically, in line 1, the aspect embedding matrix A is initialized with the centroids of clusters by running k-means on the word embeddings. We follow [43] to penalize the aspect embedding matrix and ensure diversity of different aspects. In line 13, the regularization term Ω is defined as

$$\Omega = \|\mathcal{A}\mathcal{A}^\top - I\|, \tag{7.7}$$

where each row of matrix $\mathcal{A}$, denoted by $\mathcal{A}_j$, is obtained by normalizing the corresponding row in A, i.e., $\mathcal{A}_j = A_j / \|A_j\|$.

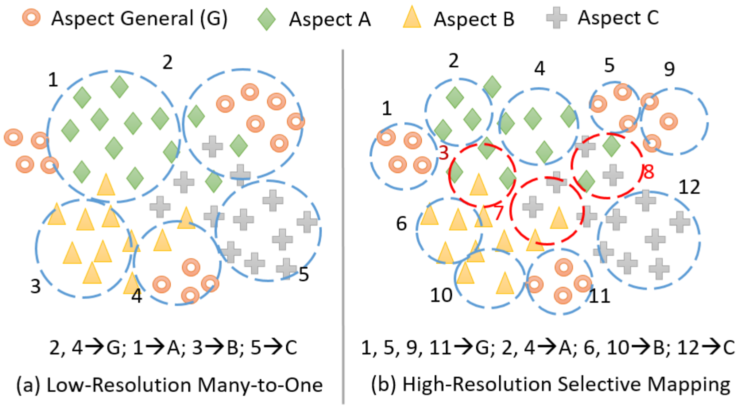
7.2.2 Aspect Interpretation and Mapping

Aspect Interpretation

In the training stage, we map aspect embeddings to the word embedding space in order to extract interpretable aspects. With embedding matrices A and E, we first calculate a similarity matrix

$$G = AE^\top,$$

where $G \in \mathbb{R}^{N \times V}$. Then, we use the top-ranked words based on G_n to represent and interpret each model-inferred aspect n. In our experiments, the matrix with inner product similarity produces more meaningful representative words compared to using the cosine similarity (see Table 7.6).



(a) Low-Resolution Many-to-One

(b) High-Resolution Selective Mapping

Figure 7.2: Comparison of aspect mappings. For HRSMap, aspects 3, 7, and 8 are not mapped to gold-standard aspects.

Aspect Mapping

Most unsupervised aspect detection methods focus on the coherence and meaningfulness of model-inferred aspects, and prefer to map every model-inferred aspect (**MIA**) to a gold-standard aspect (**GSA**) [43]. Here, we call this mapping as **many-to-one mapping**, since the number of model-inferred aspects are usually larger than the number of gold-standard aspects. Weakly supervised approaches leverage human-annotated datasets to extract the aspect representative words, so that model-inferred aspects and gold-standard aspects have **one-to-one mapping** [1]. Different from the two mapping strategies described above, we propose a **high-resolution selective mapping (HRSMap)** strategy as shown in Fig. 7.2. Here, high-resolution means that the number of model-inferred aspects should be at least 3 times more than the number of gold-standard aspects, so that model-inferred aspects have a better coverage. Selective mapping means noisy or meaningless aspects will not be mapped to gold-standard aspects.

In our experiments, we set the number of MIAs to 30, considering the balance between aspect coverage and human-effort to manually map them to GSAs[1]. First, we automatically generate keywords of MIAs based aspect interpretation results, where the number of the most relevant keywords for each aspect is set to 10. Second, we create several rules for aspect mapping: (i) If keywords of a MIA are clearly related to one specific GSA (not *General*), we map this MIA to the GSA. For example, we map "*apps, app, netflix, browser, hulu, youtube, stream*" to *Apps/Interface* (see Table 7.6). (ii) If keywords are coherent but not related to any specific GSA, we map this MIA to *General*. For instance, we map "*pc, xbox, dvd, ps3, file, game*" to *General*. (iii) If keywords are related to more than one GSA, we treat this MIA as a noisy aspect and it will not be mapped. For example, "*excellent, amazing, good, great, outstanding, fantastic, impressed, superior*" may be related to several different GSAs. (iv) If keywords

[1] Usually, it takes less than 15 minutes to assign 30 MIAs to GSAs.

are not quite meaningful, their corresponding MIA will not be mapped. For instance, "*ago, within, last 30, later, took, couple, per, every*" is a meaningless MIA. Third, we further verify the quality of aspect mapping using development sets.

Given the soft-labels of model-inferred aspects β, we calculate soft-labels $\gamma = \{\gamma_1, \gamma_2, ..., \gamma_K\}$ over gold-standard aspects for each review segment as follows:

$$\gamma_k = \sum_{n=1}^{N} \mathbb{I}_{[f(\beta_n)=\gamma_k]}\beta_n, \tag{7.8}$$

where $f(\beta_n)$ is the aspect mapping for model-inferred aspect n. The hard-label $\hat{y}$ of gold-standard aspects for the segment is obtained by

$$\hat{y} = \text{argmax}\{\gamma_1, \gamma_2, ...\gamma_K\}, \tag{7.9}$$

which can be converted to a one-hot vector with length K.

7.2.3 Knowledge Distillation

Given both soft- and hard-labels of gold-standard aspects for review segments, we utilize a simple knowledge distillation method, which can be viewed as **classification on noisy labeled data**. We construct a simple classification model, which consists of a segment encoder such as BERT encoder [26], a smooth self-attention layer (see Eq. (7.2)), and a classifier (i.e., a single-layer feed-forward network followed by a softmax activation). This model is denoted by SSCLS, where the last S represents **student**. SSCLS learns knowledge from the **teacher** model, i.e., SSCL. The loss function is defined as

$$\mathcal{L} = -\frac{1}{K}\sum_{k=1}^{K} \mathbb{I}_{[H(\gamma)<\xi_k]} \cdot \hat{y}_k \log(y_k), \tag{7.10}$$

where y_k is the probability of aspect k predicted by SSCLS. $\hat{y}_k$ is a hard-label given by SSCL. $H(\gamma)$ represents the Shannon entropy for the soft-labels and is calculated by $H = -\sum_{k=1}^{K}\gamma_k \log(\gamma_k)$. Here, the scalar $\xi_k = \chi_{\text{G}}$ if aspect k is *General* and $\xi_k = \chi_{\text{NG}}$, otherwise. Both χ_{G} and χ_{NG} are hyperparameters. Hereafter, we will refer to $\mathbb{I}_{[H(\gamma)<\xi_k]}$ as an **Entropy Filter**.

Entropy scores have been used to evaluate the confidence of predictions [90]. In the training stage, we set thresholds to filter out training samples with low confidence predictions from the SSCL model, thus allowing the student model to focus on training samples for which the model prediction are more confident. Moreover, the student model also benefits from pre-trained encoders and overcomes the disadvantages of data pre-processing for SSCL, since we have removed out-of-vocabulary words and punctuation, and lemmatized tokens in SSCL. Therefore, SSCLS achieves better performance in segment aspect predictions compared to SSCL.

Table 7.2: The annotated aspects for Amazon reviews across different domains.

Domains	**Aspects**
Bags	Compartments, Customer Service, Handles, Looks, Price, Quality, Protection, Size/Fit, General.
Bluetooth	Battery, Comfort, Connectivity, Durability, Ease of Use, Look, Price, Sound, General
Boots	Color, Comfort, Durability, Look, Materials, Price, Size, Weather Resistance, General
Keyboards	Build Quality, Connectivity, Extra Function, Feel Comfort, Layout, Looks, Noise, Price, General
TVs	Apps/Interface, Connectivity, Customer Service, Ease of Use, Image, Price, Size/Look, Sound, General
Vacuums	Accessories, Build Quality, Customer Service, Ease of Use, Noise, Price, Suction Power, Weight, General

Table 7.3: The vocabulary size and the number of segments in each dataset. **Vocab** and **W2V** represent vocabulary size and word2vec, respectively.

Dataset	**Vocab**	**W2V**	**Train**	**Dev**	**Test**
Citysearch	9,088	279,862	279,862	2,686	1,490
Bags	6,438	244,546	584,332	598	641
B/T	9,619	573,206	1,419,812	661	656
Boots	6,710	408,169	957,309	548	611
KBs	6,904	241,857	603,379	675	681
TVs	10,739	579,526	1,422,192	699	748
VCs	9,780	588,369	1,453,651	729	725

7.3 Experiments

7.3.1 Datasets

We train and evaluate our methods on seven datasets: Citysearch restaurant reviews [33] and Amazon product reviews [1] across six different domains, including Laptop Cases (Bags), Bluetooth Headsets (B/T), Boots, Keyboards (KBs), Televisions (TVs), and Vacuums (VCs).

