EXPLOITING DOMAIN KNOWLEDGE TO ENHANCE OPINION MINING USING A HYBRID SEMANTIC KNOWLEDGEBASE-MACHINE LEARNING APPROACH

A thesis by Rowida Abdulmajed Alfrjani submitted in partial fulfilment of the requirements of Nottingham Trent University for the degree of Doctor of Philosophy

Department of Computer Science & Technology
Nottingham Trent University

March 2018

Abstract

With the fast growth of World Wide Web 2.0, a great number of opinions about a variety of products have been published on blogs, forums, and social networks. Online opinions play an important role in supporting consumers make decisions about purchasing products or services. In addition, customer reviews allow companies to understand the strengths and limitations of their products and services, which aids in improving their marketing campaigns. The challenge is that online opinions are predominantly expressed in natural language text, and hence opinion mining tools are required to facilitate the effective analysis of opinions from the unstructured text and to allow for qualitative information extraction. This research presents a Hybrid Semantic Knowledgebase-Machine Learning approach for mining opinions at the domain feature level and classifying the overall opinion on a multi-point scale. The proposed approach benefits from the advantages of deploying a novel Semantic Knowledgebase approach to analyse a collection of reviews at the domain feature level and produce a set of structured information that associates the expressed opinions with specific domain features. The information in the knowledgebase is further supplemented with domain-relevant facts sourced from public Semantic datasets, and the enriched semantically-tagged information is then used to infer valuable semantic information about the domain as well as the expressed opinions on the domain features by summarising the overall opinions about the domain across multiple reviews, and by averaging the overall opinions about other cinematic features. The retrieved semantic information represents a valuable resource for training a Machine Learning classifier to predict the numerical rating of each review. Experimental evaluation revealed that the proposed Hybrid Semantic Knowledgebase-Machine Learning approach improved the precision and recall of the extracted domain features, and hence proved suitable for producing an enriched dataset of semantic features that resulted in higher classification accuracy.

Copyright Statement

Declaration

I declare that the presented work in this thesis is the original work of the author except where explicitly stated otherwise in the text. I declare that this thesis as well as the materials contained in the thesis have not been used in any other submission for an academic award.

Parts of the presented work in the thesis have been published in:

Paper 1: “A New Approach to Ontology-Based Semantic Modelling for Opinion Mining”, published in “2016 UKSim-AMSS 18th International Conference on Computer Modelling and Simulation (UKSim)”, Publisher: IEEE

Paper 2: “Exploiting domain knowledge and public linked data to extract opinions from reviews”, published in “2017 2nd International Conference on Knowledge Engineering and Applications (ICKEA)”, Publisher: IEEE

Paper 3: “A Hybrid Semantic Knowledgebase-Machine Learning Approach for Opinion Mining”, submitted to “Elsevier Data and Knowledge Engineering Journal” on 26/1/2018. Current status: Invited for Revision

Acknowledgement

First and foremost, Praise is to Allah, the Almighty for having guided me at every stage of my life and for giving me the strength and patience through all years of my PhD research.

I am grateful to the Libyan Government for the scholarship which enabled me to undertake a PhD research at the Nottingham Trent University.

I would like to express my sincere gratefulness and honour to my supervisor, Dr. Taha Osman, for his enriched knowledge, support and valuable guidance during all stages of my PhD research. Without his richness in experience, encouragement and continuous follow-up; this research would never have been.

I would like also to express my deep gratitude to my second supervisor, Dr. Georgina Cosma who was abundantly helpful and offered invaluable assistance, valuable feedback and suggestions; always having time to listen, always prepared to help and always open to discussion.

Many thanks to all my friends and relatives for their constant support and encouragement.

Special and lovely thanks to my parents, sisters and brothers for their encouragement and support even I am abroad far away from them. Thank you for giving me strength to reach for the stars and chase my dreams.

Last but not least, I owe gratefulness thanks to a very special person, my husband, Abubaker Brgeliel for his continued and unfailing love, support and understanding during all stages of my PhD research that made the completion of thesis possible. He was always around at times I thought that it is impossible to continue. I appreciate my baby boy Ayoub, my little girls Yakin, Ghofran and Rahma for abiding my ignorance and the patience they showed during my thesis writing. Words would never say how grateful I am to all of you.

Dedication

To my dearly beloved mum and dad,

To my wonderful husband,

To my precious kids,

I dedicate this work.

Table of Contents

List of Tables

List of Figures

Chapter 1

1 Introduction

Opinions, often in the form of reviews, are increasingly being published on websites, blogs and social media outlets. Consumers often consult the opinion of others when considering whether to make a purchase. For instance, it is common to seek out the opinion of friends, favourite bloggers and reviewers when making a decision about purchasing a product, voting for a political candidate or choosing a movie. Analysing online customer reviews that have been published on E-commerce websites enable organisations to track the strengths and limitations of their products/services as a technique for improving products and services. Opinion mining is also increasingly being used in numerous applications such as analysing views on social media during presidential campaigns (Karami, Bennett and He 2018). Organisations invest considerable resources to collate and analyse online material in order to identify the underlying user trends regarding consumer sentiments, and use such information to improve their products and services and to shape their production strategies and marketing campaigns (Ibrahim, Wang and Bourne 2017). Google Analytics, Review Seer and Opinion Observer are examples of applications that perform opinion mining tasks on online contents such as: determining the overall polarity of the content, providing a structured summary of online reviews and comments, and providing a search engine for users to retrieve products based on their features as well as the sentiment polarity of the features (Vinodhini and Chandrasekaran 2012, Chakraborty and Pagolu 2014). The challenge is that online opinions are predominantly expressed in natural language text, images, videos, etc.; and hence opinion mining tools are required to facilitate the effective extraction and analysis of opinions from unstructured text, images, videos, etc. Such tools often adopt algorithms from the Natural Language Processing, Information Retrieval and Machine Learning disciplines.

1.1 Research Motivation

Opinion mining is commonly implemented by extracting contents for a specific domain (e.g. movie, music, car, hotel, cellular phone, restaurant and product) and performing opinion mining at various levels of text granularity: document, sentence or domain feature level. At document and sentence level, opinion mining aims to classify the overall sentiment orientation that is expressed in a document (Pang, Lee and Vaithyanathan 2002, Pang and Lee 2005) or a sentence (Pang and Lee 2004, Meena and Prabhakar 2007, Yu and Hatzivassiloglou 2003).

Opinion mining at the domain feature level is considered to be a challenging task because it requires deep understanding of the sentence structure and knowledge of the problem domain (e.g. movie reviews) in order to correctly classify domain features based on their polarity (Hu and Liu 2004, Somprasertsri and Lalitrojwong 2010). Particularly challenging is the extraction of the domain feature mentions (e.g. actress, show, script, story) from the reviews and associating each domain feature with its corresponding sentiment to determine its polarity score (e.g. the beauty of the script +1; Bulletproof Heart is not an excellent movie -1; The great Matt Craven will probably be forever remembered +1). Opinion mining at domain feature level can be further considered for enhancing the opinion classification task via summing or averaging the sentiment polarity score of each extracted domain feature to determine the numerical rating of the review (e.g. 4,3,2,1 and 0 for very positive, positive, neutral, negative, and very negative respectively).

Opinion mining research at domain feature level employs different approaches such as Machine Learning, Association Rule Mining and Semantic Knowledgebase approaches to primarily improve the outcome of the domain feature extraction task, which consequently enhances the performance of opinion classification task.

Machine Learning Approaches deliver significant results for domain feature extraction task using training datasets that have been manually annotated by a human expert. However, this can be an extremely time-consuming task as the required size of the training dataset (i.e. cover the domain feature) should be sufficiently large to bootstrap the learning algorithms, whereas, the automatic preparation cannot be accurate. Association Rule Mining approaches for domain feature extraction tasks do not require manual or automatic preparation of dataset as they primarily rely on Natural Language Processing techniques to identify frequent nouns and noun phrases to be domain features. However, the extracted domain features tend to be frequent domain features, whereas infrequent domain features are ignored, which can result in a reduced recall rate. In addition, some of the extracted nouns and noun phrases may not be domain features even if these occur more frequently in the textual contents, and this can affect the precision of the domain feature extraction task. The Semantic Knowledgebase approaches are based on utilising domain knowledge to extract domain features from textual contents, which contains a conceptualized knowledge background of the domain. Such domain knowledge can be utilised to extract the frequent and infrequent domain features to improve the performance of domain feature extraction task. The Domain Knowledge captures the key concepts and relations of the problem domain's environment, which is then populated with entities and facts/events that subscribe to the modelled concepts and relations (Dalvi, et al. 2015).

The World Wide Web Consortium (W3C) defines the Semantic Web[1] as "the Semantic Web technologies provide a common framework that allows data to be shared and reused across application, enterprise, and community boundaries. It is a collaborative effort led by W3C with participation from a large number of researchers and industrial partners. It is based on the Resource Description Framework, which integrates a variety of applications using XML for syntax and URIs for naming". Semantic Web technologies are considered ideal for modelling domain knowledge as they organise knowledge in a formalised semantic knowledgebase that provides efficient support for linking and sharing data between resources, and presenting data in a way that computer machines can process. In addition, the formalised semantic knowledgebase is capable of presenting the domain knowledge in a structured and consistent manner which facilitates the qualitative interpretation of domain specific contents in a way that people can understand. Moreover, Semantic Web technologies provide support for populating the semantically structured domain knowledgebase with relevant ground facts extracted from public-sourced Linked Open Data resources.

Semantic Knowledgebase approaches have been deployed for domain feature extraction with promising success (Ali, Kim and Kim 2015). However, the success of

[1] https://www.w3.org/2001/sw/

these techniques largely depends on the domain knowledge coverage, and the conducted investigation into the state-of-the-art approaches showed that the domain knowledge coverage is often limited. In addition, there is shortcoming in investigating the use of Open Linked Data resources for enhancing the performance of domain feature extraction task. Moreover, domain features that are extracted from a textual content might not have any subjective opinions about them as users maybe describe factual information about the extracted domain features as in “The Addiction movie is an American movie”. Most of the conducted related work have used syntactic parsing techniques (i.e. identify both descriptive and subjective phrases) without considering the utilisation of the domain knowledge to eliminate such non-opinionated domain features to enhance the domain feature-sentiment association task.

Hence, there is an opportunity of investigating whether exploiting the knowledge of the problem domain alongside with Linked Open Data resources can improve the performance of the domain feature extraction task, which consequently enhances the performance of opinion classification task.

Opinion classification is the process of classifying opinions into a binary classification (i.e. whether it is a positive or negative) or a multi-point scale (i.e. classify the polarity of the content at fine-grained level) such as very negative, negative, neutral, positive and very positive (Pang and Lee 2005).

The problem of classifying opinions using a multi-point scale (also referred to as the rating inference problem) has been an interesting research area in the recent years. Machine Learning approaches have been commonly applied for the process of opinion classification and are known to deliver outstanding performance, especially when they are trained using an effective dataset of features that have been manually annotated by a human expert who tends to enhance the annotation process with domain background knowledge. However, this can be an extremely time-consuming task as the required size of the training dataset should be sufficiently large to bootstrap the learning algorithms.

The Semantic Knowledgebase approach uses a knowledgebase that represents a shared understanding of the domain of interest, hence, the Semantic Knowledgebase approach can be used to enrich a dataset with semantic features, which can improve the performance of opinion classification task. However, the reported efforts as in

(Polpinij and Ghose 2008, Sulthana and Subburaj 2016) have mainly focused on binary classification tasks, i.e. identifying whether the content has a positive or negative opinion. Whereas there appear to be no studies that investigate the use of Semantic Knowledgebase approaches to produce dataset of semantic features that are then used to build a Machine Learning classifier to classify the opinions on multi-point rating scale. Moreover, the challenge remains on developing approaches for extracting semantic features from the constructed knowledgebase and putting these into a suitable format for training a Machine Learning classifier.

The work presented in this thesis is also motivated by the finding that Semantic Knowledgebase approaches are an attractive, but yet under-researched, approaches for Opinion Classification applications. Hence, there is an opportunity of investigating whether combining a Semantic Knowledgebase approach with a Machine Learning approach (i.e. adding additional semantic features to a dataset of statistical features and use that to build a classifier) can result in higher classification accuracy for multi-point rating scale compared to using Machine Learning approaches alone.

1.2 Research Aim and Questions

The aim of this research is to develop a Hybrid Semantic Knowledgebase-Machine Learning approach to enhance the performance of opinion mining at domain feature level. In particular, improving the main tasks of opinion mining that include extracting domain features, associating them with their corresponding sentiments and opinion classification i.e. solving the rating inference problem on a multi-point scale.

The following research questions are established according to the aim of this research:

RQ1. How can the semantic modelling of the domain knowledge further contribute to improving the opinion mining at domain feature level, in particular to the domain feature extraction and opinion classification tasks?

RQ2. Can the domain knowledge improve the precision and recall of the feature extraction task?

RQ3. How can the semantically structured public datasets be exploited to improve the performance of domain feature extraction task?

RQ4. Given the fact that the target domain feature is presented by a single name or pronoun (i.e. termed non-explicit domain features), how can the semantically constructed knowledgebase be utilised with co-reference resolution to extract non-explicit domain features to further improve the domain feature extraction task?

RQ5. Can the domain's sentiment lexicon contribute to improve the domain feature-sentiment association task?

RQ6. Is the aggregation of the domain features' sentiment polarities based on Semantic Knowledgebase approach sufficient for the accurate classification of the review opinion?

RQ7. How can we use Semantic Knowledgebase approach to improve the quality of training features that are then used to build a Machine Learning classifier in order to improve the accuracy of opinion classification on a multi-point scale?

1.3 Thesis Contributions

The contribution of the proposed Hybrid Semantic Knowledgebase-Machine Learning approach can be summarised as follows:

- A new Domain Feature Extraction algorithm that improves the precision and recall of the extracted domain features. The Domain Feature Extraction algorithm utilises a comprehensive knowledge of the chosen domain (key concepts and their synonyms and ground facts) and public Linked Open Data sources such as DBpedia and Internet Movie Database.
- A novel Domain Feature-Sentiment Association algorithm that reduces false positive opinions (i.e. the domain feature-sentiment pairs) that objectively describe factual information using a generated sentiment lexicon for each domain feature.
- A new Opinion Classification algorithm that delivers enhanced opinion classification on a multi-point scale. The Opinion Classification algorithm generates an enriched set of semantic data from a semantically structured semantic knowledgebase, merges it with a statistical dataset, and then uses the combined data as input into Machine Learning algorithms. This is the

first study that presents an approach combining semantic and statistical data for classifying opinions on a multi-point scale.

- A novel comprehensive methodology for exploiting domain knowledge both in extracting opinion-related features and their associated sentiments using Semantic Knowledgebase approach, as well as exploiting the semantic domain knowledgebase to enrich the training dataset of the Machine Learning opinion classifiers and subsequently improve their accuracy.

1.4 Thesis Organisation

The remainder of the thesis is organised as follows:

Chapter 2 presents Review of Opinion Mining approaches.

Chapter 3 presents the architecture framework of the proposed Hybrid Semantic Knowledgebase-Machine Learning approach for opinion mining, and addresses the first research question RQ1 via introducing the methodology for the semantic modelling of the problem domain knowledge.

Chapter 4 addresses three research questions RQ2, RQ3 and RQ4 via introducing details of the conducted Domain Feature Extraction phase together with the experimental evaluation as well as the state-of-the-art related work on domain feature extraction task.

Chapter 5 addresses research questions RQ5 and RQ6 via introducing details of the conducted Domain Feature-Sentiment Association phase together with the experimental evaluation as well as the state-of-the-art related work on domain feature-sentiment association task.

Chapter 6 addresses the final research question RQ7 via introducing details of the conducted Multi-point Opinion Classification phase alongside the experimental evaluation and the related works on opinion classification on a multi-point scale.

Chapter 7 illustrates dynamics of the population and interrogation of the developed domain knowledge.

Chapter 8 presents a conclusion of the conducted work in this research and the remaining future work.