The Citysearch dataset only has training and testing sets. To avoid optimizing any models on the testing set, we use restaurant subsets of SemEval 2014 [110] and SemEval 2015 [109] datasets as a development set, since they adopt the same aspect labels as Citysearch. Similar to previous work [43], we select sentences that only express one aspect, and disregard those with multiple and no aspect labels. We have also restricted ourselves to three labels (Food, Service, and Ambience), to form a fair comparison with prior work [141]. Amazon product reviews are obtained from the OPOSUM dataset [1]. Different from Citysearch, EDUs [91] are used as segments and each domain has eight representative aspect labels as well as aspect *General* (see Table 7.2).

In order to train *SSCL*, all reviews are preprocessed by removing punctuation, stop-words, and less frequent words (<10). For Amazon reviews, reviews are segmented into elementary discourse units (EDUs) through a Rhetorical Structure Theory parser [29]. We have converted EDUs back to sentences to avoid training word2vec [96] on very short segments. However, we still use EDU-segments for training and evaluating different models following previous work [1]. Table 7.3 shows statistics of different datasets.

7.3.2 Comparison Methods

We compare our methods against five baselines on the Citysearch dataset.

- **SERBM** [148] is a sentiment-aspect extraction restricted Boltzmann machine, which jointly extracts review aspects and sentiment polarities in an unsupervised manner.
- **W2VLDA** [34] is a topic modeling based approach, which combines word embeddings [96] with Latent Dirichlet Allocation [6]. It automatically pairs discovered topics with pre-defined aspect names based on user provided seed-words for different aspects.
- **ABAE** [43] is an autoencoder that aims at learning highly coherent aspects by exploiting the distribution of word co-occurrences using neural word embeddings, and an attention mechanism that can put emphasis on aspect-related keywords in segments during training.
- **AE-CSA** [87] improves ABAE by leveraging sememes to enhance lexical semantics, where sememes are obtained via WordNet [97].
- **CAt** [141] is a simple heuristic model that consists of a contrastive attention mechanism based on Radial Basis Function kernels and an automated aspect assignment method.

For Amazon reviews, we compare our methods with several weakly supervised baselines, which explicitly leverage seed words extracted from human annotated development sets [55] as supervision for aspect detection.

- **ABAE**$_{init}$ [1] replaces each aspect embedding vector in ABAE with the corresponding centroid of seed word embeddings, and fixes aspect embedding vectors during training.
- **MATE** [1] uses the weighted average of seed word embeddings to initialize aspect embeddings. **MATE-MT** extends MATE by introducing an additional multi-task training objective.
- **TS-*** [55] is a weakly supervised student-teacher co-training framework, where **TS-Teacher** is a bag-of-words classifier (teacher) based on seed words. **TS-Stu-W2V** and **TS-Stu-BERT** are student networks that use word2vec embeddings and the BERT model to encode text segments, respectively.

7.3.3 Implementation Details

We implemented all deep learning models using PyTorch [105]. For each dataset, the best parameters and hyperparameters are selected based on the development set.

For our SSCL model, word embeddings are pre-loaded with 128-dimensional word vectors trained by the skip-gram model [96] with negative sampling and fixed during training. For each dataset, we use gensim[2] to train word embeddings from scratch and set both window and negative sample size to 5. The aspect embedding matrix is initialized with the centroids of clusters by running k-means on word embeddings. We set the number of aspects to 30 for all datasets because the model can achieve competitive performance while it will still be relatively easier to map model-inferred aspects to gold-standard aspects. The smooth factor λ is tuned in $\{0.5, 1.0, 2.0, 3.0, 4.0, 5.0\}$ and set to 0.5 for all datasets. The temperature μ is set to 1. For SSCLS, we have experimented with two pretrained encoders, i.e., BERT [26] and DistilBERT [119]. We tune smoothing factor λ in $\{0.5, 1.0\}$, χ_{G} in $\{0.7, 0.8, 1.0, 1.2\}$, and χ_{NG} in $\{1.4, 1.6, 1.8\}$. We set $\chi_{\text{G}} < \chi_{\text{NG}}$ to alleviate the label imbalance problem, since the majority of sentences in the corpus are labeled as *General*.

For both SSCL and SSCLS, model parameters are optimized using the Adam optimizer [63] with $\beta_1 = 0.9, \beta_2 = 0.999$, and $\epsilon = 10^{-8}$. Batch size is set to 50. For learning rates, we adopt a warmup schedule strategy proposed in [143], and set warmup step to 2000 and model size to 10^5. Gradient clipping with a threshold of 2 has also been applied to prevent gradient explosion. Our codes are available at `https://github.com/tshi04/AspDecSSCL`.

7.3.4 Performance on Amazon Product Reviews

Following previous works [1, 55], we use micro-averaged F1 score as our evaluation metric to measure the aspect detection performance among different models on Amazon product reviews. All results are shown in Table 7.4, where we use **bold** font to highlight the best performance values. The results of the compared models are obtained from the corresponding published papers. From this table, we can observe that weakly supervised ABAE$_{init}$, MATE and MATE-MT perform significantly better than unsupervised ABAE since they leverage aspect representative words extracted from human-annotated datasets and this leads to more accurate aspect predictions. TS-Teacher outperforms MATE and MATE-MT on most of the datasets, which further demonstrates that these words are highly correlated with gold-standard aspects. The better performance of both TS-Stu-W2V and TS-Stu-BERT over TS-Teacher demonstrates the effectiveness of their teacher-student co-training framework.

In our experiments, we conjecture that low-resolution many-to-one aspect mapping may be one of the reasons for the low performance of traditional ABAE. Therefore, we have re-implemented ABAE and combined it with HRSMap. The new model (i.e., ABAE +

[2] `https://radimrehurek.com/gensim/`

Table 7.4: Micro-averaged F1 scores for 9-class EDU-level aspect detection in Amazon reviews. **AVG** denotes the average of F1 scores across all domains.

Methods	**Bags**	**B/T**	**Boots**	**KBs**	**TVs**	**VCs**	**AVG**
Unsupervised Methods							
ABAE [43]	38.1	37.6	35.2	38.6	39.5	38.1	37.9
ABAE + HRSMap	54.9	62.2	54.7	58.9	59.9	54.1	57.5
Weakly Supervised Methods							
ABAE$_{init}$ [1]	41.6	48.5	41.2	41.3	45.7	40.6	43.2
MATE [1]	46.2	52.2	45.6	43.5	48.8	42.3	46.4
MATE-MT [1]	48.6	54.5	46.4	45.3	51.8	47.7	49.1
TS-Teacher [55]	55.1	50.1	44.5	52.0	56.8	54.5	52.2
TS-Stu-W2V [55]	59.3	66.8	48.3	57.0	64.0	57.0	58.7
TS-Stu-BERT [55]	61.4	66.5	52.0	57.5	63.0	60.4	60.2
Our Models							
SSCL	61.0	65.2	57.3	60.6	64.6	57.2	61.0
SSCLS-BERT	**65.5**	**69.5**	60.4	**62.3**	**67.0**	**61.0**	**64.3**
SSCLS-DistilBERT	64.7	68.4	**61.0**	62.0	66.3	59.9	63.7

HRSMap) obtains significantly better results compared to the traditional ABAE on all datasets (performance improvement of 51.7%), showing HRSMap is effective in mapping model-inferred aspects to gold-standard aspects. Compared to the TS-* baseline methods, our SSCL achieves better results on Boots, KBs, and TVs, and competitive results on Bags, B/T, and VCs. On average, it outperforms TS-Teacher, TS-Stu-W2V, and TS-Stu-BERT by 16.9%, 3.9%, and 1.3%, respectively. SSCLS-BERT and SSCLS-DistilBERT further boost the performance of SSCL by 5.4% and 4.4%, respectively, thus demonstrating that knowledge distillation is effective in improving the quality of aspect prediction.

7.3.5 Performance on Restaurant Reviews

We have conducted more detailed comparisons on the Citysearch dataset, which has been widely used to benchmark aspect detection models. Following previous work [141], we use weighted macro averaged precision, recall and F1 score as metrics to evaluate the overall performance. We also evaluate performance of different models for three major individual aspects by measuring aspect-level precision, recall, and F1 scores. Experimental results are presented in Table 7.5. Results of compared models are obtained from the corresponding published papers.

From Table 7.5, we also observe that ABAE + HRSMap performs significantly better than traditional ABAE. Our SSCL outperforms all baselines in terms of weighted macro averaged F1 score. SSCLS-BERT and SSCLS-DistilBERT further improve the performance of SSCL,

Table 7.5: Aspect-level precision (**P**), recall (**R**), and F-scores (**F**) on the Citysearch testing set. For overall, we calculate weighted macro averages across all aspects.

	Food			**Staff**			**Ambience**			**Overall**		
Methods	**P**	**R**	**F**	**P**	**R**	**F**	**P**	**R**	**F**	**P**	**R**	**F**
SERBM [148]	89.1	85.4	87.2	81.9	58.2	68.0	80.5	59.2	68.2	86.0	74.6	79.5
ABAE [43]	95.3	74.1	82.8	80.2	72.8	75.7	81.5	69.8	74.0	89.4	73.0	79.6
W2VLDA [34]	96.0	69.0	81.0	61.0	86.0	71.0	55.0	75.0	64.0	80.8	70.0	75.8
AE-CSA [87]	90.3	92.6	91.4	92.6	75.6	77.3	91.4	77.9	77.0	85.6	86.0	85.8
CAt [141]	91.8	92.4	92.1	82.4	75.6	78.8	76.6	80.1	76.6	86.5	86.4	86.4
ABAE + HRSMap	93.0	88.8	90.9	85.8	75.3	80.2	67.4	89.6	76.9	87.0	85.8	86.0
SSCL	91.7	94.6	93.1	88.4	75.9	81.7	79.1	86.1	82.4	88.8	88.7	88.6
SSCLS-BERT	89.6	97.3	93.3	95.5	71.9	82.0	84.0	87.6	85.8	90.0	89.7	89.4
SSCLS-DistilBERT	91.3	96.6	**93.9**	92.4	75.9	**83.3**	84.4	88.0	**86.2**	90.4	90.3	**90.1**

and SSCLS-DistilBERT achieves the best results. From aspect-level results, we can observe that, for each individual aspect, our SSCL, SSCLS-BERT and SSCLS-DistilBERT performs consistently better than compared baseline methods in terms of F1 score. SSCLS-DistilBERT gets the best F1 scores across all three aspects. This experiment demonstrates the strength of the contrastive learning framework, HRSMap, and knowledge distillation, which are able to capture high-quality aspects, effectively map model-inferred aspects to gold-standard aspects, and accurately predict aspect labels for the given segments.

7.3.6 Aspect Interpretation

As SSCL achieves promising performance quantitatively on aspect detection compared to the baselines, we further show some qualitative results to interpret extracted concepts. From Table 7.6, we notice that there is at least one model-inferred aspect corresponding to each of the gold-standard aspects, which indicates model-inferred aspects based on HRSMap have a good coverage. We also find that model-inferred concepts, which are mapped to non-general gold-standard aspects, are fine-grained, and their representative words are meaningful and coherent. For example, it is easy to map "*app, netflix, browser, hulu, youtube*" to *Apps/Interface*. Compared to weakly supervised methods (such as MATE), SSCL is also able to discover new concepts. For example, for aspects mapped to *General*, we may label "*pc, xbox, dvd, ps3, file, game*" as *Connected Devices*, and "*plastic glass screw piece metal base*" as *Build Quality*. Similarly, we observe that model-inferred aspects based on Bluetooth Headsets reviews also have sufficient coverage for gold-standard aspects (see Table 7.7). We can easily map model inferred aspects to gold-standard ones since their keywords are meaningful and coherent. For instance, it is obvious that "*red, light, blinking, flashing, color, blink*" are related to *Look* and "*charge, recharge, life, standby, battery, drain*" are about *Battery*. For new aspect detection, "*motorola, model, plantronics, voyager, backbeatjabra*" can be interpreted as *Brand*. "*player, video, listen, streaming, movie, pandora*" are about *Usage*.

Table 7.6: Left: Gold-standard aspects for TVs reviews. Right: Model-inferred aspects presented by representative words.

Aspects	**Representative Keywords**
Apps/Interface	apps app netflix browser hulu youtube
Connectivity	channel antenna broadcast signal station
	optical composite hdmi input component
Customer Serv.	service process company contact support
	call email contacted rep phone repair
Ease of Use	button remote keyboard control use qwerty
Image	setting brightness mode contrast color
	motion scene blur action movement effect
Price	dollar cost buck 00 pay tax
Size/Look	32 42 37 46 55 40
Sound	speaker bass surround volume sound stereo
General	forum read reading review cnet posted
	recommend research buy purchase decision
	plastic glass screw piece metal base
	foot wall mount stand angle cabinet
	football watch movie kid night game
	pc xbox dvd ps3 file game
	series model projection plasma led sony

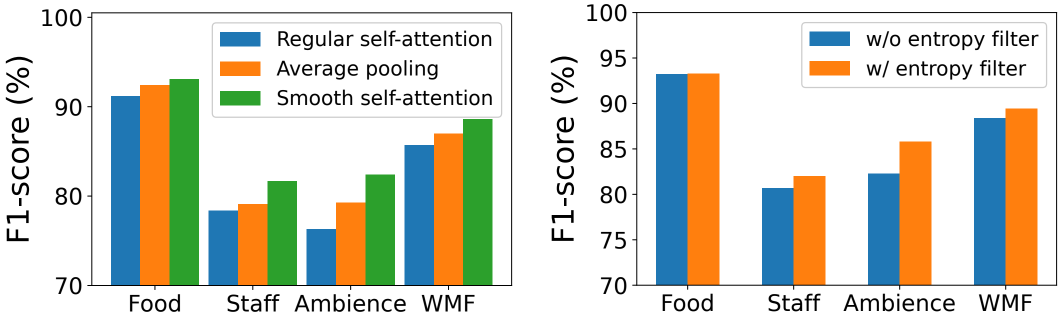
7.3.7 Ablation Study and Parameter Sensitivity

In addition to self-supervised contrastive learning framework and HRSMap, we also attribute the promising performance of our models to (i) Smooth self-attention mechanism, (ii) Entropy filters, and (iii) Appropriate batch size. Hence, we systematically conduct ablation studies and parameter sensitivity analysis to demonstrate the effectiveness of them, and provide the results in Fig. 7.3 and Fig. 7.4.

First, we replace the smooth self-attention (SSA) layer with a regular self-attention (RSA) layer used in [1] and an average pooling (AP) layer. The model with SSA performs better than the one with AP or RSA. Next, we examine the entropy filter for SSCLS-BERT, and observe that adding it has a positive impact on the model performance. Then, we study the effect of smoothness factor λ in SSA and observe that our model achieves promising and stable results when $\lambda \leq 1$. Finally, we investigate the effect of batch size. F1 scores increase with batch size and become stable when batch size is greater than 20. However, very large batch size increases the computational complexity; see Algorithm 2. Therefore, we set batch size to 50 for all our experiments.

Table 7.7: Left: Gold-standard aspects for Bluetooth Headsets reviews. Right: Model inferred aspects presented by representative words.

Aspects	**Representative Keywords**
Battery	charge recharge life standby battery drain
Comfort	uncomfortable hurt sore comfortable tight pressure
Connectivity	usb cable charger adapter port ac
	paired htc galaxy android macbook connected
Durability	minute hour foot day min second
Ease of Use	button pause track control press forward
Look	red light blinking flashing color blink
Price	00 buck spend paid dollar cost
Sound	bass high level low treble frequency
	noisy wind environment noise truck background
General	rating flaw consider star design improvement
	christmas gift son birthday 2013 new husband
	warranty refund shipping contacted sent email
	motorola model plantronics voyager backbeat jabra
	gym walk house treadmill yard kitchen
	player video listen streaming movie pandora
	read reading website manual web review
	purchased bought buying ordered buy purchase



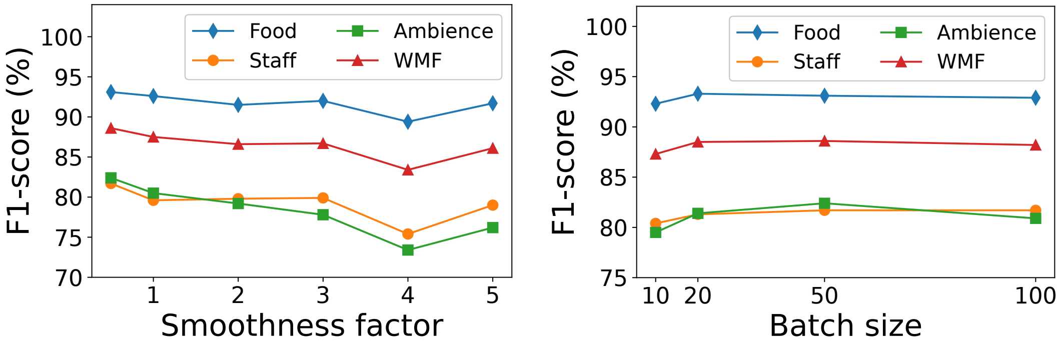
(a) Smooth Self-Attention

(b) Entropy Filter

Figure 7.3: Ablation study on the Citysearch testing set. **WMF** represents weighted macro averaged F1-score.

7.3.8 Case Study

Fig. 7.5 compares heat-maps of attention weights obtained from SSA and RSA on two segments from the Amazon TVs testing set. In each example, RSA attempts to use a single word to represent the entire segment. However, the word may be either a representative



(a) F1 vs. smoothness factor λ

(b) F1 vs. Batch Size

Figure 7.4: Parameter sensitivity analysis on Citysearch.