Chapter 2

2 Review of Opinion Mining Approaches

Opinion mining analysis has been used for various aspects in our daily life such as in marketing, brand monitoring and election results. In addition, opinion mining has been conducted on various domains, such as the product domain (Deng, Luo and Yu 2014, Cosma and Acampora 2016, Qiao, et al. 2017); and the tourism domain (Pang, Lee and Vaithyanathan 2002, Bhatnagar, Goyal and Hussain 2018). Studies have also been carried out on movie domain (Lunardi, et al. 2016, Manek, et al. 2017, Chakraborty, et al. 2018); hotel domain (Hu, Chen and Chou 2017) ; and in the music domain (Dalvi, et al. 2015).

This chapter discusses related literature on existing approaches to opinion mining process, which can be broadly categorised into Lexicon-Based, Association Rule Mining, Machine Learning, and Semantic Knowledgebase approaches.

2.1 Lexicon-Based Approaches

Lexicon-Based approaches for opinion mining are based on utilising dictionaries which contain sentiment terms and phrases along with the orientations and strength of the terms and phrases. Lexicon-Based approaches compute the overall polarity of the content based on sentiment term orientations and strength with respect to any associated modifier terms and negations (Greene 2007). Lexicon-Based approaches have demonstrated a successful performance for mining various domains' contents (Liu 2012). Some studies incorporate part of speech tagging for sentiment terms within the utilised lexicon in order to enhance determining the score of disambiguation terms (Gezici, et al. 2013). Opinion mining has performed at document level on Spanish contents via summing the semantic orientation of phrases extracted from reviews (Cruz, et al. 2008). The semantic orientation of reviews was calculated based on comparing their similarity between positive and negative adjectives that were obtained from sentiment lexicon. The authors in (Palanisamy, Yadav and Elchuri 2013) developed a Lexicon-Based approach for classifying tweets as positive or negative, where sentiments were discovered using a lexicon that was constructed from the Serendio taxonomy. The Serendio taxonomy contains positive and negative terms, stop terms and phrases. The authors pre-processed the contents by applying stemming and normalization, and then they identified emoticons and hashtag terms. Thereafter, the contents were classified based on the contextual sentiment orientation of the terms. The authors in (Taboada, et al. 2011) introduced a Semantic Orientation CALculator (SO-CAL) to extract sentiments from on-line contents and perform opinion mining at document level via utilisation of lexicon that contains annotated terms (adjectives, adverbs, nouns and verbs), as well as the semantic orientation for each term. In addition, intensification and negation terms were used that help to modify the polarity of each term. The SO-CAL system was demonstrated a consistent performance across various domains as it was aimed to be consistent and reliable via using the Mechanical Turk. The authors in (Mumtaz and Ahuja 2016) classified the online tweets as positive, neutral or negative using a lexicon that contains positive, negative and negation terms in which the polarity value of the tweet was calculated based on summing the score of each identified positive and negative terms and also by shifting the score of the terms that associated with a negation term. Tweets with a polarity value greater than zero were classified as positive; tweets with a polarity value which is less than zero were classified as negative; and tweets with a polarity value equal to zero were classified as neutral. The authors in (Muhammad, Wiratunga and Lothian 2016) used the Lexicon-Based approach to classify the contextual polarity of the content at local and global levels. The authors in (Krishnan, Elayidom and Santhanakrishnan 2017) have used a Lexicon-Based approach to analyse customers' reviews about mobile phones that are published on Twitter and measure the popularity of the mobile phones based on user's opinion on deciding whether buying the product or not. The authors in (Agarwal and Toshniwal 2018) have used the Lexicon-Based approach for calculating the sentiment of sports' event fans over the time and establishing the relationship between the fans sentiments and players performance.

Lexicon-Based techniques work on an assumption that the collective polarity of a document or sentence is the sum of polarities of the individual words or phrases. Creating such words lists is often easier than labelling instances as less resources are required and no require for labelled datasets. However, Lexicon-Based approaches demand powerful linguistic resources which is not always available. In addition, the major limitation of Lexicon-Based approach is incorrect sentiment scoring of opinion words by the existing lexicons, such as SentiWordNet as they may assign incorrect scores to most of the domain specific words. In this research, sentiment lexicons will be created for the problem domain to contain the sentiment polarity for each domain feature. The sentiment lexicons will be used only to extract sentiments from the text and assign to them their polarities and any adjacent shifters (negation or adverb) will be taken into account to moderate the sentiment's score accordingly.

2.2 Association Rule Mining Approaches

Association Rule Mining approaches deal with the content as a bag of terms and perform opinion mining at the document level via aggregating the sentiment score of all the extracted terms from the content. An earlier study on opinion mining based on Association Rule Mining approach was published by the author in (Turney 2002), where rules miming were applied to extract two consecutive words from the contents that their POS tags match one of specified bigrams. After that, pointwise mutual information was used to calculate the polarity of the extracted bigrams and then averaging them to determine the overall polarity of the document. Adjective terms play an important role for classifying the sentiment of the content (Hatzivassiloglou and McKeown 1997, Kamps, et al. 2004), and using nouns and verbs in addition to adjectives can result in better determination of the sentiment orientation of the sentence (Riloff, Wiebe and Wilson 2003, Kim and Hovy 2004). Some researchers focused on using pre-identified seeds of positive and negative terms to calculate the value of point wise mutual information and determining the sentiment score of the extracted phrases (Turney and Littman 2003) or determining the sentiment score of the extracted terms (Baroni and Vegnaduzzo 2004); and then averaging the score to classify the sentiment orientation of the review. The work done by (Hu and Liu 2004) was based on generating a collection of sentiment terms from WordNet, which were then used to determine the polarity of the prevalent terms and to classify the polarity of sentences. The authors in (Taboada, et al. 2011) utilised adverb terms (such as very, quite, none, a little, somewhat) as well as the negation term “not” to adjust the determined polarity of the extracted terms. The researchers in (Vilares, Alonso and Gómez-Rodríguez 2015) used syntactic dependencies to improve the association process between sentiment words and their corresponding adverbs or negations as well as dealing with the conjunction term “but” in order to increase the success of determining the polarity of the sentiment words. The authors in (Shih, et al. 2018) used Association Rule Mining approach to classify the situation of patients whether they with dementia or not based on generated rules from patients’ details such as Gender, age, type of dementia, number of days in hospital, and hospital medical expenses. The authors in (Jia, et al. 2018) used Association Rule Mining approach for cross-domain sentiment classification via defining the strong association rules between domain-shared words and domain-specific words in the same domain.

Association Rule Mining is a technique for finding interesting relationships or patterns hidden in large datasets. Association Rule Mining approaches rely on using rules that their generation is completely dependent on finding Frequent Item sets .However, the efficiency and accuracy of the Association Rule Mining approach depends on the defined rules as due to the noisy nature of the input datasets, the defined rules can be non-interesting, huge number or non-accurate. According to the authors in (Shridhar and Parmar 2017) “The principle disadvantages of the Association Rule Mining are the accompanying: obtaining non intriguing tenets, huge number of found principles, low calculation execution”. In this research, to find the relationship between a domain feature and its expressed sentiment (e.g. good movie) a set of dependency pattern rules will be defined based on the syntactical structure of the content to identify patterns that contain both domain feature and sentiment, which will be then associated together; and for more accurate association, domain sentiment lexicons will be used to discard the identified patterns that contain descriptive opinions.

2.3 Machine Learning Approaches

Machine Learning approaches are an interesting subject area of computer science where classifiers predict the target output based on learning the behaviour of a large collection of contents. Machine Learning approaches deal with the contents as a bag of features in which a classifier such as Naïve Bayes, Support Vector Machine, Artificial Neural Networks, Maximum Entropy, etc. learn from the specified features the target class of the provided contents (Rebolledo, L'Huillier and Velásquez 2010). The class of the content can be a binary class [0 and 1] for negative and positive respectively, or can be a multi-class such as [0, 1, 2 and 3] for strong negative, negative, positive and strong positive respectively. To build the classifier a large collection of contents are required, which is commonly split into two groups. The first group is a "training dataset" and it is for training the classifier on differentiating the features of the contents, whereas the second group is a "testing dataset" and it is for evaluating the performance of the trained classifier (Sebastiani 2002, Li, et al. 2015). The majority of the literature on Machine Learning approaches for opinion mining at document and sentence level trained the adopted classifier on unigrams, bigrams or n-grams features of the contents where they found that unigrams features have resulted in better performance than bigrams or n-grams features (Taboada, et al. 2011). Part of speech tagging, frequent terms, infrequent terms and word position are other kind of features that were also used for training three classifiers Naïve Bayes, Maximum Entropy and Support Vector Machine on a combination of features (Pang, Lee and Vaithyanathan 2002). Their results demonstrated that Support Vector Machines performed better classification than the other classifiers. The authors in (Yu and Hatzivassiloglou 2003) used polarity, terms, part of speech, bigrams and trigrams features to build a Naïve Bayes classifier for classifying sentences as positive or negative. Others have used features for training a classifier to classify the content such as sentiment terms, sentiment phrases, sentiment shifters, rules and syntactic dependency of the expressed opinions (Joshi and Penstein-Rosé 2009, Liu 2012). The study by (Pak and Paroubek 2010) was based on training three classifiers Support Vector Machine, Naïve Bayes and Conditional Random Field on a dataset which contains labelled features of sad and happy emoticons that are used in social media applications such as Twitter. The obtained results showed that Naïve Bayes classified the polarity of the content better than the other classifiers via using emoticons features. The authors in (Davidov, Tsur and Rappoport 2010) used hashtags as a labelled feature in addition to the sad and happy emoticons features to build a K-Nearest Neighbours classifier to classify tweets as positive or negative. The researchers in (Martínez-

Cámara, Martín-Valdivia and Ureña-López 2011) evaluated Naïve Bayes and Support Vector Machine classifiers to classify Spanish texts. Their study stated that Support Vector Machine performed best. The work done by (Tang, Qin and Liu 2015) was based on using a different type of features to build the classifier for opinion mining at document level, which are low-dimensional, real-valued and dense. The authors in (Bespalov, et al. 2011) proposed a Deep Neural Network classifier based on n-grams features and a low-dimensional latent semantic space features in order to enhance the performance of opinion mining at document level. The author in (Asghar 2016) evaluated the performance of opinion mining using a combination of four types of extracted features (unigrams, bigrams, trigrams and latent semantic indexing) with four types of Machine Learning algorithms which are: Naïve Bayes, Perceptron Neural Networks, Logistic Regression and Linear Support Vector Classifier. The authors in (Shubha and Suresh 2017) proposed a Machine Learning Bayes Sentiment Classification method to classify the content at document level via training a probabilistic Bayes classifiers on related opinion words that were extracted from user review comments.

Machine Learning approaches have been commonly applied for the process of opinion miming and are known to deliver outstanding performance, especially when they are trained using an effective dataset of features that have been manually annotated by a human expert who tend to enhance the annotation process with domain background knowledge. However, this can be an extremely time-consuming task as the required size of the training dataset should be sufficiently large to bootstrap the learning algorithms. In this research, we will benefit of the knowledge of the problem domain to provide a deep understanding of the structure and knowledge of the content to produce an enrich dataset of semantic features; which will be used to build a Machine Learning classifier for classifying the overall opinion on a multi-point scale.

2.4 Semantic Knowledgebase Approach

Semantic Knowledgebase approach is a new approach that has been used lately for opinion mining at domain feature level, and it is based on utilising a knowledgebase that contains a conceptualised knowledge background of the domain to primarily extract domain features from the content and determine their polarity based on their corresponding sentiments. Domain Knowledge is knowledge about a domain's environment, i.e. key concepts and their synonyms and ground facts, as well as the relation between them (Dalvi, et al. 2015). Such domain knowledge can be modelled via a concept map, translated into a knowledgebase that is then populated with relevant information to improve the processes of opinion mining process (Alfrjani, Osman and Cosma 2016). The authors in (Zhao and Li 2009, Penalver-Martinez, et al. 2014, Agarwal, et al. 2015a) translated the knowledge background of a chosen domain into a knowledgebase, and utilised this knowledgebase to extract domain features from pre-processed contents. However, their approaches are different to that of the authors in (Zhao and Li 2009) who constructed a knowledgebase that contains only the domain's key concepts and their synonymous. Whereas, the authors in (Penalver-Martinez, et al. 2014) adopted a general domain knowledgebase, which contains some domain's key concepts and their synonymous and collected ground facts from Internet Movie Database resources. The researchers in (Agarwal, et al. 2015a) proposed an approach based on constructing a knowledgebase for a specific domain using some concepts from the top four levels of ConceptNet knowledgebase, and then extended the knowledgebase with synonyms from WordNet. The work done by (Zhou and Chaovalit 2008) was based on modelling the movie domain concepts and then developing it into a knowledgebase, which was then used to extract movie domain's key concepts from the content. The Semantic Knowledgebase approach has also be utilised to classify the polarity of whole documents based on different techniques such as summing or averaging the polarity of all extracted domain features. The authors in (Cambria, et al. 2010) utilised a common-sense reasoning with a combination with domain's key concepts to build the knowledgebase that was used primarily to extract domain features. Thereafter, the extracted domain features were used to classify the contents at document level. The work done in (Poria, et al. 2013) was similar to that of the authors in (Cambria, et al. 2010) except that the developed knowledgebase was expanded with additional information about emotions (e.g. happy, sad, anger, joy, surprise and disgust) that were extracted from WordNet-Affect resource. The study by (Miao, Li and Zeng 2010) was based on integrating the domain's knowledge with lexical and syntactic knowledge to classifying the content at document level. The authors in (El-Halees and Al-Asmar 2017) used the domain knowledge to identify the relevant features from Arabic reviews in order to classify Arabic user-generated reviews that have different features with different opinion strengths.

Semantic Knowledgebase approaches rely on using the knowledge of the problem domain and the success of these techniques largely depends on the domain knowledge coverage, and the conducted investigation into the state-of-the-art approaches showed that the domain knowledge coverage is often limited. In this research, the main objective is to utilise a comprehensive domain knowledgebase and populate it with domain's ground facts that are obtained from Linked Open Data resources in order to provide deep understanding of the free-textual contents, which is envisaged to improve the performance of opinion mining.

2.5 Chapter Summary

In this chapter, various opinion mining approaches have been reviewed such as Lexicon-Based, Association Rule Mining, Machine Learning and Semantic Knowledgebase approaches. Machine Learning approaches require labelled data for training a classifier, whereas, Lexicon-Based approaches do not require labelled datasets. However, Lexicon-Based approaches demand powerful linguistic resources which is not always available. Machine Learning approaches require sufficiently large size of labelled datasets which are used during the training process. Furthermore, Machine Learning classifiers which have been trained on predicting polarity in texts in one domain are not suitable for predicting sentiments in another domain. The advantage of Machine Learning classifiers is that once they are trained they can be applied to predict opinion from text without further human intervention. Association Rule Mining approaches rely on using rules and syntactic dependency. However, their efficiency and accuracy depend on the defining rules. Semantic Knowledgebase approaches rely on using the knowledge of the problem domain and the success of these techniques largely depends on the domain knowledge coverage. Performing opinion mining analysis by different approaches will produce different results and each approach has its own strengths and shortcoming. In this research, the proposed Hybrid Semantic Knowledgebase-Machine Learning approach combines the strengths of the other approaches to deliver an approval method for opinion mining within determining problem domains.

Chapter 3

3 Semantic Modelling of the Problem Domain Knowledge

Opinion mining is commonly implemented by extracting contents for a specific domain (e.g. movie, music, car, hotel, cellular phone, restaurant and product) and performing opinion mining at various levels of text granularity: document, sentence or domain feature level. At document and sentence level, opinion mining aims to classify the overall sentiment orientation that is expressed in a document (Pang, Lee and Vaithyanathan 2002, Pang and Lee 2005) or a sentence (Pang and Lee 2004, Meena and Prabhakar 2007, Yu and Hatzivassiloglou 2003). At domain feature level, opinion mining aims to discover the expressed sentiments on the domain and/or its features (Hu and Liu 2004, Somprasertsri and Lalitrojwong 2010).

Discovering what exactly people liked and disliked about the domain and/or its features cannot be obtained via applying opinion mining at document or sentence level, which can in turn affect the accuracy of the overall determined sentiment (i.e. opinion classification). For example, in a sentence about a specific movie, "although the story was not great, the acting was amazing" clearly there are two movie's features (i.e. story and acting) that their sentiment polarity are negative and positive respectively. Hence, using opinion mining at sentence or document level, the overall determined sentiment will be on the domain its self "movie", whereas, the sentence is positive about the movie's feature "acting", but it is negative about the movie's feature "story". Therefore, realizing the importance of determining the sentiment polarity that expressed on a domain feature can help in resulting better overall determined sentiment (Liu 2012).