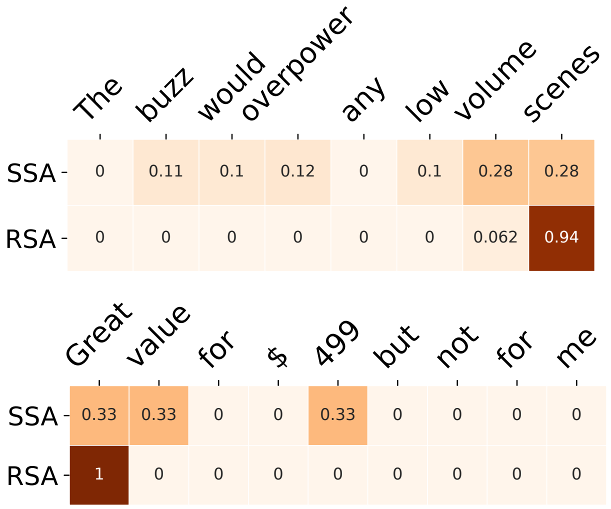


Figure 7.5: Visualization of attention weights. SSA and RSA represent smooth and regular self-attention, respectively.

word for another aspect (e.g., "*scene*" for *Image* in Table 7.6) or a word with no aspect tendency (e.g., "*great*" is not assigned to any aspect). In contrast, SSA captures phrases and multiple words, e.g., "*volume scenes*" and "*great value, 499*". Based on the results in Fig. 7.3 and Fig. 7.5, we argue SSA is more robust and intuitively meaningful than RSA for aspect detection.

7.4 Summary

In this study, we propose a self-supervised contrastive learning framework for aspect detection. Our model is equipped with two attention modules, which allows us to represent every segment with word embeddings and aspect embeddings, so that we can map aspect embeddings to the word embedding space through a contrastive learning mechanism. In the attention module over word embeddings, we introduce a SSA mechanism. Thus, our model can learn robust representations, since SSA encourages the model to capture phrases and multiple keywords in the segments. In addition, we propose a HRSMap method for aspect mapping, which dramatically increases the accuracy of segment aspect predictions for both ABAE and our model. Finally, we further improve the performance of aspect detection through knowledge distillation. BERT-based student models can benefit from pretrained encoders and overcome the disadvantages of data preprocessing for the teacher model. During training, we introduce entropy filters in the loss function to ensure student models focus on high confidence training samples. Our models have shown better performance compared to several recent unsupervised and weakly-supervised models on several publicly available review datasets across different domains. Aspect interpretation results show that extracted aspects are meaningful, have a good coverage, and can be easily mapped to gold-standard aspects. Ablation studies and visualization of attention weights further demonstrate the effectiveness of SSA and entropy filters.

Chapter 8

Conclusion and Future Work

8.1 Conclusion

The main goal of this dissertation is to develop innovative solutions to understand online customer reviews and learn structured knowledge from them. To achieve this goal, we studied the review understanding problem in three directions, including corpus-level, document-level and sentence-level review understanding, which are associated with many NLP tasks. In this dissertation, we primarily focus on three tasks, i.e., topic modeling, sentiment analysis, and aspect detection. We have developed machine learning techniques based on unsupervised, multi-task and self-supervised learning frameworks to deal with the challenges in these tasks.

For the topic modeling task, we introduced a SeaNMF model to discover topics for the short texts and use a block coordinate descent algorithm to infer parameters for our SeaNMF model. We also developed a sparse SeaNMF model in order to get a better interpretability. Extensive quantitative evaluations on various real-world short text datasets demonstrate the superior performance of the proposed models over several other state-of-the-art methods in terms of topic coherence and classification accuracy. The qualitative semantic analysis demonstrates the interpretability of our models by discovering meaningful and consistent topics. With a simple formulation and the superior performance, SeaNMF can be an effective standard topic model for short texts.

For the document-level multi-aspect sentiment analysis task, we systematically investigated the dataset from the `ratemds.com` review platform, where each review for a doctor comes with an overall rating and ratings of four different aspects. We also proposed a multi-task learning framework for the document-level multi-aspect sentiment classification. Extensive experiments have been conducted on two subsets of the ratemds dataset to demonstrate effectiveness of the proposed model. Qualitative results show the power of attention mechanisms and reveal some linguistic problems in the textual reviews.

In order to improve the interpretability of attention-based deep sentiment classification models, we proposed a general-purpose corpus-level explanation approach, which can capture causal relationships between keywords and model predictions via learning importance of keywords for predicted labels across a training corpus based on attention weights. Experimental results have shown that the keywords are semantically meaningful for predicted labels. We further proposed a concept-based explanation method to identify important concepts for model predictions. Our experimental results also demonstrate that this method effectively captures semantically meaningful concepts. It also provides the relative importance of each concept to model predictions.

For the document-level multi-aspect sentiment analysis task, we have also focused on interpretability and reliability of our proposed model. We have developed a deliberate self-attention based deep neural network model, which can achieve competitive performance while also being able to interpret the predictions made. We proposed an attention-driven keywords ranking method, which is based on the corpus-level explanation approach and can automatically discover aspect keywords and aspect-level opinion keywords from a review corpus based on the attention weights. In addition, we proposed a lecture-audience strategy to estimate model uncertainty in the context of multi-task learning. Our extensive set of experiments on five different open-domain datasets demonstrate the superiority of the proposed models. We further introduced two new datasets in the healthcare domain and benchmark different baseline models and our models on them. Attention weights visualization results and visualization of aspect and opinion keywords demonstrate the interpretability of our models.

For the aspect detection task, we proposed a self-supervised contrastive learning framework and an attention-based model equipped with a novel smoothing self-attention module in order to learn better representations for aspects and review segments. We also introduced a high-resolution selective mapping method to efficiently assign aspects discovered by the model to the aspects of interest. In addition, we proposed using a knowledge distillation technique to further improve the aspect detection performance. Our methods outperform several recent unsupervised and weakly supervised approaches on publicly available benchmark user review datasets. Aspect interpretation results show that extracted aspects are meaningful, have a good coverage, and can be easily mapped to aspects of interest.

8.2 Future Work

In the future, there are many ways to advance techniques for understanding online customer reviews. In this section, some research directions are discussed as follows:

8.2.1 Mining Structured Knowledge from Reviews

Several tasks in ABSA, including aspect term extraction, opinion term extraction, and opinion-target detection, play a fundamental role in review understanding. They enable us to convert reviews into a structured knowledge base that will benefit many downstream tasks, such as review summarization and question-answering. As aforementioned, most deep learning methods for ABSA rely on fully supervised training, so their applications are limited to a few areas with annotated corpora. In order to have a broad impact in other domains, unsupervised, weakly-supervised, and transfer learning methods will continue to be investigated to deal with these problems. There are many interesting research questions, such as 1) Can we apply open-domain entity extraction techniques (e.g., phrase extraction) to the aspect and opinion term extraction tasks? 2) Can we use a distant-supervision approach to deal with the opinion-target detection problem?

In practical applications, we can use the discovered knowledge (i.e., aspects and sentiment) to analyze changes of aspects/topics over time for products and services. For example, for TVs, customers may have been interested in high-resolution LCD screens 10 years ago. Nowadays, they may be more interested in OLED screens and smart features. Personalization is another direction that has gained attention. By considering user groups, we can apply extracted aspects to the development of recommender systems.

8.2.2 Review-Based Natural Language Generation

Other research tasks include question-answering (question generation and answer generation), multi-document summarization, and recommendation justification tasks in the direction of NLG for online reviews, and plan to deal with two common challenges: 1) *Lack of ground-truth or paired examples.* In these tasks, we do not have paired answers, ground-truth summaries or justification available for training natural language generation models. Therefore, there are two strategies to solve this problem, including creating synthetic datasets (e.g., pseudo paired examples) based on aspect-based sentiment analysis and developing unsupervised deep learning models (e.g., autoencoder based models). 2) *Fact correctness.* These tasks are conditioned on multiple reviews with different facts and it is difficult to generate a coherent story while retaining the facts. To solve this problem, clustering methods will be studied to group reviews or review segments based on their aspect and sentiment. Text matching or natural language inference methods will also be considered to show if different review segments have the same meaning or not. In addition, automatic evaluation metrics will be investigated to evaluate the fact correctness of generated content.

Bibliography

[1] S. Angelidis and M. Lapata. 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*, pages 3675–3686, Brussels, Belgium, Oct.-Nov. 2018. Association for Computational Linguistics.

[2] D. Antognini and B. Faltings. Rationalization through concepts. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 761–775, Online, Aug. 2021. Association for Computational Linguistics.

[3] D. Antognini, C. Musat, and B. Faltings. Interacting with explanations through critiquing. *arXiv preprint arXiv:2005.11067*, 2020.

[4] D. Bahdanau, K. Cho, and Y. Bengio. Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473*, 2014.

[5] I. Beltagy, M. E. Peters, and A. Cohan. Longformer: The long-document transformer. *arXiv preprint arXiv:2004.05150*, 2020.

[6] D. M. Blei, A. Y. Ng, and M. I. Jordan. Latent Dirichlet Allocation. *J. Mach. Learn. Res.*, 3:993–1022, Mar. 2003.

[7] C. Blundell, J. Cornebise, K. Kavukcuoglu, and D. Wierstra. Weight uncertainty in neural networks. In *Proceedings of the 32nd International Conference on International Conference on Machine Learning - Volume 37*, ICML'15, page 1613–1622. JMLR.org, 2015.

[8] F. Bodria, F. Giannotti, R. Guidotti, F. Naretto, D. Pedreschi, and S. Rinzivillo. Benchmarking and survey of explanation methods for black box models. *arXiv preprint arXiv:2102.13076*, 2021.

[9] D. Bouchacourt and L. Denoyer. EDUCE: Explaining model decisions through unsupervised concepts extraction. *arXiv preprint arXiv:1905.11852*, 2019.

[10] S. Brody and N. Elhadad. 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, Los Angeles, California, June 2010. Association for Computational Linguistics.

[11] M. Ceyhan, Z. Orhan, and E. Domnori. Health service quality measurement from patient reviews in Turkish by opinion mining. In A. Badnjevic, editor, *CMBEBIH 2017*, pages 649–653, Singapore, 2017. Springer Singapore.

[12] H. Chen, M. Sun, C. Tu, Y. Lin, and Z. Liu. Neural sentiment classification with user and product attention. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 1650–1659, Austin, Texas, Nov. 2016. Association for Computational Linguistics.

[13] L. Chen, G. Chen, and F. Wang. Recommender systems based on user reviews: the state of the art. *User Modeling and User-Adapted Interaction*, 25(2):99–154, Jun 2015.

[14] T. Chen, S. Kornblith, M. Norouzi, and G. Hinton. A simple framework for contrastive learning of visual representations. In *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pages 1597–1607. PMLR, 13–18 Jul 2020.

[15] T. Chen, Y. Sun, Y. Shi, and L. Hong. On sampling strategies for neural network-based collaborative filtering. In *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '17, page 767–776, New York, NY, USA, 2017. Association for Computing Machinery.

[16] Z. Chen, Y. Bei, and C. Rudin. Concept whitening for interpretable image recognition. *Nature Machine Intelligence*, 2(12):772–782, Dec 2020.

[17] Z. Chen, A. Mukherjee, and B. Liu. Aspect extraction with automated prior knowledge learning. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 347–358. ACL, 2014.

[18] K. Cho, B. van Merriënboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio. 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, Doha, Qatar, Oct. 2014. Association for Computational Linguistics.

[19] J. Choo, C. Lee, C. K. Reddy, and H. Park. UTOPIAN: User-driven topic modeling based on interactive nonnegative matrix factorization. *IEEE Transactions on Visualization and Computer Graphics*, 19(12):1992–2001, 2013.

[20] J. Choo, C. Lee, C. K. Reddy, and H. Park. Weakly supervised nonnegative matrix factorization for user-driven clustering. *Data Mining and Knowledge Discovery*, 29(6):1598–1621, Nov. 2015.

[21] E. Chu and P. Liu. MeanSum: A neural model for unsupervised multi-document abstractive summarization. In K. Chaudhuri and R. Salakhutdinov, editors, *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pages 1223–1232. PMLR, 09–15 Jun 2019.

[22] A. Cichocki, R. Zdunek, A. H. Phan, and S. Amari. *Nonnegative matrix and tensor factorizations: applications to exploratory multi-way data analysis and blind source separation*. John Wiley & Sons, 2009.

[23] R. Collobert, J. Weston, L. Bottou, M. Karlen, K. Kavukcuoglu, and P. Kuksa. Natural language processing (almost) from scratch. *The Journal of Machine Learning Research*, 12:2493–2537, Nov. 2011.

[24] A. Das and P. Rad. Opportunities and challenges in explainable artificial intelligence (XAI): A survey. *arXiv preprint arXiv:2006.11371*, 2020.

[25] S. Deerwester, S. T. Dumais, G. W. Furnas, T. K. Landauer, and R. Harshman. Indexing by latent semantic analysis. *Journal of the American Society for Information Science*, 41(6):391–407, 1990.

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

[27] R.-E. Fan, K.-W. Chang, C.-J. Hsieh, X.-R. Wang, and C.-J. Lin. LIBLINER: A library for large linear classification. *The Journal of Machine Learning Research*, 9:1871–1874, June 2008.

[28] Z. Fan, Z. Wu, X.-Y. Dai, S. Huang, and J. Chen. Target-oriented opinion words extraction with target-fused neural sequence labeling. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 2509–2518, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics.

[29] V. W. Feng and G. Hirst. A linear-time bottom-up discourse parser with constraints and post-editing. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 511–521, Baltimore, Maryland, June 2014. Association for Computational Linguistics.

[30] N. Friedman, D. Geiger, and M. Goldszmidt. Bayesian network classifiers. *Machine Learning*, 29(2):131–163, Nov 1997.

[31] Y. Gal. *Uncertainty in Deep Learning*. PhD thesis, University of Cambridge, 2016.

[32] Y. Gal and Z. Ghahramani. Dropout as a Bayesian approximation: Representing model uncertainty in deep learning. In M. F. Balcan and K. Q. Weinberger, editors, *Proceedings of The 33rd International Conference on Machine Learning*, volume 48 of *Proceedings of Machine Learning Research*, pages 1050–1059, New York, New York, USA, 20–22 Jun 2016. PMLR.

[33] G. Ganu, N. Elhadad, and A. Marian. Beyond the stars: improving rating predictions using review text content. In *Proceedings of the 12th International Workshop on the Web and Databases (WebDB 2009)*, pages 1–6, 2009.

[34] A. García-Pablos, M. Cuadros, and G. Rigau. W2VLDA: Almost unsupervised system for aspect based sentiment analysis. *Expert Systems with Applications*, 91:127–137, 2018.

[35] R. Ghaeini, X. Fern, and P. Tadepalli. Interpreting recurrent and attention-based neural models: a case study on natural language inference. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 4952–4957, Brussels, Belgium, Oct.-Nov. 2018. Association for Computational Linguistics.

[36] A. Ghorbani, A. Abid, and J. Zou. Interpretation of neural networks is fragile. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 3681–3688, 2019.

[37] A. Ghorbani, J. Wexler, J. Y. Zou, and B. Kim. Towards automatic concept-based explanations. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019.

[38] L. H. Gilpin, D. Bau, B. Z. Yuan, A. Bajwa, M. Specter, and L. Kagal. Explaining explanations: An overview of interpretability of machine learning. In *2018 IEEE 5th International Conference on Data Science and Advanced Analytics (DSAA)*, pages 80–89, 2018.

[39] S. Gohil, S. Vuik, and A. Darzi. Sentiment analysis of health care tweets: Review of the methods used. *JMIR Public Health Surveill*, 4(2):e43, Apr 2018.

[40] M. Grusky, M. Naaman, and Y. Artzi. Newsroom: A dataset of 1.3 million summaries with diverse extractive strategies. 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 708–719, New Orleans, Louisiana, June 2018. Association for Computational Linguistics.

[41] V. Gulshan, L. Peng, M. Coram, M. C. Stumpe, D. Wu, A. Narayanaswamy, S. Venugopalan, K. Widner, T. Madams, J. Cuadros, R. Kim, R. Raman, P. C. Nelson,

J. L. Mega, and D. R. Webster. Development and validation of a deep learning algorithm for detection of diabetic retinopathy in retinal fundus photographs. *JAMA*, 316(22):2402–2410, 12 2016.

[42] K. He, H. Fan, Y. Wu, S. Xie, and R. Girshick. Momentum contrast for unsupervised visual representation learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020.

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

[44] R. He and J. McAuley. 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*, WWW '16, page 507–517, Republic and Canton of Geneva, CHE, 2016. International World Wide Web Conferences Steering Committee.

[45] K. M. Hermann, T. Kocisky, E. Grefenstette, L. Espeholt, W. Kay, M. Suleyman, and P. Blunsom. Teaching machines to read and comprehend. In C. Cortes, N. Lawrence, D. Lee, M. Sugiyama, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 28. Curran Associates, Inc., 2015.

[46] G. Hinton, O. Vinyals, and J. Dean. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015.

[47] S. Hochreiter and J. Schmidhuber. Long short-term memory. *Neural computation*, 9(8):1735–1780, Nov. 1997.

[48] M. Hoffman, F. Bach, and D. Blei. Online learning for Latent Dirichlet Allocation. In J. Lafferty, C. Williams, J. Shawe-Taylor, R. Zemel, and A. Culotta, editors, *Advances in Neural Information Processing Systems*, volume 23. Curran Associates, Inc., 2010.

[49] T. Hofmann. Probabilistic latent semantic indexing. In *Proceedings of the 22nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '99, page 50–57, New York, NY, USA, 1999. Association for Computing Machinery.

[50] T. Hofmann. Probabilistic latent semantic indexing. In *Proceedings of the 22nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '99, page 50–57, New York, NY, USA, 1999. Association for Computing Machinery.

[51] L. Hong and B. D. Davison. Empirical study of topic modeling in Twitter. In *Proceedings of the First Workshop on Social Media Analytics*, SOMA '10, page 80–88, New York, NY, USA, 2010. Association for Computing Machinery.

[52] A. M. Hopper and M. Uriyo. Using sentiment analysis to review patient satisfaction data located on the internet. *Journal of Health Organization and Management*, 29(2):221–233, Jan 2015.