We hypothesise that we can improve the accuracy of the Machine Learning opinion classifiers by bootstrapping them with a rich training dataset generated by knowledge-based extraction of opinions (i.e. domain features associated with their sentiments). We envisage that for a particular problem domain, the training dataset can be further enriched by relevant ground facts extracted from semantically structured public datasets.

Opinion mining at domain feature level is a challenging problem because it focuses on extracting domain features from textual reviews and associating them with their corresponding sentiments. Such a task requires deep understanding of the structure and knowledge of the content in order to correctly extract domain features and their relevant sentiments and then determine the polarity of each sentiment.

The required domain knowledge represents the domain's environment contains information such as key concepts and synonyms and ground facts, as well as the relation between them (Dalvi, et al. 2015). Information from a domain's semantic knowledgebase can be utilised to improve the performance of opinion mining process, in particular, the domain feature extraction task.

The modelling stage is the initial and critical stage for building a comprehensive framework that relies on knowledgebase modelling for opinion mining at domain feature level, which addresses our first research question RQ1 (*How can the semantic modelling of the domain knowledge improve the domain feature extraction and opinion classification tasks?*). Figure *3.1* illustrates the interface that interconnects the developed domain knowledgebase with the main phases of the proposed Hybrid Semantic Knowledgebase-Machine Learning approach: Domain Feature Extraction, Domain Feature-Sentiment Association and Multi-point Opinion Classification phases.

Domain Feature Extraction Phase: The purpose of this phase is to improve the precision and recall of extracting domain features. The main objective is to utilise a comprehensive domain knowledgebase that is populated with domain's ground facts from Linked Open Data resources in order to provide a deep understanding of the free-textual contents. Another objective is to deploy co-referencing resolution to identify non-explicit domain features. The domain knowledgebase is used also to eliminate irrelevant (i.e. false positive) domain features.

Domain Feature-Sentiment Association Phase: The purpose of this phase is to improve the precision of the associated domain features with their corresponding sentiments. The main objective is to generate sentiment lexicons for domain features to identify subjective opinions and remove descriptive opinions.

Multi-point Opinion Classification Phase: The purpose of this phase is to improve the accuracy of the classified opinions on a multi-point scale by integrating an enriched set of semantic features generated from the developed domain knowledgebase with a set of statistical features; the integrated data is then used to train Machine Learning classification algorithms.

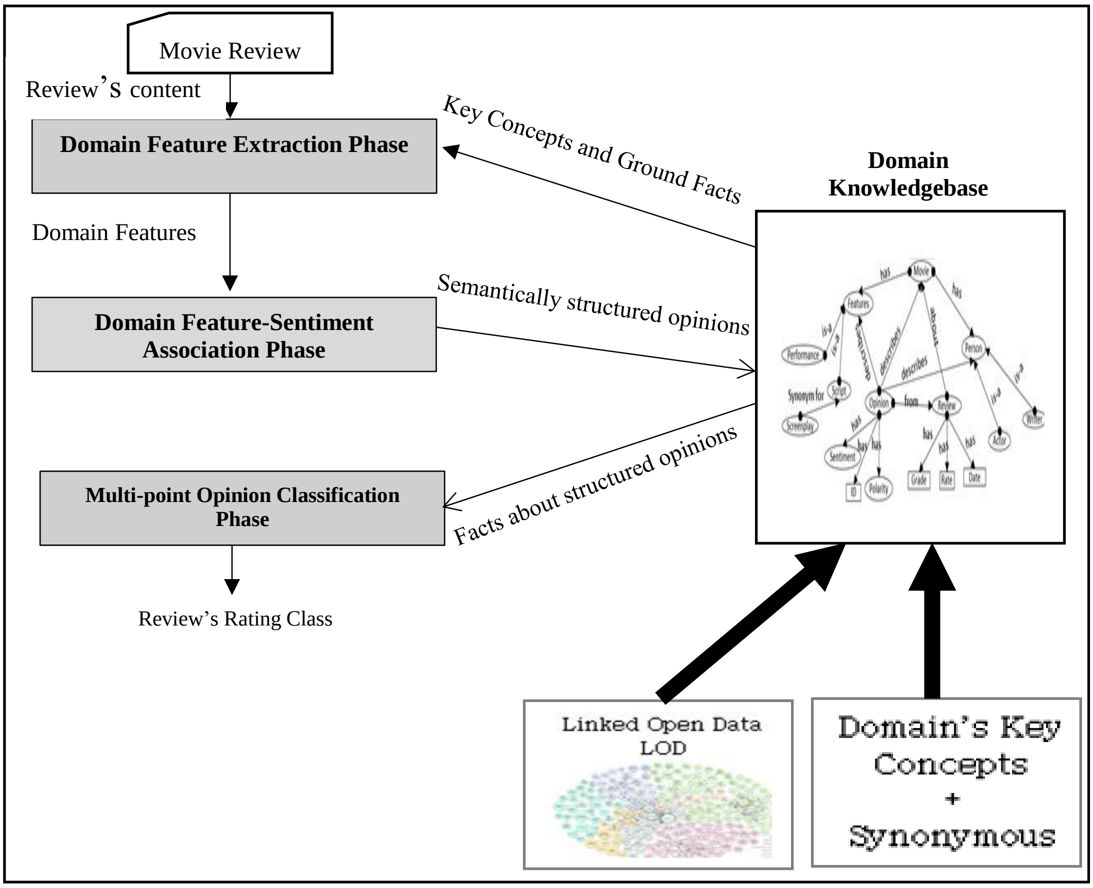


Figure 3.1 A hybrid semantic knowledgebase-machine learning approach for opinion mining at domain feature level

In brief, the new Hybrid Semantic Knowledgebase-Machine Learning approach processes unstructured textual reviews, extracts domain features using a developed domain knowledgebase, and then associates the extracted domain features with relevant sentiments. Thereafter, the new Hybrid Semantic Knowledgebase-Machine Learning approach calculates the polarity for each associated feature-sentiment pair and inserts all the obtained semantic information into the developed domain knowledgebase. The developed domain knowledgebase is used by the new Hybrid Semantic Knowledgebase-Machine Learning approach to further produce a semantic feature dataset, which it is merged with a statistical dataset and then used as input to Machine Learning classifier that delivers multi-point scale rating for the processed reviews.

Constructing a semantic knowledgebase starts with modelling the domain knowledge before translating the knowledge map into formal ontologies that represent the schemata for populating the knowledgebase with structured information. The semantic structure of the knowledgebase provides for obtaining data from other public sources that use similar standards for data structuring such as Linked Open Datasets, which can be used, for instance, to populate the proposed use-case knowledgebase with dynamic ground facts about the problem domain (Omitola, et al. 2014)

Opinion mining of movie reviews is considered a challenging topic because movie reviews tend to include a rich set of domain features (actors, script, plot, etc.). Furthermore, the popularity of the movie domain provides for the opportunity to exploit the ever-increasing crowd-sourced Linked Open Data repository corresponding to the movie and celebrity industry (Gadekallu, et al. 2019).

Using movie reviews as the target problem domain, the next sections describe the proposed methodology for modelling the domain knowledge into a semantic knowledgebase that will be used in our proposed Hybrid Semantic Knowledgebase-Machine Learning approach for extracting and storing the expressed sentiments and associated domain features, and for retrieving semantic features to investigate whether combining it with statistical feature can improve the performance of opinion classification on a multi-point scale.

3.1 Conceptualising the Knowledge of a Problem Domain

Conceptualising the domain's knowledge is based on capturing its knowledge into concepts that are connected together using relations. In addition, the model should illustrate the external relations interrelating concepts from different domains. Our use-case scenario requires interfacing concepts from three domains: Movie, Opinion and Review. The proposed model, termed the movie-review model in this document, encompasses the interaction (relationships) between the three mentioned domains as shown in Figure 3.2, in which the problem domain for opining mining at domain feature level is the Movie domain, whereas, the Opinion and Review domains are complementary domains that present the problem solution. Thus, the model can represent and associate generic information about the movie, opinions as well as its reviews. In addition, our movie-review model is presented in a structured way that allows it to source data from external Linked Open Data resources such as DBpedia and Internet Movie Database (Alfrjani, Osman and Cosma 2016).

The DBpedia knowledgebase is the best source for collecting such ground facts because it contains richer information about the movie domain than other knowledgebases. DBpedia is a knowledgebase that covers multi-domains and enriched with lots of structured ground facts for each domain. These ground facts are extracted from Wikipedia pages. The DBpedia knowledgebase is aimed to represent actual community agreement, to be decentralised, to be evolved automatically when Wikipedia changes and to support multi-languages. Moreover, the DBpedia knowledgebase is stored in the Resource Description Framework and it is available on the Web as one of the Linked Open Data resources which can be semantically retrieved and manipulated using the SPARQL query language (Bizer, et al. 2009).

Internet Movie Database is one of the largest sources of movie information. It is aimed to capture every pertinent details about each movie starting from names of its stars, directors, writers, editors, etc.; filmed location; language; plot; key words; names of its fans and reviewers. Although the contents of Internet Movie Database data is updated regularly, such information is presented in ad-hoc format; which means that this information cannot be retrieved using Linked Open Data resources but can be retrieved via accessing its page source programmatically (Peralta 2007).

Some of the current Semantic Knowledgebase approaches to opinion mining for movie reviews constructed a knowledgebase were restricted to the movie's key concepts and their synonyms as in (Zhao and Li 2009), whereas other Semantic Knowledgebase approaches enriched the knowledgebase by adding more facts about movies that are found in Internet Movie Database as in (Penalver-Martinez, et al. 2014). Different from (Zhao and Li 2009, Penalver-Martinez, et al. 2014), the movie-review model stated in this thesis covers a comprehensive movie's key concepts such as actor, writer, producer, editor, sound, script, twist, performance, special effect, footage, humour, movie theme, costumes, cinematography, emotion, scene, images, ends, background, pacing, staging, story, plot, style and sets. In addition, the movie-review model is populated with ground facts about movies that are retrieved from both DBpedia and Internet Movie Database datasets.

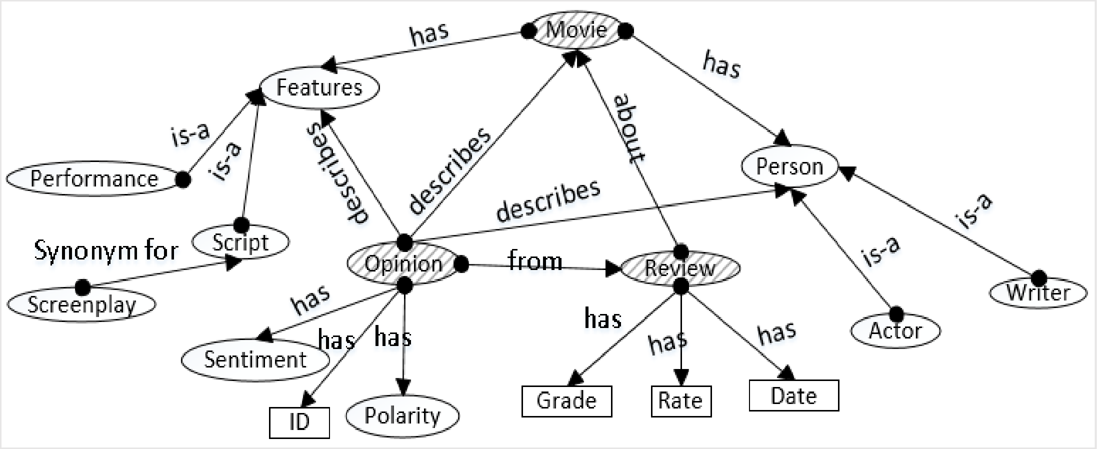


Figure 3.2 Movie-opinion-review domain concept map

Highlights of the important relation modelling decisions are illustrated below:

- For each role related to movie there are sub concepts of PERSON concept such as WRITER, EDITOR, STAR
- For each group of movie’s feature there are sub concepts of FEATURES such as WRITING, EDITIING, CINEMATOGRAPHY, ANIMATION
- For each role related to movie, there is a property such as HAS-STAR property between a movie and a star, and HAS-WRITER property between a movie and a writer.
- There is a SYNONYM annotation for each concept and instance that has a synonym word.
- There is a DESCRIBE-OBJECT property between an opinion and a movie.
- There is a DESCRIBE-FEATURE property between an opinion and a movie’s feature.
- There is an HAS-SENTIMENT property between an opinion and a sentiment.
- There is a ABOUT property between a review and a movie.
- There is an EXTRACTED-FROM property between an opinion and a review.

3.2 Translating the Modelled Domain to a Semantic Knowledgebase

In this research we utilised Semantic Web technologies to translate the domain conceptual model into a formal semantic ontology that represents the template box (T-Box) of the domain knowledgebase. The Semantic Web technologies are concerned with making unstructured data on the Web more understandable to computers via adding linguistic and semantic metadata to the web content (Berners-Lee, Hendler and Lassila 2001). Semantic Web technologies organise domain's knowledge in formalised concept ontologies that provide efficient support for linking and sharing data between resources, and presenting data in a way that computer machines can process. In addition, Semantic Web technologies are capable of presenting the domain's knowledge in a structured and consistent way which facilitates the qualitative interpretation of domain specific contents in a way that people can understand. Moreover, Semantic Web technologies provide support for populating the semantically structured domain knowledgebase with relevant ground facts from public-sourced Linked Open Data resources (Omitola, Ríos and Breslin 2015).

The movie-review conceptual model was translated into a semantic ontology as follows:

1) For the Movie domain, we manually collected comprehensive knowledge of the movie domain with approximately 504 concepts related to movie domain as well as their synonyms and the relationships between them from the Movie Terminology Glossary in (Gartenberg 1989). Then, based on the movie-review conceptual model, we distributed the collected terms as classes (Concepts), instances (ground facts), object properties and annotations. The created primary classes in the movie-review knowledgebase are: MOVIE, FEATURES and PERSON. The class MOVIE is a simple upper class that contains all the individuals that characterise movie names. Each individual movie has datatype values such as released date and running time. The classes PERSON and FEATURES are upper classes that capture movie domain's key concepts. For each role related to movie there are sub-classes of the class PERSON such as WRITER, EDITOR, STAR, DIRECTOR, CINEMATOGRAPHER, PRODUCER, which

represent names of people as individuals with respect to their roles in the movie. For example, the class STAR cantinas names of actors and actresses as individuals. For each group of movie's features there are sub-classes of the class FEATURES such as WRITING, EDITIING, CINEMATOGRAPHY, ANIMATION, SPECIAL EFFECT, SOUND, MUSIC, etc. as shown in Figure 3.3. In addition, four more upper classes were created: AWARD, LOCATION, COUNTRY, and LANGUAGE to capture other semantic information about movies such as nominated award, filming location, and the original language. Moreover, the annotation synonyms is designed to annotate all synonym terms for each concept. For example, the terms "film, show and picture" are annotated as synonyms for the concept MOVIE, hence different terms can be mapped under one concept during extracting domain features from the reviews during opinion mining process.