[53] S. Jain and B. C. Wallace. Attention is not Explanation. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 3543–3556, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics.

[54] W. Jin, H. H. Ho, and R. K. Srihari. OpinionMiner: A novel machine learning system for web opinion mining and extraction. In *Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '09, page 1195–1204, New York, NY, USA, 2009. Association for Computing Machinery.

[55] G. Karamanolakis, D. Hsu, and L. Gravano. Leveraging just a few keywords for fine-grained aspect detection through weakly supervised co-training. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 4611–4621, Hong Kong, China, Nov. 2019. Association for Computational Linguistics.

[56] A. Kendall, Y. Gal, and R. Cipolla. Multi-task learning using uncertainty to weigh losses for scene geometry and semantics. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018.

[57] Y. Keneshloo, T. Shi, N. Ramakrishnan, and C. K. Reddy. Deep reinforcement learning for sequence-to-sequence models. *IEEE transactions on neural networks and learning systems*, 31(7):2469–2489, 2019.

[58] B. Kim, M. Wattenberg, J. Gilmer, C. Cai, J. Wexler, F. Viegas, and R. Sayres. Interpretability beyond feature attribution: Quantitative testing with concept activation vectors (TCAV). In J. Dy and A. Krause, editors, *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pages 2668–2677. PMLR, 10–15 Jul 2018.

[59] H. Kim, J. Choo, J. Kim, C. K. Reddy, and H. Park. Simultaneous discovery of common and discriminative topics via joint nonnegative matrix factorization. In *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '15, pages 567–576, New York, NY, USA, 2015. ACM.

[60] J. Kim, Y. He, and H. Park. Algorithms for nonnegative matrix and tensor factorizations: a unified view based on block coordinate descent framework. *Journal of Global Optimization*, 58(2):285–319, Feb 2014.

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

[62] P.-J. Kindermans, S. Hooker, J. Adebayo, M. Alber, K. T. Schütt, S. Dähne, D. Erhan, and B. Kim. *The (Un)reliability of Saliency Methods*, pages 267–280. Springer International Publishing, Cham, 2019.

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

[64] D. Kuang, J. Choo, and H. Park. *Nonnegative Matrix Factorization for Interactive Topic Modeling and Document Clustering*, pages 215–243. Springer International Publishing, Cham, 2015.

[65] D. Kuang, C. Ding, and H. Park. Symmetric nonnegative matrix factorization for graph clustering. In *Proceedings of the 2012 SIAM International Conference on Data Mining (SDM)*, pages 106–117, 2019.

[66] D. Kuang, S. Yun, and H. Park. SymNMF: nonnegative low-rank approximation of a similarity matrix for graph clustering. *Journal of Global Optimization*, 62(3):545–574, Jul 2015.

[67] S. Kullback. *Information theory and statistics*. Courier Corporation, 1997.

[68] D. D. Lee and H. S. Seung. Learning the parts of objects by non-negative matrix factorization. *Nature*, 401(6755):788–791, Oct 1999.

[69] T. Lei, R. Barzilay, and T. Jaakkola. Rationalizing neural predictions. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 107–117, Austin, Texas, Nov. 2016. Association for Computational Linguistics.

[70] O. Levy and Y. Goldberg. Neural word embedding as implicit matrix factorization. In Z. Ghahramani, M. Welling, C. Cortes, N. Lawrence, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems*, volume 27. Curran Associates, Inc., 2014.

[71] O. Levy, Y. Goldberg, and I. Dagan. Improving distributional similarity with lessons learned from word embeddings. *Transactions of the Association for Computational Linguistics*, 3:211–225, 2015.

[72] C. Li, H. Wang, Z. Zhang, A. Sun, and Z. Ma. Topic modeling for short texts with auxiliary word embeddings. In *Proceedings of the 39th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '16, page 165–174, New York, NY, USA, 2016. Association for Computing Machinery.

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

[74] J. Li, H. Yang, and C. Zong. Document-level multi-aspect sentiment classification by jointly modeling users, aspects, and overall ratings. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 925–936, Santa Fe, New Mexico, USA, Aug. 2018. Association for Computational Linguistics.

[75] Z. Lin, M. Feng, C. N. d. Santos, M. Yu, B. Xiang, B. Zhou, and Y. Bengio. A structured self-attentive sentence embedding. *arXiv preprint arXiv:1703.03130*, 2017.

[76] G. Ling, M. R. Lyu, and I. King. Ratings meet reviews, a combined approach to recommend. In *Proceedings of the 8th ACM Conference on Recommender Systems*, RecSys '14, page 105–112, New York, NY, USA, 2014. Association for Computing Machinery.

[77] B. Liu. *Sentiment Analysis and Opinion Mining.* Morgan & Claypool Publishers, 2012.

[78] F. Liu and B. Avci. Incorporating priors with feature attribution on text classification. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 6274–6283, Florence, Italy, July 2019. Association for Computational Linguistics.

[79] L. Liu, H. Jiang, P. He, W. Chen, X. Liu, J. Gao, and J. Han. On the variance of the adaptive learning rate and beyond. In *International Conference on Learning Representations*, 2019.

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

[81] Q. Liu, B. Liu, Y. Zhang, D. S. Kim, and Z. Gao. Improving opinion aspect extraction using semantic similarity and aspect associations. In *Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence*, AAAI'16, page 2986–2992. AAAI Press, 2016.

[82] Y. Liu, K. Gadepalli, M. Norouzi, G. E. Dahl, T. Kohlberger, A. Boyko, S. Venugopalan, A. Timofeev, P. Q. Nelson, G. S. Corrado, et al. Detecting cancer metastases on gigapixel pathology images. *arXiv preprint arXiv:1703.02442*, 2017.

[83] Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer, and V. Stoyanov. RoBERTa: A robustly optimized BERT pretraining approach. *arXiv preprint arXiv:1907.11692*, 2019.

[84] C. Louizos and M. Welling. Structured and efficient variational deep learning with matrix Gaussian posteriors. In *Proceedings of the 33rd International Conference on International Conference on Machine Learning - Volume 48*, ICML'16, page 1708–1716. JMLR.org, 2016.

[85] B. Lu, M. Ott, C. Cardie, and B. K. Tsou. Multi-aspect sentiment analysis with topic models. In *2011 11th IEEE International Conference on Data Mining Workshops*, pages 81–88. IEEE, 2011.

[86] S. M. Lundberg and S.-I. Lee. A unified approach to interpreting model predictions. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, NIPS'17, page 4768–4777, Red Hook, NY, USA, 2017. Curran Associates Inc.

[87] L. Luo, X. Ao, Y. Song, J. Li, X. Yang, Q. He, and D. Yu. Unsupervised neural aspect extraction with sememes. In *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence, IJCAI-19*, pages 5123–5129. International Joint Conferences on Artificial Intelligence Organization, 7 2019.

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

[89] A. L. Maas, R. E. Daly, P. T. Pham, D. Huang, A. Y. Ng, and C. Potts. Learning word vectors for sentiment analysis. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 142–150, Portland, Oregon, USA, June 2011. Association for Computational Linguistics.

[90] A. Mandelbaum and D. Weinshall. Distance-based confidence score for neural network classifiers. *arXiv preprint arXiv:1709.09844*, 2017.

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

[92] R. McAllister, Y. Gal, A. Kendall, M. van der Wilk, A. Shah, R. Cipolla, and A. Weller. Concrete problems for autonomous vehicle safety: Advantages of Bayesian deep learning. In *Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence, IJCAI-17*, pages 4745–4753, 2017.

[93] J. McAuley and J. Leskovec. Hidden factors and hidden topics: Understanding rating dimensions with review text. In *Proceedings of the 7th ACM Conference on Recommender Systems*, RecSys '13, page 165–172, New York, NY, USA, 2013. Association for Computing Machinery.

[94] J. McAuley, J. Leskovec, and D. Jurafsky. Learning attitudes and attributes from multi-aspect reviews. In *2012 IEEE 12th International Conference on Data Mining*, pages 1020–1025, 2012.

[95] T. Mikolov, K. Chen, G. Corrado, and J. Dean. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*, 2013.

[96] T. Mikolov, I. Sutskever, K. Chen, G. Corrado, and J. Dean. Distributed representations of words and phrases and their compositionality. In *Proceedings of the 26th International Conference on Neural Information Processing Systems - Volume 2*, NIPS'13, page 3111–3119, Red Hook, NY, USA, 2013. Curran Associates Inc.

[97] G. A. Miller. WordNet: A lexical database for english. *Communications of the ACM*, 38(11):39–41, Nov. 1995.

[98] R. Miotto, F. Wang, S. Wang, X. Jiang, and J. T. Dudley. Deep learning for healthcare: review, opportunities and challenges. *Briefings in Bioinformatics*, 19(6):1236–1246, 05 2017.

[99] M. Mitchell, J. Aguilar, T. Wilson, and B. Van Durme. Open domain targeted sentiment. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 1643–1654, Seattle, Washington, USA, Oct. 2013. Association for Computational Linguistics.