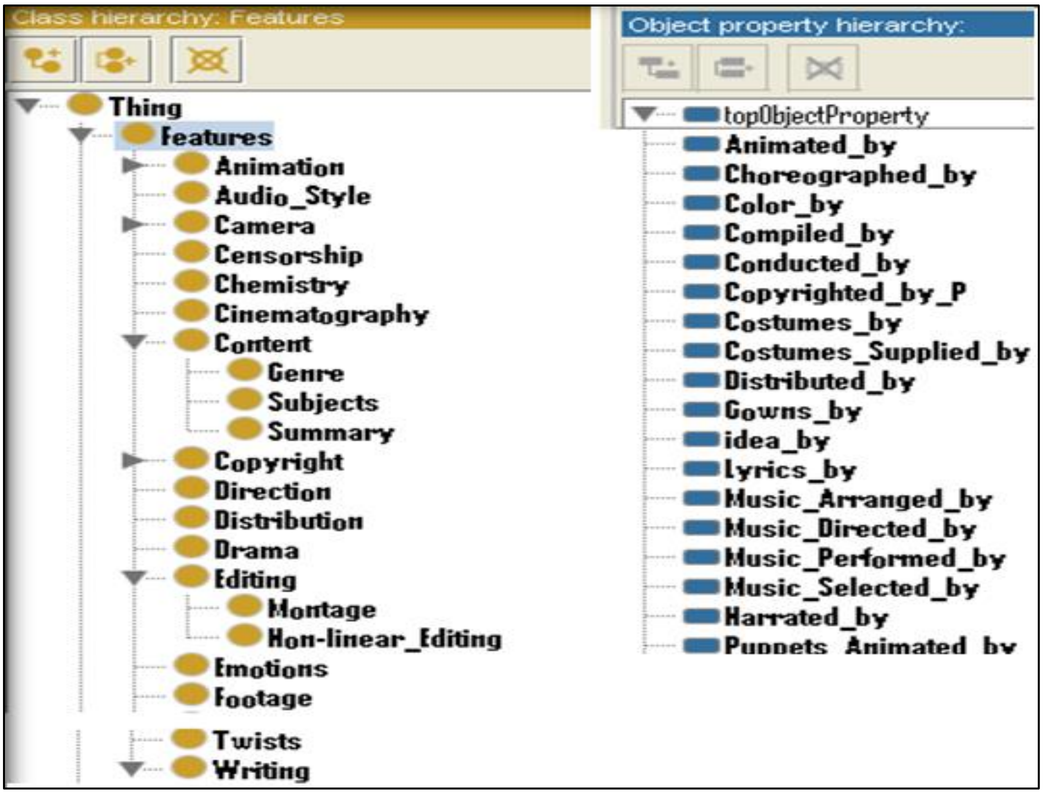


Figure 3.3 A snapshot of the distribution of movie's features within the movie-review knowledgebase

2) The class REVIEW was created to capture the semantic information about movie reviews that contain opinions. The class REVIEW contains reviews' ID as individuals that have REVIEW-DATE as a datatype value, and reviewer's name

via a REVIEWED-BY relation that interconnect the class REVIEW with the class PERSON.

3) The class OPINION was created to capture all the individuals that characterise the expressed opinions in reviews via adopting the designed model of the Marl[2] ontology as shown in Figure 3.4. Marl is a standardised data schema designed to address and to describe subjective opinions expressed on the textual reviews.
4) Opinion mining at domain feature level also focuses on extracting sentiment terms that are used to express opinions, and for this reason the class SENTIMENT was created too. This class includes sentiment terms as individuals such as great, bad, good, interesting, etc. The class SENTIMENT is connected with the movie-review knowledgebase via creating a HAS-SENTIMENT relation between the class OPINION and the class SENTIMENT.

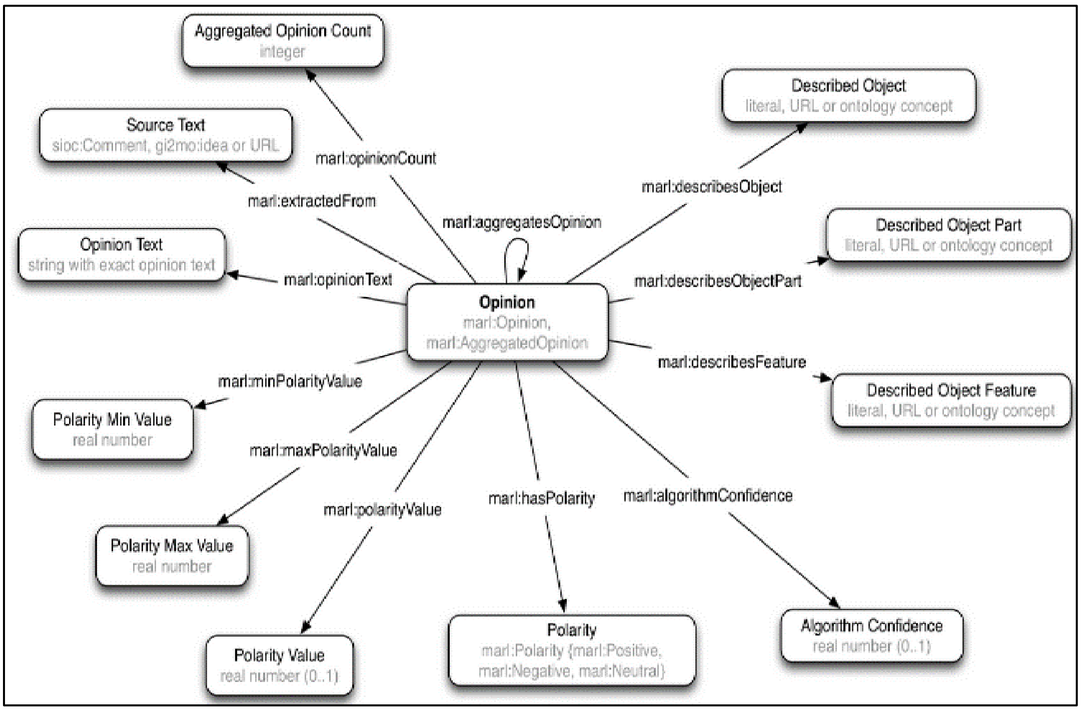


Figure 3.4 Marl ontology model

The advantage of utilising Semantic Web technologies to translate the domain conceptual model into a formal semantic ontology that represents the template box (T-Box) of the movie-review knowledgebase is that they provide for the formal (standardised) representation of the domain's key concepts, their synonyms and

[2] http://www.gsi.dit.upm.es/ontologies/marl/

ground facts, and then link them using relations (i.e. object properties). For example, the ABOUT property links a review to movie, the EXTRACTED-FROM property links an opinion to a review, and the DESCRIBE-FEATURE property links between an opinion and a movie. In addition, Semantic Web Ontology Language (OWL) allows for representing complex relationship between concepts using "typed" object properties such as Functional, Inverse, Transitive, Symmetric and Reflexive relation, which improves the reasoning performed on the knowledgebase. For example, the relation actedIn is the inverse relation of hasStar, since if we state that the movie "Harry Potter" has a star "Hermione Granger", then we can infer that "Hermione Granger" acted in the movie "Harry Potter". Such information can be used to infer valuable semantic information about the main domain concepts (such as movie) as well as the expressed opinions on its constituent features. Therefore, it is possible to compute the overall opinions about a movie across multiple reviews as well as for the cinematic features (actors, script, sound effects, etc.). For example, all movies that have a positive screenplay review can be retrieved by firing one query against the movie-review knowledgebase.

The semantic structure of the knowledgebase provides for obtaining data from other public sources that use similar standards for data structuring such as Linked Open Datasets, which can be used, for instance, to populate the proposed use-case knowledgebase with dynamic ground facts about movies, actors etc., which can contribute to enhance the performance of domain feature extraction task. In this research, the population process is conducted regularly for each processed review as will be explained in the next chapter.

3.3 Chapter Summary

This chapter presented the architecture framework of the proposed Hybrid Semantic Knowledgebase-Machine Learning approach for opinion mining, and addressed the first research question RQ1 (*How can the semantic modelling of the domain knowledge further contribute to improving the opinion mining at domain feature level, in particular to the domain feature extraction and opinion classification tasks?*) via introducing the methodology for the semantic modelling of the problem domain knowledge for opinion mining at domain feature level. The required domain knowledge represents the domain's environment that contains the problem domain's key concepts and synonyms and ground facts, as well as the relation between them. The methodology focused on modelling the domain knowledge in such way that it can be translated to a semantic knowledgebase, which can then be automatically bootstrapped with relevant information from Linked Open Data resources. The semantic modelling of domain knowledge provided for the comprehensive representation of the problem domain, which eased the connection with other related domains such as reviews and opinions for opinion mining process as well as it can facilitate identifying domain features from movie review. In addition, the semantic structure of the knowledgebase based on the semantic modelling can provide us for obtaining dynamic ground facts about the problem domain from other public sources that use similar standards for data structuring such as Linked Open Datasets. Moreover, the semantic modelling of the domain knowledgebase can facilitate the inference of valuable semantic information about the main domain concepts (such as movie) as well as the expressed opinions on its constituent features that in turn can enhance the accuracy of the opinion classification task. Furthermore, the semantic modelling can improve the usability of the developed knowledgebase for sophisticated interrogation of opinions and for recommending a specific domain.

Chapter 4

4 Domain Feature Extraction

This chapter addresses three research questions RQ2 (*Can the domain knowledge improve the precision and recall of the feature extraction task?*), RQ3 (*How can the semantically structured public datasets be exploited to improve the performance of domain feature extraction task*?) and RQ4 (*Given the fact that the target domain feature is presented by a single name or pronoun (i.e. termed non-explicit domain features), how can the semantically constructed knowledgebase be utilised with co-reference resolution to extract non-explicit domain features to further improve the domain feature extraction task?*) via illustrating details of the conducted domain feature extraction process together with the experimental evaluation. The domain feature extraction process is based on a Semantic Knowledgebase approach. The main objective is to utilise a comprehensive domain knowledgebase and populate it with domain's ground facts that are obtained from Linked Open Data resources in order to provide deep understanding of the free-textual contents, which is envisaged to improve the performance of domain feature extraction task.

4.1 Related Work on Domain Feature Extraction

This section discusses related literature in opinion mining with a focus on methods for extracting domain features from natural language text reviews.

The Association Rule Mining approach, which primarily relies on Natural Language Processing techniques, is the most popular for mining online contents to extract domain features. The authors in (Hu and Liu 2004) extracted frequent nouns or noun-phrases to be domain features using an Apriori algorithm. The approach by (Eirinaki, Pisal and Singh 2012) involved initially extracting nouns, and then computing the score for each noun with respect to the total number of their nearest adjectives in all processed textual contents. Nouns with scores less than a particular threshold were removed and the reset were determined to be domain features. The work by (Ghorashi, et al. 2012) was similar to the work in (Hu and Liu 2004) for extracting domain features except that they applied the H-Mine algorithm instead of the Apriori algorithm. The researchers in (Yang, et al. 2015) extracted domain features utilising a semi-automatic constructed knowledgebase that contains the top hundreds of frequently normalized nouns and noun phrases which were extracted from a collection of pre-processed contents.

Association Rule Mining approaches extract domain features without performing human pre-processing tasks (e.g. preparing manually training dataset) because automatic Natural Language Pre-Processing is used to identify nouns and noun phrases to be domain features. However, the extracted domain features tend to be frequent domain features, whereas infrequent domain features are ignored, which can result in a reduced recall rate. In addition, some of the extracted nouns and noun phrases may not be domain features even if these occur more frequently in the textual contents, and this can affect the precision of the domain feature extraction task.

Machine Learning approaches require large trained datasets in order to perform the domain feature extraction task with satisfactory accuracy. The authors in (Zhuang, Jing and Zhu 2006) extracted domain features by training Machine Learning algorithms on manually labelled textual contents with the domain frequent features (key concepts and ground facts). The study by (Ma, et al. 2013) was based on extracting domain features by training Latent Dirichlet Allocation algorithm on automatically labelled contents with nouns or noun phrases, which were tagged via part of speech tagger and the learned domain features were expanded with synonyms, then the obtained candidate domain features were filtered by removing non-relevant domain features. The authors in (Agarwal, et al. 2015b) extracted domain features by training a Machine Learning model to identify the semantic information in a text, which were detected by utilising Concept Net knowledgebase. Thereafter, the irrelevant domain features were removed using Minimum Redundancy and Maximum Relevance techniques.

In general, Machine Learning approaches deliver significant results for domain feature extraction task using training datasets that have been manually annotated by a human expert. However, this can be an extremely time-consuming task as the required size of the training dataset should be sufficiently large to bootstrap the learning algorithms.

More recently, a new trend of studies has utilised Semantic Knowledgebase approaches that are mainly based on the knowledge of the problem domain. These approaches commonly translate the knowledge background of a chosen domain into a semantic knowledgebase, and then utilise this semantic knowledgebase to extract domain features from the pre-processed contents. However, their approaches are different with respect to the coverage of the problem domain. The work done by (Zhao and Li 2009) was based on constructing a semantic knowledgebase that contained only the domain's key concepts and their synonyms. The authors in (Penalver-Martinez, et al. 2014) adopted a general semantic knowledgebase of a chosen domain that contained the domain's key concepts and their synonyms and collected ground facts from Internet Movie Database resources. The study has done by (Agarwal, et al. 2015a) was based on constructing a semantic knowledgebase for a specific domain using concepts from the top four levels of Concept Net knowledgebase, then the contrasted semantic knowledgebase was expanded with synonyms from WordNet.

Semantic Knowledgebase approaches have demonstrated improved performance for domain feature extraction when the knowledge of the domain of interest is utilised to extract domain features. However, the success of these techniques largely depends on the domain knowledge coverage, and the conducted investigation into the state-of-the-art approaches showed that the domain knowledge coverage is often limited.

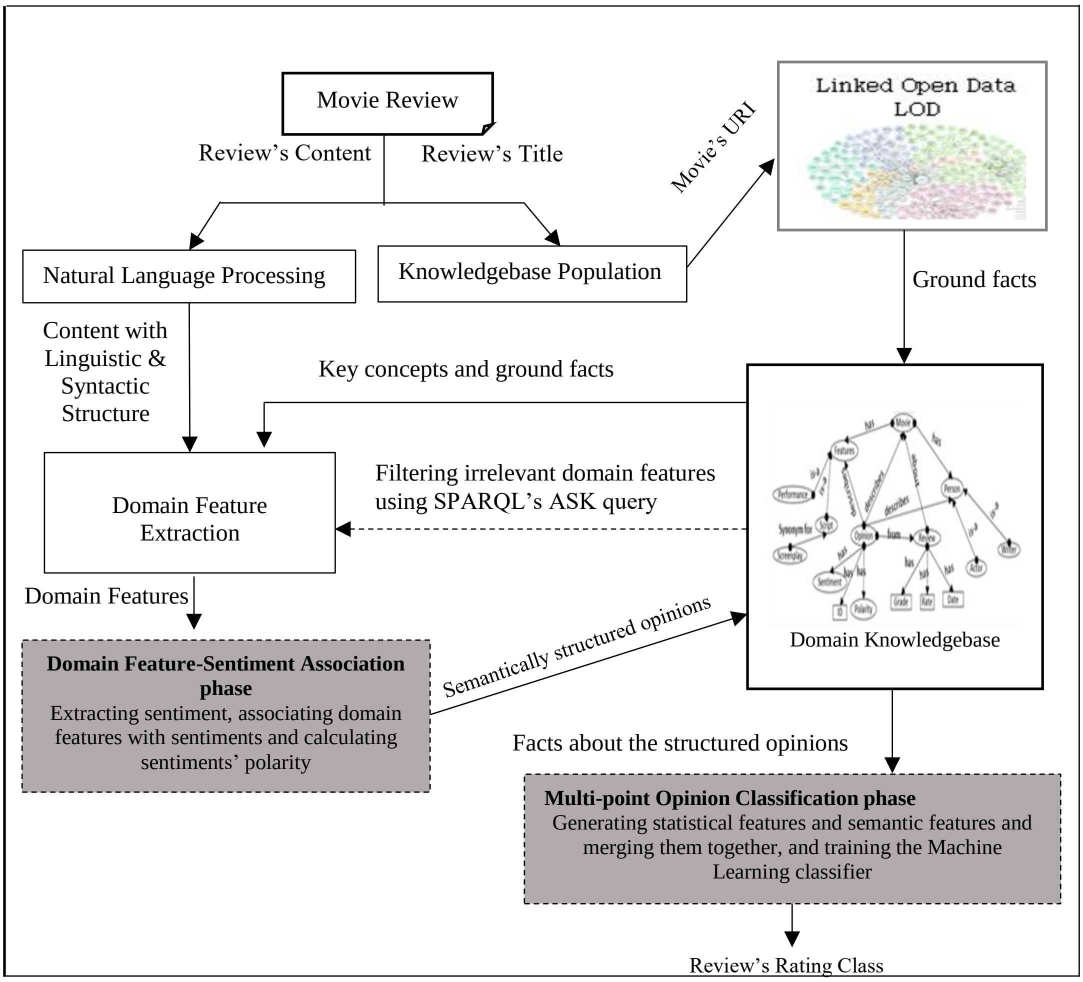
4.2 Design and Implementation of Domain Feature Extraction Phase

To improve the performance of domain feature extraction, firstly public data sources such as DBpedia is exploited to populate the generated movie-review knowledgebase with relevant ground facts about movies, actors, directors, prizes, etc. Then, the movie-review knowledgebase is utilised to extract the movie's features from movie reviews. The movie-review knowledgebase, as described in chapter 3, hosts comprehensive knowledge of the chosen domain: key concepts and synonyms. Secondly, co-reference resolution is deployed to identify non-explicit domain features. Finally, the movie-review knowledgebase is used to eliminate irrelevant (i.e. false positive) domain features.

Figure 4.1 illustrates the architecture of the Domain Feature Extraction phase, which comprises the following main components: Knowledgebase Population, Natural Language Processing and Domain Feature Extraction.

In brief, the Domain Feature Extraction phase processes unstructured textual reviews, populates the developed domain knowledgebase with relevant domain's ground facts and extracts domain features from the processed reviews.

The extracted domain features resulting from Domain Feature Extraction phase will be associated with their corresponding sentiments and determine their sentiment polarities through Domain Feature-Sentiment Association phase, which will be explained in details in the next chapter.



Movie Review
Review's Content
Review's Title
Movie's URI
Linked Open Data
LOD
Natural Language Processing
Knowledgebase Population
Ground facts
Content with Linguistic & Syntactic Structure
Key concepts and ground facts
Filtering irrelevant domain features using SPARQL's ASK query
Domain Feature Extraction
Domain Features
Semantically structured opinions
Domain Knowledgebase
Domain Feature-Sentiment Association phase
Extracting sentiment, associating domain features with sentiments and calculating sentiments' polarity
Facts about the structured opinions
Multi-point Opinion Classification phase
Generating statistical features and semantic features and merging them together, and training the Machine Learning classifier
Review's Rating Class

Figure 4.1 The architecture of domain feature extraction phase

4.2.1 Populating the Semantic Knowledgebase

The aim of populating the knowledgebase is to construct semantically structured information about the problem domain, which is considered valuable for the process of opinion mining at domain feature level. Thus, for each movie review, we populate the movie-review knowledgebase with the relevant ground facts (movie’s name, released date, running time, country and language; movie’s stars, directors, writers, editors, cinematographers, producers, etc.) that are gathered from public data sets such as DBpedia and Internet Movie Database.

As the problem domain is the movie domain, we chose to benefit from Internet Movie Database in addition to DBpedia in terms of gathering names of movie’s stars only. This is because relying on DBpedia as the only resource for gathering movie’s stars may not be sufficient sometimes. This is due to the fact that DBpedia depends on Wikipedia info box as the main resource for Resource Description Framework; and according to our observation, Wikipedia Info box includes only the top main names of movie’s stars, whereas Internet Movie Database contains all names of stars for each movie. Moreover, as in our research, gathering ground facts from DBpedia for a specific movie requires the Uniform Resource Identifiers (i.e. a key to search for any resource in any knowledgebase over the World Wide Web) that we obtained via Google Search Engine and Wikipedia website. We noticed that Google Search Engine sometimes does not return results for some movie reviews that contain movie titles that are written in a format which is different to the way is saved in Wikipedia website. For example, the title of a movie called “THE ADDICTION_1995” sometimes is written in the review as “ADDICTION, THE 1995”, whereas, according to our observation, Internet Movie Database provides advanced search tools that can retrieve the name of the movie even with different format title.

The population process in general is based on extracting a movie’s title from a review, then the relevant ground facts about this movie (movie’s name, released date, running time, country and language; movie’s stars, directors, writers, editors, cinematographers and producers) are gathered from DBpedia and Internet Movie Database resources. The process was performed automatically by following the illustrated steps in Algorithm 1.

Algorithm 1 **Knowledgebase Population**

```
Input:
Reviews R, movie-review Knowledgebase
1.  Do for i=1:R,
2.      MovieName=Extract ( Review[i] )
3.      /* Populating via DBpedia*/
4.      MovieWikiURI=Search (MovieName)
5.      MovieDBpediaURI=MovieWikiURI.Replace(http://en.wikipedia.org,
        "http://dbpedia.org/resource")
6.      MovieGroundFacts=Retrieve (MovieDBpediaURI)
7.      movie-review Knowledgebase =Insert  (MovieGroundFacts)
8.      /* Populating via Internet Movie Database */
9.      MovieIMD-URI=Search (MovieName)
10.     Movie'sStars=Retrieve (MovieIMD-URI)
11.     movie-review Knowledgebase =Insert  (Movie'sStars)
12. End for
Output: Populated movie-review Knowledgebase
```

Regarding gathering ground facts from DBpedia, steps 2-7 in the above algorithm are executed, which is based on obtaining the target movie’s URI (i.e. Uniform Resource Identifiers) in DBpedia knowledgebase by searching for the Wikipedia page of the target movie (i.e. movie’s Wiki-URI), and replacing the first part of movie’s Wiki-URI with DBpedia URI “http://dbpedia.org/resource”. For example, the Wiki-URI for THE ADDITION 1995 movie is “https://en.wikipedia.org/wiki/The_Addiction_1995” will be changed to “http://dbpedia.org/resource/The_Addiction_1995” and this the DBpedia URI for the target movie.

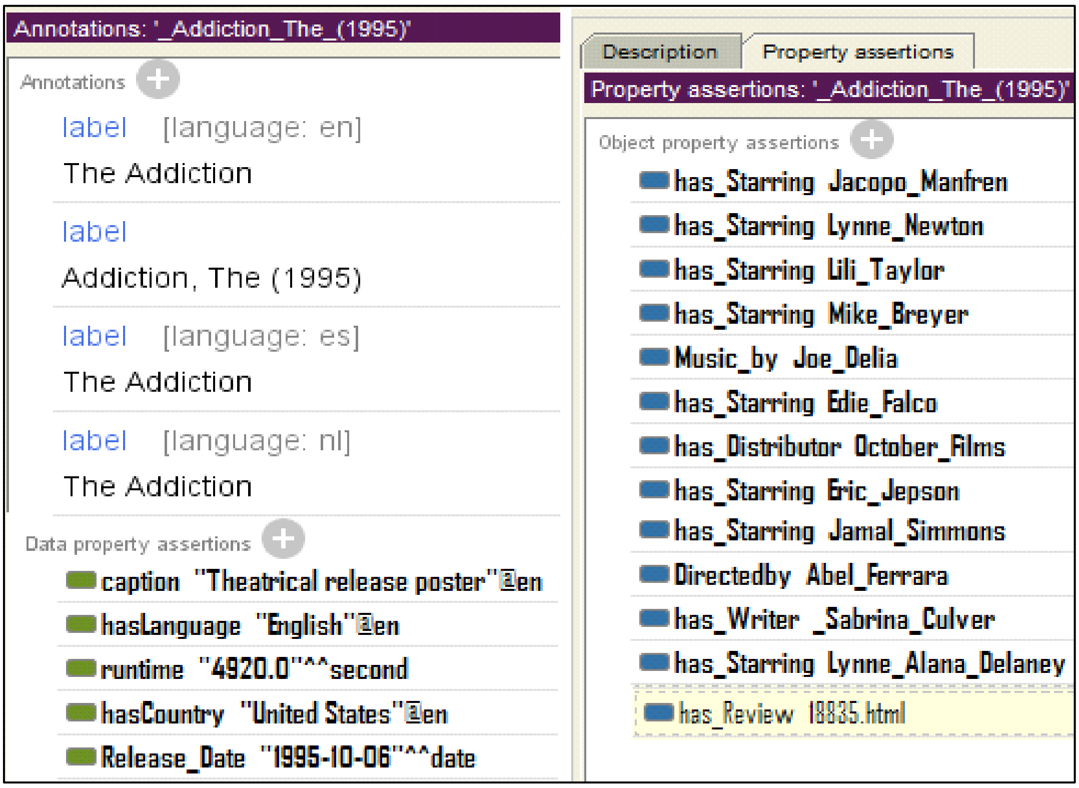
After that, the obtained DBpedia URI for the target movie, it is used to retrieve from the DBpedia knowledgebase the ground facts about the target movie and inserting them into the movie-review knowledgebase. The retrieving and inserting steps are performed together via composed SPARQL Construct queries as shown in Figure 4.2. SPARQL Construct query is a language that is used to perform semantic queries over semantic knowledgebase where the retrieved data is stored in Resource Description Framework (Prud and Seaborne 2006).

```
prefix owl:<http://www.movie-review-ontology.owl#>
prefix dbpedia-owl:<http://dbpedia.org/ontology/>
prefix rdfs:<http://www.w3.org/2000/01/rdf-schema#>
prefix rdf:<http://www.w3.org/1999/02/22-rdf-syntax-ns#>
prefix dbpprop:<http://dbpedia.org/property/>
CONSTRUCT         {      ?subject owl:movie_Title ?name .
         ?subject rdfs:label ?label .
         ?subject rdfs:label "ADDICTION,THE (1995)".
         ?subject rdf:type owl:Movie .
         ?subject owl:hasLanguage ?language .
         ?subject owl:hasCountry ?country .
         ?subject owl:has_Starring ?star .
         ?subject owl:has_Writer    ?writer .
         ?subject owl:directed_by   ?director .
         ?subject owl:edited_by      ?editor.  }
WHERE      {     VALUES  ?subject
{<http://dbpedia.org/resource/The_Addiction_1995>}
      ?subject a dbpedia-owl:Film.
      OPTIONAL  {?subject rdfs:label ?label.}
      OPTIONAL  {?subject dbpprop:name ?name.}
      OPTIONAL {?subject dbpprop:language ?language.}
      OPTIONAL {?subject dbpprop:country ?country.}
      OPTIONAL {?subject dbpedia-owl:starring ?star .}
      OPTIONAL {?subject dbpedia-owl:writer    ?writer .}
      OPTIONAL {?subject dbpedia-owl:editing   ?editor .}
      OPTIONAL {?subject dbpedia-owl:director  ?director .  } }
```

Figure 4.2 Example of sparql construct query

Although movie reviews are collected from the crowd-sourced data that provides extensive information with a high level of accuracy, it is likely that some movie reviews may contain incorrect information due to human error. For example, THE ADDICTION_ (1995) movie sometimes is written in the review as "ADDICTION, THE". Therefore, for disambiguation, the extracted title is inserted into the movie-review knowledgebase in addition to movie's name that is retrieved from the DBpedia knowledgebase.

Regarding gathering ground facts from Internet Movie Database website, steps 9-11 are performed. The obtained results from this step are names of stars, which they are retrieved from Internet Movie Database page source of the target movie. The obtained list of star names were injected into the movie-review knowledgebase using SPARQL Construct queries. Figure 4.3 presents a snapshot of the populated semantic information about THE ADDICTION (1995) movie into movie-review knowledgebase.



Annotations: '_Addiction_The_(1995)'

Annotations

label [language: en]
The Addiction

label
Addiction, The (1995)

label [language: es]
The Addiction

label [language: nl]
The Addiction

Data property assertions

caption "Theatrical release poster"@en
hasLanguage "English"@en
runtime "4920.0"^^second
hasCountry "United States"@en
Release_Date "1995-10-06"^^date

Description | Property assertions

Property assertions: '_Addiction_The_(1995)'

Object property assertions

has_Starring Jacopo_Manfren
has_Starring Lynne_Newton
has_Starring Lili_Taylor
has_Starring Mike_Breyer
Music_by Joe_Delia
has_Starring Edie_Falco
has_Distributor October_Films
has_Starring Eric_Jepson
has_Starring Jamal_Simmons
Directedby Abel_Ferrara
has_Writer _Sabrina_Culver
has_Starring Lynne_Alana_Delaney
has_Review 18835.html

Figure 4.3 A snapshot of populated semantic information into movie-review knowledgebase about the addiction movie

4.2.2 Pre-processing the Domain Reviews Using Natural Language Engine

The main objective of this process is to obtain the linguistic and syntactic structure of the textual review. Hence, Natural Language Processing tools have been implemented via the GATE[3] framework (General Architecture for Text Engineering). GATE is a well-established infrastructure that facilitate users to customise and develop Natural Language Processing components, whereas handling other routine processes (e.g. format analysis, data visualisation, data storage, etc.) are done automatically by GATE.

The pre-processing phase is described below using a running example of the sentence S1: “The movie is not excellent”.

1) Tokenisation: each review in the dataset is converted into tokens. Each token has a unique number, position (start and end), and other features such as length of the token. Table 4.1 shows an example of the tokenised sentence S1.

[3] http://gate.ac.uk

Table 4.1 Example of a tokenised sentence

Type	Start	End	ID	Features
Token	0	3	1	{ kind=word, length=3, orth=upperInitial, string=The}
Token	4	9	3	{ kind=word, length=5, orth=lowercase, string=movie}
Token	10	12	5	{ kind=word, length=2, orth=lowercase, string=is}
Token	13	16	7	{ kind=word, length=3, orth=lowercase, string=not}
Token	17	26	9	{ kind=word, length=9, orth=lowercase, string=excellent}

2) Sentence Splitting: each tokenised review is split into sentences based on a delimiter such as a full stop punctuation mark ".".

3) Part of Speech Tagging: is applied to identify the part of speech of each token in the review whether it is a noun, verb, adjective, adverb, etc. This category will be added to each token as a feature. Table 4.2 shows part of speech tagging to the sentence S1.

Table 4.2 Example of a tagged sentence

Type	Start	End	ID	Features
Token	0	3	1	{category=DT, kind=word, length=3, orth=upperInitial, string=The}
Token	4	9	3	{category=NN, kind=word, length=5, orth=lowercase, string=movie}
Token	10	12	5	{category=VBZ, kind=word, length=2, orth=lowercase, string=is}
Token	13	16	7	{category=RB, kind=word, length=3, orth=lowercase, string=not}
Token	17	26	9	{category=JJ, kind=word, length=9, orth=lowercase, string=excellent}

4) Morphological Analysis: is about formatting each token in the review to its root. This feature "root" will be added to the token as a feature. In the sentence S1 for example the root of the word "is" will be "be".

5) Syntax and Dependency Parsing: aims to identify the grammatical relationships between tokens in a sentence such as "amod" and "nsubj" for adjectival phrase (i.e. serves to modify the meaning of the noun phrase such as "nice movie") and noun subject phrase (i.e. The syntactic subject of a clause such as “This is great movie”) respectively. Table 4.3 and Table 4.4 show the syntax and the dependency parse for the sentence S1 respectively.

Table 4.3 Example of applying a syntax analyse on a sentence

Type	Start	End	ID	Features
SyntaxTreeNode	0	26	19	{ID=19, cat=ROOT, consists=[18], text=The movie is not excellent}
SyntaxTreeNode	0	26	18	{ID=18, cat=S, consists=[12, 17], text=The movie is not excellent}
SyntaxTreeNode	0	3	10	{ID=10, cat=DT, text=The}
SyntaxTreeNode	0	9	12	{ID=12, cat=NP, consists=[10, 11], text=The movie}
SyntaxTreeNode	4	9	11	{ID=11, cat=NN, text=movie}
SyntaxTreeNode	10	26	17	{ID=17, cat=VP, consists=[13, 14, 16], text=is not excellent}
SyntaxTreeNode	10	12	13	{ID=13, cat=VBZ, text=is}
SyntaxTreeNode	13	16	14	{ID=14, cat=RB, text=not}
SyntaxTreeNode	17	26	16	{ID=16, cat=ADJP, consists=[15], text=excellent}
SyntaxTreeNode	17	26	15	{ID=15, cat=JJ, text=excellent}

Table 4.4 Example of applying a dependency analyse on a sentence

Type	Start	End	ID	Features	String
Dependency	0	9	20	{ args=[2, 0], kind=det }	The
Dependency	0	26	24	{ args=[9, 8], kind=root }	The movie is not excellent
Dependency	4	26	21	{ args=[8, 2], kind=nsubj }	movie is not excellent
Dependency	10	26	22	{ args=[8, 4], kind=cop }	is not excellent
Dependency	13	26	23	{ args=[8, 6], kind=neg }	not excellent

In our work, tokenisation, sentence splitting, and part of speech tagging, were performed using the relevant components found in A Nearly-New Information

Extraction framework (ANNIE) that is included within GATE[4]. Regarding morphological analysis, we relied on the GATE Morphological component. Finally, we adopted the Stanford Parser as an embedded application in GATE for syntax and dependency parsing. Figure 4.4 illustrates a high level diagram of the linguistic and syntactic analysis that were carried by the Natural Language Processing components for the sentence "This movie makes me happy". In the figure, "Token.Category" points to the part of speech for each tokenised word, "Token.root" indicates the root of each tokenised word and "Dependency.kind" lists various relationships between tokens.