[100] A. Mukherjee and B. Liu. 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, Jeju Island, Korea, July 2012. Association for Computational Linguistics.

[101] R. M. Neal. *Bayesian Learning for Neural Networks*. Springer-Verlag, Berlin, Heidelberg, 1996.

[102] J. Ni, J. Li, and J. McAuley. Justifying recommendations using distantly-labeled reviews and fine-grained aspects. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 188–197, Hong Kong, China, Nov. 2019. Association for Computational Linguistics.

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

[104] B. Pang, L. Lee, and S. Vaithyanathan. Thumbs up? Sentiment classification using machine learning techniques. In *Proceedings of the 2002 Conference on Empirical Methods in Natural Language Processing (EMNLP 2002)*, pages 79–86. Association for Computational Linguistics, July 2002.

[105] A. Paszke, S. Gross, S. Chintala, G. Chanan, E. Yang, Z. DeVito, Z. Lin, A. Desmaison, L. Antiga, and A. Lerer. Automatic differentiation in PyTorch. In *NeurIPS Autodiff Workshop*, 2017.

[106] J. Pennington, R. Socher, and C. Manning. GloVe: Global vectors for word representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural*

Language Processing (EMNLP), pages 1532–1543, Doha, Qatar, Oct. 2014. Association for Computational Linguistics.

[107] M. E. Peters, M. Neumann, M. Iyyer, M. Gardner, C. Clark, K. Lee, and L. Zettlemoyer. Deep contextualized word representations. 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 2227–2237, New Orleans, Louisiana, June 2018. Association for Computational Linguistics.

[108] M. Pontiki, D. Galanis, H. Papageorgiou, I. Androutsopoulos, S. Manandhar, M. AL-Smadi, M. Al-Ayyoub, Y. Zhao, B. Qin, O. De Clercq, V. Hoste, M. Apidianaki, X. Tannier, N. Loukachevitch, E. Kotelnikov, N. Bel, S. M. Jiménez-Zafra, and G. Eryiğit. SemEval-2016 task 5: Aspect based sentiment analysis. In *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016)*, pages 19–30, San Diego, California, June 2016. Association for Computational Linguistics.

[109] M. Pontiki, D. Galanis, H. Papageorgiou, S. Manandhar, and I. Androutsopoulos. SemEval-2015 task 12: Aspect based sentiment analysis. In *Proceedings of the 9th International Workshop on Semantic Evaluation (SemEval 2015)*, pages 486–495, Denver, Colorado, June 2015. Association for Computational Linguistics.

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

[111] G. Qiu, B. Liu, J. Bu, and C. Chen. Opinion word expansion and target extraction through double propagation. *Computational Linguistics*, 37(1):9–27, 2011.

[112] X. Quan, C. Kit, Y. Ge, and S. J. Pan. Short and sparse text topic modeling via self-aggregation. In *Proceedings of the 24th International Conference on Artificial Intelligence*, IJCAI'15, page 2270–2276. AAAI Press, 2015.

[113] V. Rakesh, W. Ding, A. Ahuja, N. Rao, Y. Sun, and C. K. Reddy. A sparse topic model for extracting aspect-specific summaries from online reviews. In *Proceedings of the 2018 World Wide Web Conference*, WWW '18, page 1573–1582, Republic and Canton of Geneva, CHE, 2018. International World Wide Web Conferences Steering Committee.

[114] V. K. Rangarajan Sridhar. Unsupervised topic modeling for short texts using distributed representations of words. In *Proceedings of the 1st Workshop on Vector Space Modeling for Natural Language Processing*, pages 192–200, Denver, Colorado, June 2015. Association for Computational Linguistics.

[115] C. E. Rasmussen. Gaussian processes in machine learning. In *Advanced Lectures on Machine Learning*, pages 63–71. Springer Berlin Heidelberg, 2004.

[116] S. Rendle. Factorization machines. In *2010 IEEE International Conference on Data Mining*, pages 995–1000, 2010.

[117] M. T. Ribeiro, S. Singh, and C. Guestrin. Model-agnostic interpretability of machine learning. *arXiv preprint arXiv:1606.05386*, 2016.

[118] A. Rietzler, S. Stabinger, P. Opitz, and S. Engl. Adapt or get left behind: Domain adaptation through BERT language model finetuning for aspect-target sentiment classification. In *Proceedings of the 12th Language Resources and Evaluation Conference*, pages 4933–4941, Marseille, France, May 2020. European Language Resources Association.

[119] V. Sanh, L. Debut, J. Chaumond, and T. Wolf. DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter. *arXiv preprint arXiv:1910.01108*, 2019.

[120] J. B. Schafer, D. Frankowski, J. Herlocker, and S. Sen. Collaborative filtering recommender systems. In *The Adaptive Web*, pages 291–324. Springer, 2007.

[121] S. Serrano and N. A. Smith. Is attention interpretable? In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 2931–2951, Florence, Italy, July 2019. Association for Computational Linguistics.

[122] T. Shi, K. Kang, J. Choo, and C. K. Reddy. Short-text topic modeling via non-negative matrix factorization enriched with local word-context correlations. In *Proceedings of the 2018 World Wide Web Conference*, WWW '18, page 1105–1114, Republic and Canton of Geneva, CHE, 2018. International World Wide Web Conferences Steering Committee.

[123] T. Shi, Y. Keneshloo, N. Ramakrishnan, and C. K. Reddy. Neural abstractive text summarization with sequence-to-sequence models. *ACM Transactions on Data Science*, 2(1):1–37, 2021.

[124] T. Shi, L. Li, P. Wang, and C. K. Reddy. A simple and effective self-supervised contrastive learning framework for aspect detection. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 13815–13824, 2021.

[125] T. Shi, V. Rakesh, S. Wang, and C. K. Reddy. Document-level multi-aspect sentiment classification for online reviews of medical experts. In *Proceedings of the 28th ACM International Conference on Information and Knowledge Management*, CIKM '19, page 2723–2731, New York, NY, USA, 2019. Association for Computing Machinery.

[126] T. Shi, P. Wang, and C. K. Reddy. Leafnats: An open-source toolkit and live demo system for neural abstractive text summarization. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics (Demonstrations)*, pages 66–71, 2019.

[127] T. Shi, P. Wang, and C. K. Reddy. An interpretable and uncertainty aware multi-task framework for multi-aspect sentiment analysis. *arXiv preprint arXiv:2009.09112*, 2020.

[128] T. Shi, X. Zhang, P. Wang, and C. K. Reddy. Corpus-level and concept-based explanations for interpretable document classification. *arXiv preprint arXiv:2004.13003*, 2020.

[129] B. Shickel, P. J. Tighe, A. Bihorac, and P. Rashidi. Deep EHR: A survey of recent advances in deep learning techniques for electronic health record (EHR) analysis. *IEEE Journal of Biomedical and Health Informatics*, 22(5):1589–1604, 2018.

[130] A. J. Smola and B. Schölkopf. A tutorial on support vector regression. *Statistics and Computing*, 14(3):199–222, Aug 2004.

[131] B. Sriram, D. Fuhry, E. Demir, H. Ferhatosmanoglu, and M. Demirbas. Short text classification in Twitter to improve information filtering. In *Proceedings of the 33rd International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '10, page 841–842, New York, NY, USA, 2010. Association for Computing Machinery.

[132] A. Srivastava and C. Sutton. Autoencoding variational inference for topic models. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*. OpenReview.net, 2017.

[133] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov. Dropout: A simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research*, 15(56):1929–1958, 2014.

[134] R. K. Srivastava, K. Greff, and J. Schmidhuber. Highway networks. *arXiv preprint arXiv:1505.00387*, 2015.

[135] H. Strobelt, S. Gehrmann, M. Behrisch, A. Perer, H. Pfister, and A. M. Rush. Seq2seq-vis: A visual debugging tool for sequence-to-sequence models. *IEEE Transactions on Visualization and Computer Graphics*, 25(1):353–363, 2019.

[136] Y. Suhara, X. Wang, S. Angelidis, and W.-C. Tan. OpinionDigest: A simple framework for opinion summarization. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 5789–5798, Online, July 2020. Association for Computational Linguistics.

[137] M. Sundararajan, A. Taly, and Q. Yan. Axiomatic attribution for deep networks. In *Proceedings of the 34th International Conference on Machine Learning - Volume 70*, ICML'17, page 3319–3328. JMLR.org, 2017.

[138] D. Tang, B. Qin, and T. Liu. 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, Sept. 2015. Association for Computational Linguistics.

[139] D. Tang, B. Qin, and T. Liu. Aspect level sentiment classification with deep memory network. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 214–224, Austin, Texas, Nov. 2016. Association for Computational Linguistics.

[140] E. Tjoa and C. Guan. A survey on explainable artificial intelligence (XAI): Toward medical XAI. *IEEE Transactions on Neural Networks and Learning Systems*, PP, October 2020.