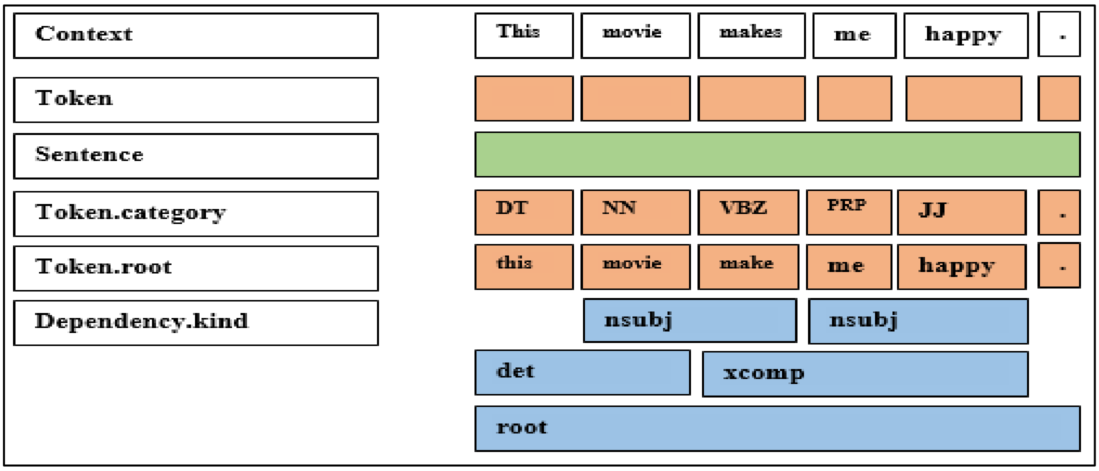


Figure 4.4 Example of a processed sentence linguistically and syntactically

The obtained grammatical categories from these analyses are used to enhance the domain feature extraction task. For example, many words in reviews cannot be matched to the conceptualised domain features in the movie-review knowledgebase because they are found as nouns (singular and plural) or verbs. Hence, morphological analysis is performed to lemmatise each word in the review to enable the matching with the domain feature via the common base. Also, as part of the Natural Language Processing process, dependency relations are analysed to determine the relation between the domain feature and a sentiment in a sentence. For example, the dependency relations (amod and nsubj) are used to identify adjectival and noun subject phrases respectively, which intend to contain a domain feature and its corresponding sentiment.

[4] http://gate.ac.uk

4.2.3 A Novel Domain Feature Extraction Algorithm

The domain feature extraction task is performed using the proposed new Domain Feature Extraction algorithm that is summarized below, which is primarily driven by the developed movie-review knowledgebase:

Algorithm 2 **Domain Feature Extraction**

```
Input:
Pre-Processed Reviews R, movie-review Knowledgebase contains key concepts, synonyms, and ground facts
Do for i=1: R,
-----------------------------------------------------------------------------------------------------
                              /*Extracting Domain features*/
    KeyConcepts=Extract(Review [i], movie-review)
    GroundFacts=Extract(Review [i], movie-review)
     MovieNames=Extract(Review [i], movie-review)
-----------------------------------------------------------------------------------------------------
                         /*Extracting Non-explicit Domain Feature*/
    FullNamePeople=Identify(GroundFacts)
    SingleNamePeople=Identify(GroundFacts)
    Pronouns=Identify(Reviews[i] )
    CoReferencedSingleNames=InheritOrthographic (FullNamePeople, SingleNamePeople)
    CoReferencedPronouns=InheritPronominal (FullNamePeople, Pronouns)
    ExpandedGroundFacts=Specify(GroundFacts,CoReferencedSingleNames
    CoReferencedPronouns)
-----------------------------------------------------------------------------------------------------
                              /*Filtering Domain features*/
    K= Count(KeyConcepts)
    Do for j=1:K,
        If (KeyConcepts[j] is Uppercase Letter),
            Discard (KeyConcepts[j])
        End if
    End for
    K= Count(ExpandedGroundFacts)
    Do for j=1:K,
        If (ExpandedGroundFacts[j] is not related to the reviewed movie in review[i]),
            Discard(ExpandedGroundFacts[j])
        End if
    End for
    K= Count(MovieNames)
    Do for j=1:K,
        If (MovieNames[j] is not related to the reviewed movie in review[i]),
            Discard(MovieNames[j])
        End if
        If (MovieNames[j] is Lowercase Letter),
            Discard (MovieNames[j])
        End if
    End for
    Domain-Features=Specify(KeyConcepts,ExpandedGroundFacts,MovieNames)
End for
Output: Domain features
```

As illustrated in the Domain Feature Extraction algorithm, the process contains the steps described below.

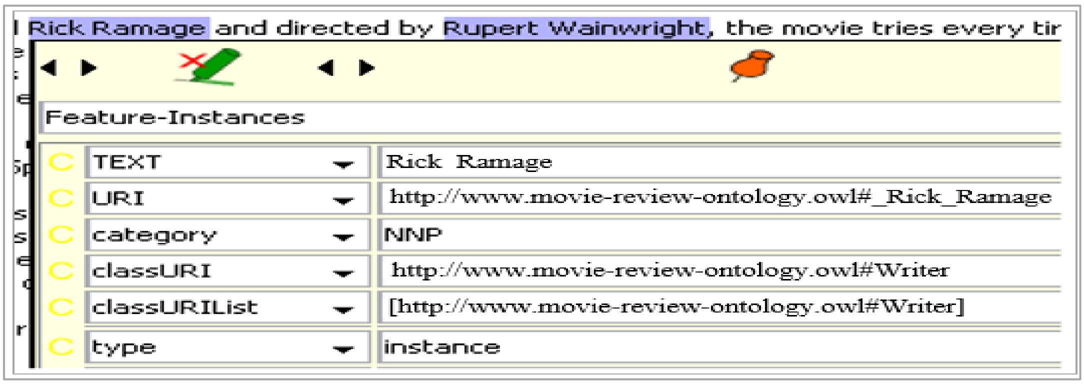
Step 1: Extracting domain features by the movie-review knowledgebase

The movie-review knowledgebase was utilised to link between its conceptualised knowledge (domain's key concepts and their synonyms and ground facts) and the lemmatised words in the review. Synonym words are matched to their key concepts in the movie-review knowledgebase. For example, the word (movie) and synonym words (film, show and picture) are matched to the same key concept (MOVIE) in the movie-review knowledgebase. Words that represent ground facts such as movie names, names of stars, writers, and editors are matched to the same individuals in the movie-review knowledgebase. In the use-case movie review (Figure 4.5), the identified domain features by the movie-review knowledgebase are (ADDICTION THE (1995), THE ADDICTION, movie, Spike Lee, movie, ADDICTION THE, script, Katie Virant and performance) respectively.

> ADDICTION, THE (1995)
> ADDICTION, THE is an excellent movie. From Spike Lee's very first movie, ADDICTION, THE, he has demonstrated fresh and interesting approaches to standard material…The script is good and provides several large laughs…The great Katie Virant will probably be forever remembered. She is fantastic and her performance is amazing.

Figure 4.5 Example of movie review

In this research, we used GATE's Onto Root Gazetteer (ORG) to link between the root of each word in the pre-processed reviews and the conceptualised terms in the semantically structured movie-review knowledgebase. In particular, ORG annotates domain features (domain's key concepts, synonyms and ground facts) using a flexible and dynamic source of a gazetteer. This gazetteer is produced by ORG in which it pre-processes the movie-review knowledgebase by means of tokenisation and morphological analysis. The annotated domain features within the reviews are given the same classification within the knowledgebase. For example, the annotated word "movie" is classified as a class because it is mapped using ORG to the class Movie in the movie-review knowledgebase; whereas, the annotated word "THE_ADDICTION_1995" is classified as an instance of the class Movie, and this also applies to synonyms, attributes and relationships. Figure 4.6 presents a snapshot of annotated domain features by ORG. It is important to mention that ORG annotates all domain features with their classification under a set called “Lookup”, hence we divided the annotated domain features based on their classification. For example, domain features that are instances were grouped under a set called “Feature-Instances”. For that, we devised a set of hand-crafted JAPE rules as shown in Figure 4.7.



l Rick Ramage and directed by Rupert Wainwright, the movie tries every tir

Feature-Instances

TEXT	Rick Ramage
URI	http://www.movie-review-ontology.owl#_Rick_Ramage
category	NNP
classURI	http://www.movie-review-ontology.owl#Writer
classURIList	[http://www.movie-review-ontology.owl#Writer]
type	instance

Figure 4.6 A snapshot of annotated feature by onto root gazetteer

```
Phase: Instance_Phase
Input: Lookup
Options: control = applet
Rule: InstanceLookup
({Lookup.type == “instance”}):label
-->
{
gate.AnnotationSet matchedAnns =
gate.AnnotationSet)bindings.get(“label”);
gate.Annotation matchedA =
(gate.Annotation)matchedAnns.iterator().next();
gate.FeatureMap newFeatures= Factory.newFeatureMap();
outputAS.add(matchedAnns.firstNode(),matchedAnns.lastNode(),”
Instance”, newFeatures);
 }
```

Figure 4.7 Example of jape rules for instance annotation

Step 2: Extracting non-explicit domain features using co-reference resolution process

Once domain features are identified by the movie-review knowledgebase, co-reference resolution is applied to identify non-explicit domain features from movie reviews such as names of people related to the movie (stars, editors, writers, etc.), which are found within the expressed opinions as single names or pronouns. According to the conducted observation in this research on movie reviews, reviewers tend to mention the full name of people at the first time of expressing opinions on them, and then only single names or pronoun are mentioned to express opinions. The conducted experiment in (Kessler and Nicolov 2009) revealed that the target domain feature is presented by a pronoun within 14% of the expressed opinions. Hence, identifying such non-explicit domain features is essential to enhance the domain feature extraction task, which leads to improve the process of opinion mining at domain feature level.

The proposed co-reference resolution process is based on determining the orthographic relation between two names that refer to the same person in which one name is mentioned in a full name such as “Spike Lee”, whereas the other name is mentioned in a single name such as “Lee” or “Spike”. In addition, it is based on detecting the pronominal relation between a person name and a pronoun. For example, in the sentence “Spike Lee is a great director. Also, he is an amazing actor” the anaphor “he” follows the expression to which it refers, i.e. Spike Lee.

Detecting the orthographic relation and pronominal relation requires a Person annotation to be generated first; this entails grouping all names (full names and single names) and pronouns (he and she) under a Person annotation, which in turn can ensure performing an accurate matching. Full names of stars, editors, writers, and so forth are matched by the movie-review knowledgebase as mentioned in the previous step, whereas single names and pronouns are identified using hand-crafted JAPE rules with the aid of GATE’s named entity component called ANNIE Transducer. Secondly, GATE’s co-referencing components have been used to perform matching and co-referencing between the annotated full names, single names and pronouns.

Finally, the co-referenced single names and pronouns are mapped to their corresponding individuals in the movie-review knowledgebase, where the mapped individuals present full names of people who are related to movies. For example, after determining the pronominal relation, the pronouns, he and she in the mentioned review in Figure 4.5 will be mapped to the director “Spike Lee” and actress “Katie Virant” respectively, which are individuals in the movie-review knowledgebase. Figure 4.8 demonstrates all the above mentioned procedures for performing co-reference resolution with the aid of hand crafted JAPE rules.

The experimental evaluation described in section 4.3.3 shows that the co-reference resolution process increases the recall of the extracted domain features, particularly for reviews where the mention of the participants’ (stars, directors, writers, editors, producers) varies between using their full names, single names and pronouns.

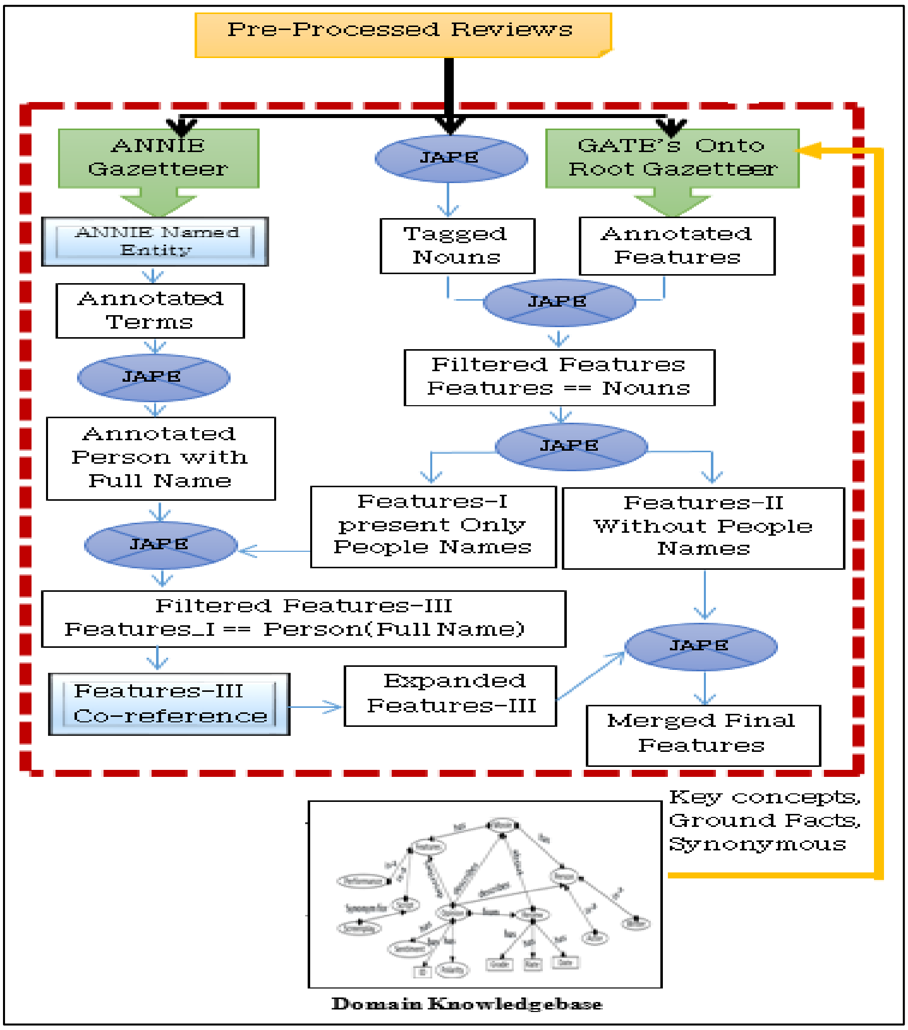


Figure 4.8 Low level diagram of using jape rules for co-reference resolution

Step 3: Filtering out the non-relevant extracted domain features

It has been observed that characteristic of reviews for movie domain is the use of uppercase letters for movie names; hence, hand-crafted rules were applied to discard matched movie names that are typed in lowercase. In addition, to deal with matched movie's features that are typed in upper case letters (similar to movie names). For example, in the sentence "Although Spike Lee's PICTURE, for which he won an Academy Award for the writing, is arguably his best-known film, his picture MALCOLM X, starring Denzel Washington, remains my personal favourite", the term "PICTURE" points to movie name, whereas the term "picture" is a movie's feature. Moreover, it has been observed that movie reviews contain opinions on movie's features such as (movie names and names of stars, writers, editors, etc.) that belong to the target movie as well as to other movies that are sometimes discussed in the review.

Hence, the relevant semantically structured ground facts about the target movie were exploited to discard irrelevant domain features. SPARQL's ASK query was used to verify whether the extracted domain feature is relevant to the semantically structured ground facts in the movie-review knowledgebase or not as illustrated in Figure 4.9. In the query, the extracted name of a person is checked to determine its relevance to the target movie or not (i.e. a star, writer, editor, director, producer or cinematographer, etc.).

```
The Query:
prefix owl:<http://www.movie-review-ontology.owl#>
ASK
{owl:The_Addiction_1995  ?Relation  owl:Spike_Lee. }
.............................................................
The Result:
True
```

Figure 4.9 Ask sparql query for examining the relevant and irrelevant domain features

4.3 Experimental Evaluation

This section presents the conducted experiments on a movie review dataset as a case study in order to evaluate the performance of the domain feature extraction task.

4.3.1 Datasets

Cornell Movie Review Dataset[5] was used for the experiments, and this dataset has been widely used in the sentiment analysis literature (Mukras and Carroll 2004, Allison 2008, Li and Liu 2012). The dataset contains 1770 movie reviews and their corresponding numerical rating for 3-class classification [0, 1, and 2 — essentially "negative", "middling", and "positive", respectively] and for 4-class classification [0, 1, 2, and 3 — essentially "negative", "middling", "positive", and "very positive", respectively]. A total of 475 sentences containing 9301 words were selected from the downloaded dataset, and then from the selected sentences, domain features (277 Key concepts and synonyms, 18 movies' names, 91 names of people related to movies, 36 pronouns) were manually extracted. The manually identified domain features baseline

[5] http://www.cs.cornell.edu/people/pabo/movie-review-data

were used to evaluate the obtained domain features via the novel Domain Feature Extraction algorithm.

4.3.2 Experimentation Methodology

Using the domain feature extraction phase, the obtained movie reviews were processed. Then, the generated movie-review knowledgebase was populated with relevant ground facts from public datasets. After that, the movie reviews were processed linguistically and syntactically to tokenise, tag and lemmatise words as well as to determine the relation between them. Further, the target domain features were extracted from reviews and filtered to remove irrelevant extracted domain features by the proposed Domain Feature Extraction algorithm.

4.3.3 Experimental Results of Domain Feature Extraction Task

The evaluation is based on comparing the performance of the proposed Domain Feature Extraction algorithm against the prepared domain feature baseline results as well as against two existing approaches that adopt Semantic Knowledgebase technique. In particular, three experiments were performed using the proposed Domain Feature Extraction algorithm on the same selected sentences from the downloaded reviews (that contain the baseline extracted domain features) for three constructed knowledgebases. In the first experiment (EXP1), the developed movie-review knowledgebase in this research was utilised, which contains a comprehensive knowledge about movie domain (key concepts, synonyms and ground facts that are collected from DBpedia and Internet Movie Database resources) as described in section4.2.1. In the second (EXP2) and third (EXP3) experiments, two knowledgebases K1 and K2 were developed and used as described in the state of the art researches (Zhao and Li 2009, Penalver-Martinez, et al. 2014) respectively. The K1 knowledgebase contains only the movie domain’s key concepts and their synonyms while the K2 knowledgebase is a general movie domain knowledgebase that contains few number of movie’s key concepts, synonyms and ground facts that were collected from Internet Movie Database resources.

Equation 4.1 and Equation 4.2 were used to compute the Precision and Recall of the extracted domain features.

Equation 4.1

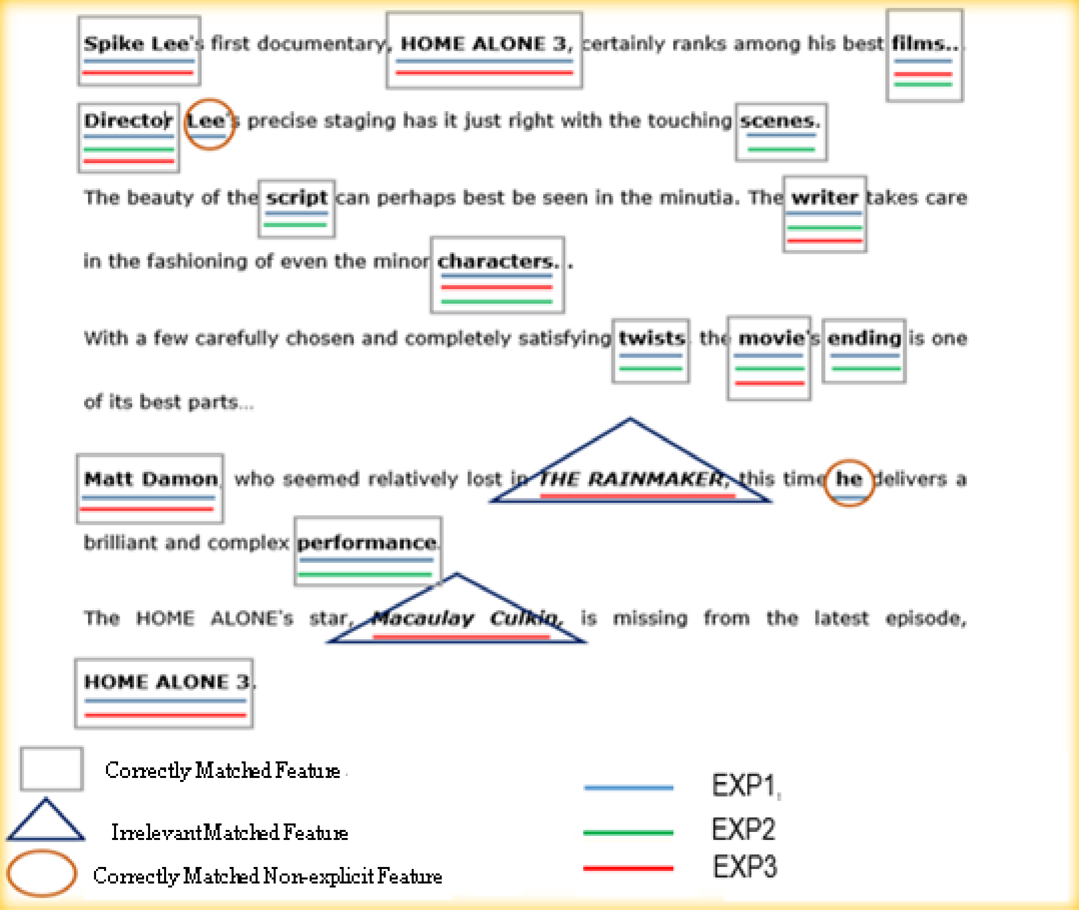
$$\textbf{Precision}= \frac{|\{\text{relevant domain features}\} \cap \{\text{retrieved domain features}\}|}{|\{\text{retrieved domain features}\}|}$$

Equation 4.2

$$\textbf{Recall}= \frac{|\{\text{relevant domain features}\} \cap \{\text{retrieved domain features}\}|}{|\{\text{relevant domain features}\}|}$$

To demonstrate the comparison between the three experiments (EXP1, EXP2 and EXP3), Figure 4.10 presents the obtained results by each experiment from a review about the “HOME ALONE 3” movie. The correctly matched domain features are labelled by squares, the irrelevant matched domain features are labelled by triangles, and the correctly matched domain features using co-reference resolution process are labelled by circles. The matched domain features by each experiment are underlined with underline colour. For example, the domain feature “film” in the review was underlined with three underline colours, which are blue, green and red for EXP1, EXP2 and EXP3 respectively.

In all experiments (EXP1, EXP2 and EXP3), the main focus was on evaluating the number of the retrieved domain features (Recall) via different coverage of the used knowledgebases. The results illustrated in Table 4.5 indicate that the proposed Domain Feature Extraction algorithm achieved high overall recall (86%) even before considering the co-reference resolution in EXP1 in the case that the developed comprehensive movie-review knowledgebase was utilised, whereas the Domain Feature Extraction algorithm achieved 64% and 57% recall in EXP2 and EXP3 when the K1 knowledgebase and K2 knowledgebase were utilised. In terms of the precision, all the experiments EXP1, EXP2 and EXP3 achieved precision (100%) because all the annotated baseline domain feature were extracted (i.e. relevant domain features) via all the experiments.



Spike Lee's first documentary, HOME ALONE 3, certainly ranks among his best films..

Director Lee's precise staging has it just right with the touching scenes.

The beauty of the script can perhaps best be seen in the minutia. The writer takes care in the fashioning of even the minor characters. .

With a few carefully chosen and completely satisfying twists the movie's ending is one of its best parts...

Matt Damon who seemed relatively lost in THE RAINMAKER, this time he delivers a brilliant and complex performance.

The HOME ALONE's star, Macaulay Culkin, is missing from the latest episode, HOME ALONE 3.

Correctly Matched Feature
Irrelevant Matched Feature
Correctly Matched Non-explicit Feature
EXP1
EXP2
EXP3

Figure 4.10 Example of extracted domain features via the experiment EXP1, EXP2 and EXP3

Table 4.5 Recall of the domain feature extraction based on the coverage of three different knowledgebases

Experiments	**EXP1**	**EXP2**	**EXP3**
Used Knowledgebase	The Proposed movie-review	K1	K2
Precision	100%	100%	100%
Recall	86%	64%	57%

Table 4.6 shows the obtained results of re-running experiment EXP1 after deploying the co-reference resolution in the new proposed Domain Feature Extraction algorithm. The recall of domain feature extraction increased by 7% after applying co-referencing, which means that single names and pronouns were co-referenced with movie domain features successfully. These single names and pronouns refer to people related to a movie in a particular review. Thus, the results show that deploying co-

reference resolution enhances the recall performance of domain feature extraction process, especially for movie review domain, where it was observed that reviewers tend to use single names and pronouns most of the time after mentioning in the review the full name of the star, writer, editor, etc. at the first time.

However, similar to (Khan, Atique and Thakare 2015), we observe that dealing with the terms "this and it" when they are used to refer to a movie name has affected slightly the success of the co-reference process. For example, the term "this" in the sentence "unlike the first movie, DR. NO, which rarely flagged, this one is very boring" is referring to the movie "DR. NO".

Table 4.6 Recall of domain feature extraction before and after deploying co-referencing resolution

Domain Feature Extraction Algorithm	**Before Co-referencing**	**After Co-referencing**
Recall	86%	93%

Experiment EXP1 was again rerun to evaluate the impact of eliminating the non-relevant domain features by querying the movie-review knowledgebase ground facts that were obtained from public Linked Open Data sources. The results evidenced that this step improved the precision of the domain feature extraction process as the number of the retrieved domain features before filtering was 525 and after filtering was 407, and hence 118 of the retrieved were detected as non-relevant and removed. Based on the experiment EXP1, all of the 118 non-relevant domain features were movie's domain ground facts such as names of star, writer, editor, etc. as well as names of movies, however, these ground facts were determined as non-relevant because they are not relevant to the reviewed movie in a particular review.

4.4 Discussion

In this chapter, the Domain Feature Extraction phase was explained, which basically relies on utilising the Semantic Knowledgebase approach to analyse the content at domain feature level to improve the precision and recall of the domain feature extraction task via a new Domain Feature Extraction algorithm. The experimental results demonstrated that the proposed algorithm improved the performance of domain feature extraction task. The main objective of our work is to utilise a comprehensive domain knowledgebase and populate it with domain's ground facts that are obtained from Linked Open Data resources in order to provide deep understanding of the free-textual contents, which is envisaged to improve the performance of domain feature extraction task. Exploiting the domain knowledgebase for domain feature extraction helped to overcome the limitation of extracting domain features from textual reviews using the other approaches; which answers our research question RQ2 (*Can the domain knowledge improve the precision and recall of the feature extraction task?*). For example, the extracted domain features via Association Rule Mining tend to be frequent domain features, whereas infrequent domain features are ignored. In addition, some of the extracted nouns and noun phrases may not be domain features even if these occur more frequently in the textual contents. The developed domain knowledgebase-based Semantic Knowledgebase approach also improves on Machine Learning approach, where training datasets need to be manually annotated by human experts in order to deliver significant results, which can be an extremely time-consuming task as the required size of the training dataset should be sufficiently large to bootstrap the learning algorithms.

During our experiments, we observed that movie review contains opinions on the target movie and its features (movie name and names of stars, writers, editors, etc.), but sometimes can contain opinions about other movies. Hence, the extracted domain features by Semantic Knowledgebase approach from movie reviews might not necessarily be relevant to the target movie. The related state-of-the-art approaches have not considered eliminating such non-relevant domain features, which can reduce the precision of the extracted domain features. In this research, we addressed this challenge by investigating, where possible, each matched domain feature against the relevant semantically structured ground facts by performing SPARQL's ASK Queries over the developed movie-review knowledgebase, which was populated utilising Linked Open Data resources. The conducted evaluation showed that the accuracy of the domain feature extraction process was further improved by consulting the semantic knowledgebase to filter out irrelevant domain features; which answers our research question RQ3 (*How can the semantically structured public datasets be exploited to improve the performance of domain feature extraction task?*).

The domain feature extraction process has performed better with the produced semantic domain knowledgebase that has more comprehensive coverage than similar reported works. However, the characteristic of the problem domain (e.g. movie reviews) affected the performance of the domain feature extraction as we observed that reviewers tend to mention the full name of people (e.g. Spike Lee) at the first time of expressing opinions on them, and then only single names (e.g. Lee) or pronouns (e.g. s/he) are mentioned to express opinions. Therefore, Co-referencing resolution process was deployed to identify the orthographic and pronominal relations between the identified domain features and single names and pronouns to further identify non-explicit domain features. As the full names already matched by the semantic knowledgebase, their specification (i.e. the full name, their object relation, etc.) were inherited to the referred single names and pronouns. The conducted evaluation showed that the performance of domain feature extraction task was further improved after deploying co-reference resolution for non-explicit domain features; which answers our research question RQ4 (*Given the fact that the target domain feature is presented by a single name or pronoun (i.e. termed non-explicit domain features), how can the semantically constructed knowledgebase be utilised with co-reference resolution to extract non-explicit domain features to further improve the domain feature extraction task?*).

Chapter 5

5 Domain Feature-Sentiment Association

This chapter addresses two research questions RQ5 (*Can the domain's sentiment lexicon contribute to improve the domain feature-sentiment association task?*) and RQ6 (*Is the aggregation of the domain features' sentiment polarities based on Semantic Knowledgebase approach sufficient for the accurate classification of the review opinion?*) via illustrating details of the conducted domain feature-sentiment association process together with the experimental evaluation. The domain feature-sentiment association process is based on using sentiment lexicon. The main objective is to utilise a sentiment lexicon and generate from it sentiment lexicons for domain features in order to increase the clarity of the subjective information (i.e. opinions) and the descriptive information (i.e. facts), which is envisaged to improve the performance of domain feature-sentiment association task.

5.1 Related Work on Domain Feature-Sentiment Association

This section discusses related literature in opinion mining with a focus on methods for associating the extracted domain features with their corresponding sentiments.

Associating domain features with their corresponding sentiments based on identified adjective terms has been widely investigated. The authors in (Eirinaki, Pisal and Singh 2012, Hu and Liu 2004) associated the identified domain features with their corresponding sentiments that were tagged as adjectives and were very adjacent to them. The authors in (Penalver-Martinez, et al. 2014) extracted adjectives as sentiments, which were placed near to the extracted domain features by utilizing N_GRAM Before, N_GRAM After, N_GRAM Around and All Phrase methods. The researchers in (Yang, et al. 2015) applied an information entropy method to associate domain features with adjectives with respect to the degree of correlation between them.

Syntactic parsing techniques have been also used for associating domain features with their corresponding sentiments by identifying patterns that contain opinions (i.e. domain features and their corresponding sentiments). The authors in (Zhuang, Jing and Zhu 2006) applied the association via training Machine Learning classifier on manually labelled opinion phrases that can help in recognizing the dependency of grammar relations between domain features and sentiments. The authors in (Agarwal, et al. 2015b) associated the extracted domain features with their corresponding sentiments by training a Machine Learning model to identify the semantic information and relations between terms in a text, which were detected by utilising dependency parse tree.

Associating domain features with their corresponding sentiments based on the adjacent adjectives or syntactic parsing techniques have demonstrated a promising success. However, the adjacent adjectives or the sentiment terms within the syntactic patterns might not present any subjective opinions as users maybe describe factual information (i.e. descriptive opinions) about a domain feature as in “the American movie is my favourite”, which can affect the precision of the domain feature-sentiment association task.

5.2 Design and Implementation of Domain Feature-Sentiment Association Phase

This section introduces the architecture of the Domain Feature-Sentiment Association phase, in which a new algorithm is used in this phase to enhance the precision of the associating domain features with corresponding sentiments.

To improve the performance of the domain feature-sentiment association task, the false positive opinions (i.e. pairs of associated domain feature with corresponding sentiment) that objectively describe factual information (e.g. It was a horrific scene; first movie, American movie, etc.) have been removed using generated sentiment lexicons for each group of movie’s features.