[141] S. Tulkens and A. van Cranenburgh. Embarrassingly simple unsupervised aspect extraction. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 3182–3187, Online, July 2020. Association for Computational Linguistics.

[142] P. D. Turney. 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. Association for Computational Linguistics, 2002.

[143] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. u. Kaiser, and I. Polosukhin. Attention is all you need. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017.

[144] J. Vig. A multiscale visualization of attention in the transformer model. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 37–42, Florence, Italy, July 2019. Association for Computational Linguistics.

[145] O. Vinyals, L. u. Kaiser, T. Koo, S. Petrov, I. Sutskever, and G. Hinton. Grammar as a foreign language. In C. Cortes, N. Lawrence, D. Lee, M. Sugiyama, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 28. Curran Associates, Inc., 2015.

[146] B. Wang and M. Liu. Deep learning for aspect-based sentiment analysis. In *cs224d*. Stanford University report, https://cs224d.stanford.edu/reports/WangBo.pdf, 2015.

[147] H. Wang, Y. Lu, and C. Zhai. Latent aspect rating analysis on review text data: A rating regression approach. In *Proceedings of the 16th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '10, page 783–792, New York, NY, USA, 2010. Association for Computing Machinery.

[148] L. Wang, K. Liu, Z. Cao, J. Zhao, and G. de Melo. Sentiment-aspect extraction based on restricted Boltzmann machines. 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 616–625, Beijing, China, July 2015. Association for Computational Linguistics.

[149] W. Wang, S. J. Pan, D. Dahlmeier, and X. Xiao. 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, Nov. 2016. Association for Computational Linguistics.

[150] W. Wang, S. J. Pan, D. Dahlmeier, and X. Xiao. Coupled multi-layer attentions for co-extraction of aspect and opinion terms. In *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence*, AAAI'17, page 3316–3322. AAAI Press, 2017.

[151] Y. Wang, M. Huang, X. Zhu, and L. Zhao. Attention-based LSTM for aspect-level sentiment classification. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 606–615, Austin, Texas, Nov. 2016. Association for Computational Linguistics.

[152] Z. Wang and H. Wang. Understanding short texts. In *the Association for Computational Linguistics (ACL) (Tutorial)*, August 2016.

[153] T. Wolf, L. Debut, V. Sanh, J. Chaumond, C. Delangue, A. Moi, P. Cistac, T. Rault, R. Louf, M. Funtowicz, J. Davison, S. Shleifer, P. von Platen, C. Ma, Y. Jernite, J. Plu, C. Xu, T. Le Scao, S. Gugger, M. Drame, Q. Lhoest, and A. Rush. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online, Oct. 2020. Association for Computational Linguistics.

[154] W. Wu, Y. Su, X. Chen, S. Zhao, I. King, M. R. Lyu, and Y.-W. Tai. Towards global explanations of convolutional neural networks with concept attribution. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020.

[155] K. Xu, J. Ba, R. Kiros, K. Cho, A. Courville, R. Salakhudinov, R. Zemel, and Y. Bengio. Show, Attend and Tell: Neural image caption generation with visual attention. In F. Bach and D. Blei, editors, *Proceedings of the 32nd International Conference on Machine Learning*, volume 37 of *Proceedings of Machine Learning Research*, pages 2048–2057, Lille, France, 07–09 Jul 2015. PMLR.

[156] G. Xun, V. Gopalakrishnan, F. Ma, Y. Li, J. Gao, and A. Zhang. Topic discovery for short texts using word embeddings. In *2016 IEEE 16th International Conference on Data Mining (ICDM)*, pages 1299–1304, 2016.

[157] X. Yan, J. Guo, Y. Lan, and X. Cheng. A Biterm topic model for short texts. In *Proceedings of the 22nd International Conference on World Wide Web*, WWW '13, page 1445–1456, New York, NY, USA, 2013. Association for Computing Machinery.

[158] X. Yan, J. Guo, S. Liu, X. Cheng, and Y. Wang. Learning topics in short texts by non-negative matrix factorization on term correlation matrix. In *Proceedings of the 2013 SIAM International Conference on Data Mining*, pages 749–757. SIAM, 2013.

[159] B. Yang and C. Cardie. Extracting opinion expressions with semi-Markov conditional random fields. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, pages 1335–1345, Jeju Island, Korea, July 2012. Association for Computational Linguistics.

[160] Z. Yang, Z. Dai, Y. Yang, J. Carbonell, R. R. Salakhutdinov, and Q. V. Le. XLNet: Generalized autoregressive pretraining for language understanding. *Advances in neural information processing systems*, 32, 2019.

[161] Z. Yang, D. Yang, C. Dyer, X. He, A. Smola, and E. Hovy. 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, June 2016. Association for Computational Linguistics.

[162] C.-K. Yeh, B. Kim, S. Arik, C.-L. Li, T. Pfister, and P. Ravikumar. On completeness-aware concept-based explanations in deep neural networks. In H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems*, volume 33, pages 20554–20565. Curran Associates, Inc., 2020.

[163] Y. Yin, Y. Song, and M. Zhang. Document-level multi-aspect sentiment classification as machine comprehension. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 2044–2054, Copenhagen, Denmark, Sept. 2017. Association for Computational Linguistics.

[164] T. Young, D. Hazarika, S. Poria, and E. Cambria. Recent trends in deep learning based natural language processing. *arXiv preprint arXiv:1708.02709*, 2017.

[165] M. N. Zaeem and M. Komeili. Cause and effect: Concept-based explanation of neural networks. *arXiv preprint arXiv:2105.07033*, 2021.

[166] H. Zaragoza and F. d'Alché Buc. Confidence measures for neural network classifiers. In *7th Conference on Information Processing and Management of Uncertainty in Knowledge-Based Systems*, Paris, France, July 1998.

[167] Z. Zeng, W. Zhou, X. Liu, and Y. Song. A variational approach to weakly supervised document-level multi-aspect sentiment classification. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics:*

Human Language Technologies, Volume 1 (Long and Short Papers), pages 386–396, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics.

[168] L. Zhang, S. Wang, and B. Liu. Deep learning for sentiment analysis: A survey. *WIREs Data Mining and Knowledge Discovery*, 8(4):e1253, 2018.

[169] Q. Zhang and C. Shi. An attentive memory network integrated with aspect dependency for document-level multi-aspect sentiment classification. In W. S. Lee and T. Suzuki, editors, *Proceedings of The Eleventh Asian Conference on Machine Learning*, volume 101 of *Proceedings of Machine Learning Research*, pages 425–440, Nagoya, Japan, 17–19 Nov 2019. PMLR.

[170] X. Zhang, F. Chen, C.-T. Lu, and N. Ramakrishnan. Mitigating uncertainty in document classification. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 3126–3136, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics.

[171] X. Zhang, Z. Qiao, A. Ahuja, W. Fan, E. A. Fox, and C. K. Reddy. Discovering product defects and solutions from online user generated contents. In *The World Wide Web Conference*, WWW '19, page 3441–3447, New York, NY, USA, 2019. Association for Computing Machinery.

[172] Y. Zhang and X. Chen. Explainable recommendation: A survey and new perspectives. *Foundations and Trends in Information Retrieval*, 14(1):1–101, 2020.

[173] W. X. Zhao, J. Jiang, J. Weng, J. He, E.-P. Lim, H. Yan, and X. Li. Comparing Twitter and traditional media using topic models. In P. Clough, C. Foley, C. Gurrin, G. J. F. Jones, W. Kraaij, H. Lee, and V. Mudoch, editors, *Advances in Information Retrieval*, pages 338–349, Berlin, Heidelberg, 2011. Springer Berlin Heidelberg.

[174] X. Zhao, J. Jiang, H. Yan, and X. Li. 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, Cambridge, MA, Oct. 2010. Association for Computational Linguistics.

[175] L. Zheng, V. Noroozi, and P. S. Yu. Joint deep modeling of users and items using reviews for recommendation. In *Proceedings of the Tenth ACM International Conference on Web Search and Data Mining*, WSDM '17, page 425–434, New York, NY, USA, 2017. Association for Computing Machinery.

[176] B. Zhou, Y. Sun, D. Bau, and A. Torralba. Interpretable basis decomposition for visual explanation. In *Proceedings of the European Conference on Computer Vision (ECCV)*, September 2018.

[177] H. Zhuang, F. Guo, C. Zhang, L. Liu, and J. Han. Joint aspect-sentiment analysis with minimal user guidance. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '20, page 1241–1250, New York, NY, USA, 2020. Association for Computing Machinery.

[178] A. Zubiaga and H. Ji. Harnessing web page directories for large-scale classification of tweets. In *Proceedings of the 22nd International Conference on World Wide Web*, WWW '13 Companion, page 225–226, New York, NY, USA, 2013. Association for Computing Machinery.

[179] Y. Zuo, J. Wu, H. Zhang, H. Lin, F. Wang, K. Xu, and H. Xiong. Topic modeling of short texts: A pseudo-document view. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '16, page 2105–2114, New York, NY, USA, 2016. Association for Computing Machinery.