Figure 5.1 illustrates the architecture of the Domain Feature-Sentiment Association phase, which comprises the following main components: Sentiment Extraction, Domain Feature-Sentiment Association, Features’ Sentiment Polarity and Knowledgebase Enrichment.

In brief, the Domain Feature-Sentiment Association phase extracts sentiments from the pre-processed reviews and associates them with the extracted domain features (i.e. were resulted from Domain Feature Extraction phase) as well as determining their sentiment polarity. Finally, inserting semantic information about the processed reviews into the domain knowledge.

The structured opinion mining related information resulting from the Domain Feature-Sentiment Association phase is going to be used for training Machine Learning opinion classifier through Multi-point Opinion Classification phase, which will be explained in details in the next chapter.

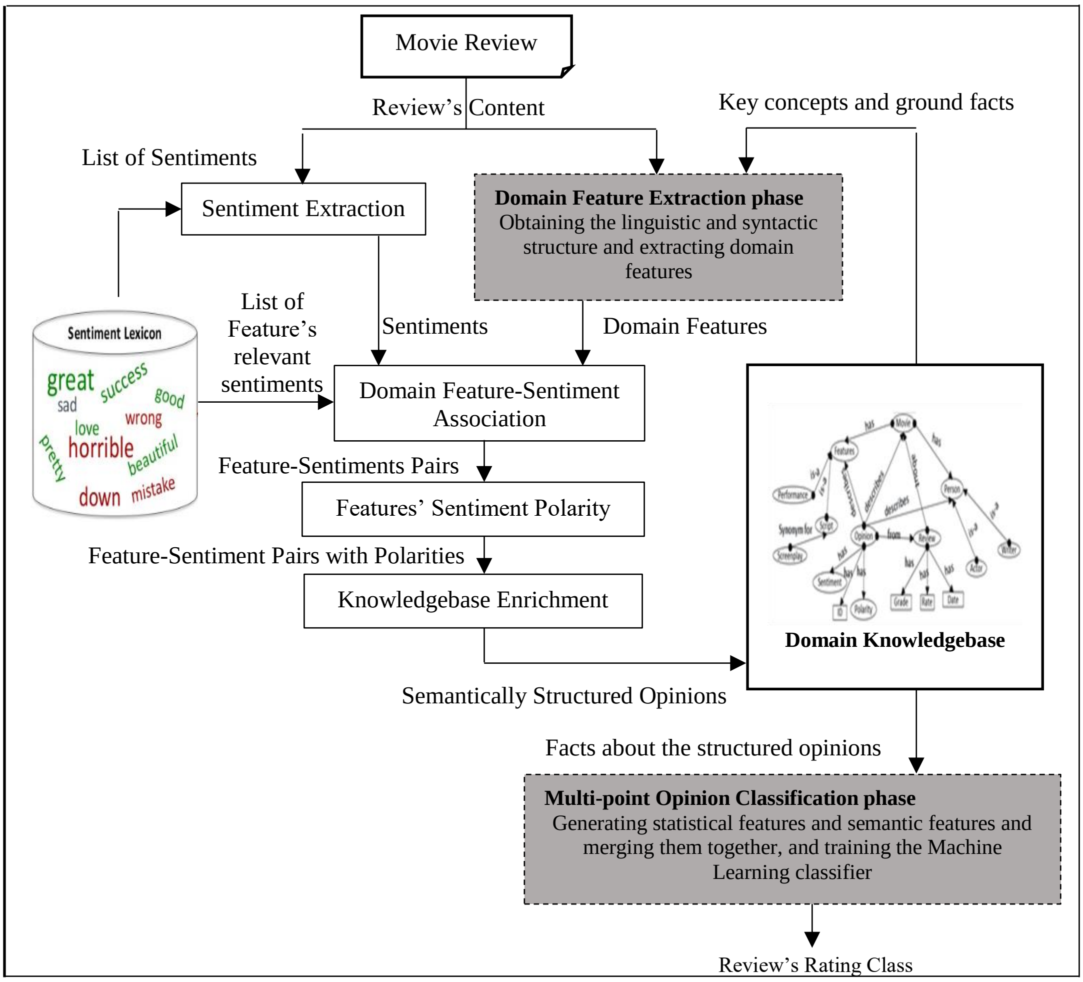
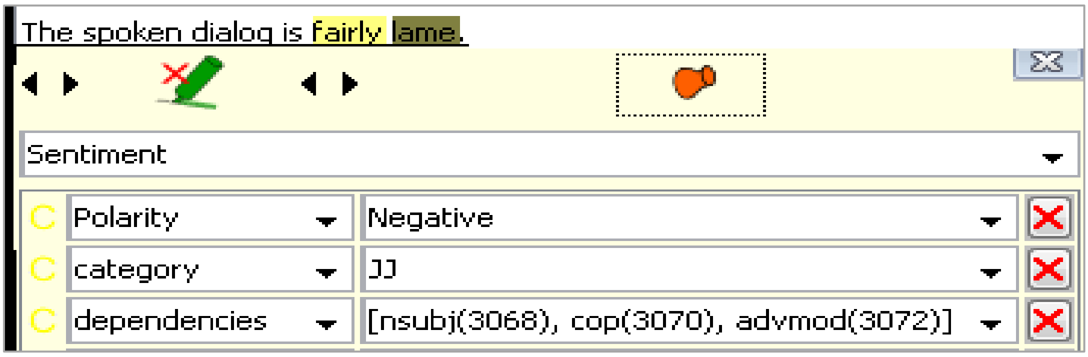

Movie Review
Review's Content
Key concepts and ground facts
List of Sentiments
Sentiment Extraction
Domain Feature Extraction phase
Obtaining the linguistic and syntactic structure and extracting domain features
Sentiment Lexicon
great
success
sad
good
love
wrong
horrible
pretty
beautiful
down
mistake
List of Feature's relevant sentiments
Sentiments
Domain Features
Domain Feature-Sentiment Association
Feature-Sentiments Pairs
Features' Sentiment Polarity
Feature-Sentiment Pairs with Polarities
Knowledgebase Enrichment
Domain Knowledgebase
Semantically Structured Opinions
Facts about the structured opinions
Multi-point Opinion Classification phase
Generating statistical features and semantic features and merging them together, and training the Machine Learning classifier
Review's Rating Class

Figure 5.1 The architecture of domain feature-sentiment association phase

5.2.1 Extracting Sentiments from the Processed Domain Reviews

Reviewers generally tend to express their opinions on domain features using various sentiments. These sentiments can be found as nouns, adjectives, and verbs. Hence, in opinion mining at domain feature level analysis, it is necessary to identify these sentiments within the review in order to be associated with their corresponding domain features, and then calculating domain features' polarities. In this process, sentiments are identified from the pre-processed reviews using a GATE's Gazetteer that we imported with a list of sentiments (around 2168 positive and 6270 negative) that were obtained from opinion lexicon[6]. The particular list has been used widely in many studies as in (Li and Liu 2012, Ma, et al. 2013). GATE's Gazetteer links between the imported sentiments and the root of each word in the pre-processed reviews. Figure 5.2 shows an example of annotated negative sentiment "lame".


The spoken dialog is fairly lame.

Sentiment

Polarity	Negative
category	JJ
dependencies	[nsubj(3068), cop(3070), advmod(3072)]

Figure 5.2 Example of annotated sentiment

Following the identification of sentiments, any adjacent shifters (negation or adverb) were taken into account to moderate the sentiment's score accordingly. For example, in the sentence "This is not a great movie", the shifter "not" is located nearby to the sentiment "great". Hence, the sentiment is modified to be "not great" with a score of -1. As shown in Figure 5.3, the modification process was performed using hand-crafted JAPE[7] (Java Annotation Patterns Engine) rules. JAPE are regular expression rules that are written via Java programming language to support annotating terms or patterns within a processed textual reviews.

[6] https://www.cs.uic.edu/~liub/FBS/sentiment-analysis.html#lexicon

[7] https://gate.ac.uk/sale/tao/splitch8.html

```
Phase: Modified-Sentiment
Input: Sentiment RB Token
// RB points to shifters, Token points to a word
Rule: Mo-Sentiment
( {RB} ({Token})? {Sentiment} ):label
-->
{ gate.AnnotationSet matchedAnns =
(gate.AnnotationSet)bindings.get(“label”);
gate.Annotation matchedA =
(gate.Annotation)matchedAnns.iterator().next();
outputAS.add(matchedAnns.firstNode(),matchedAnns.lastNode(),”Mod
ified-Sentiment”, newFeatures); }
```

Figure 5.3 Jape rules for associating sentiment with shifter

5.2.2 A Novel Domain Feature-Sentiment Association Algorithm

The domain feature-sentiment association task is performed using the proposed new Domain Feature-Sentiment Association algorithm, which is primarily driven by generating sentiment lexicons for domain features as described below:

Step1: Generating sentiment lexicons for domain features

Most opinion mining approaches involve using publically available sentiment lexicons (e.g. SentiWordNet) for the domain feature-sentiment association task. Some authors developed special sentiment lexicons for specific tasks. For example, the author in (Guarino 1998) developed a sentiment lexicon that contained sentiment terms as well as emoticons to be used for analysing Twitter messages. In our work, 6800 positive and negative sentiments were obtained from a public repository opinion lexicon[8], which has been used widely in many studies as in (Li and Liu 2012, Ma, et al. 2013). Then, a sentiment lexicon was generated for each domain feature that belongs to the chosen movie reviews domain. Each generated sentiment lexicon contains a list of sentiments that can be used only to express a subjective opinion for a specific domain feature. Different domain features may have a different list of sentiments. For example, the sentiment “horrific” in the sentence “It was a horrific scene” expresses a descriptive opinion on the domain feature “scene”, whereas, in the sentence “It was a horrific movie” expresses a subjective opinion on the domain feature “movie”.

[8] https://www.cs.uic.edu/~liub/FBS/sentiment-analysis.html#lexicon

Thus, for the “scene” feature, a sentiment lexicon was generated that did not contain the sentiment “horrific”, whereas it was included within a generated sentiment lexicon for the “movie” feature. Moreover, each list of sentiments can be applied to the same group of domain features. Hence, one sentiment lexicon was generated for each group of domain features that have the same classification. Table 5.1 shows an example of some movie’s features and their relevant sentiments. Column 1 indicates different groups of movie’s features, and column 2 indicates the relevant sentiments for each group.

Table 5.1 Example of grouped movie’s features and their relevant sentiment

Key Concepts and Associated Ground Facts	**Sentiments**
The concept “Movie” and movies’ names such as “Meet the Deedles”	Admirable, Undelivered, Horrific, Slow, Long
The concepts “Star, Writer, Editor, Director” and names of people who are stars, writer, etc.	Admirable, Able, Handsome, Gorgeous
The concepts “Writing, Screenplay, plot, script, story, idea”	Admirable, Undelivered, Well-Populated
The concept “Performance”	Admirable, Undelivered, Well, Well-Populated
The concepts “Special Effects, Visual Effects, Scene”	Admirable, Undelivered, Loud, Well-Crafted

Step 2: Associating domain features to sentiments

In this stage, the extracted filtered domain features are associated with their corresponding extracted sentiments (feature-sentiment pairs). In other words, in each review all the mentioned statements that contain sentiments about the domain features are identified. A new Domain Feature-Sentiment Association algorithm illustrates the association process in details.

Algorithm 3 **Domain Feature-Sentiment Association**

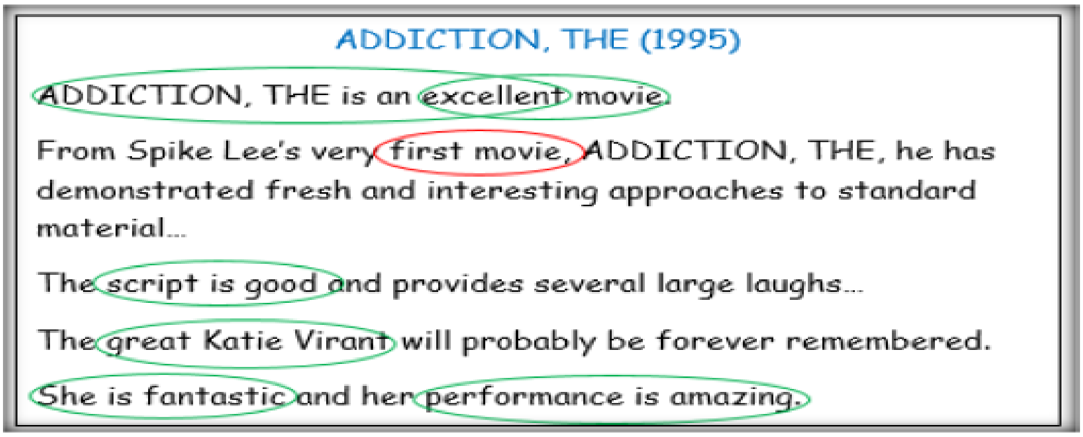
```
Input:
Pre-processed Reviews R, Extracted domain features F, Extracted Sentiments S, Sentiment Lexicons
SL for domain features
Do for i=1:R
   Sentences=IdentifySentence (Review[i])
   DependencyPatterns=IdentifyDependencyPattern(Sentences)
/* Identify Feature-Sentiment Pairs FSPs that contains both a  domain feature and a Sentiment*/
    FSPs=IdentifyFeature-Sentiment Pairs (DependencyPatterns, F,S)
    K=Count(DependencyPatterns)
    Do for j=1:K,
          If (DependencyPatterns[j] contains F and S)
```

```
            FSPs[j]=DependencyPatterns[j]
        Else
            Discard(DependencyPatterns[j])
        End if
    End for
/*Filtering Feature-Sentiment Pairs that present descriptive opinions
    K=Count(FSPs)
    Do for j=1:K,
        If (FSPs[j] contains S that is not listed within SL for F)
            Discard (FSPs[j])
        End if
    End for
End for
Output: Filtered Features-Sentiment Pairs FSPs
```

As illustrated in the above algorithm, the association process was performed via implementing dependency pattern rules (see Table 5.2), which is achieved via using the syntactical structure of the content. The identified patterns should contain both domain feature and sentiment such as “great script” and “the actor is good”. Then, the associated domain feature-sentiment pairs that hold descriptive statements were discarded using the generated sentiment lexicons for the domain features. For example, using the review example in Figure 5.4, the opinion phrase “first movie” represents a descriptive statement, hence, it is discarded. Other opinion phrases such as “excellent movie, the script is good, great Katie Virant, she is fantastic, performance is amazing” represent subjective statements, and because their domain features are associated with their sentiments they are retained.

Table 5.2 Dependency pattern rules

Dependency Relation	Pattern Rules	Example
Nsubj: a noun phrase which is the syntactic subject of a clause	Domain Feature(NN), Sentiment(JJ)	The movie is great
Dobj: the noun phrase which is the (accusative) object of the verb	Sentiment(V), Domain Feature(NN)	I hate this music
Prep-of + nn: Prepositional phrases followed by a noun	Sentiment(NN)+ “of”, Domain Feature(NN)	The beauty of the script
Amod: Adjectival phrase that serves to modify the meaning of the noun phrase	Sentiment(JJ) , Domain Feature(NN)	It is a nice script



ADDICTION, THE (1995)

ADDICTION, THE is an excellent movie.

From Spike Lee's very first movie, ADDICTION, THE, he has demonstrated fresh and interesting approaches to standard material...

The script is good and provides several large laughs...

The great Katie Virant will probably be forever remembered.

She is fantastic and her performance is amazing.

Figure 5.4 Example of descriptive and subjective statements

The experimentation results in section 5.3.3 demonstrate that analysing the subjectivity of opinion phrases improves the results obtained solely on dependency pattern rules for domain feature-sentiment association.

5.2.3 Features' Sentiment Polarity

In this process, the polarity of each extracted domain feature that has been associated with its sentiment in the previous stage is calculated using the sentiment aggregation function, which was adopted in various studies in the literature for calculating the polarity of domain features. The devised function assigns a score (weight) that indicates the proximity (distance) of the sentiment to the identified corresponding domain feature in the opinion phrase. Adopting sentiment aggregation function for domain features polarity is more effective than relying solely on syntactic dependencies that can indicate the right relation between a domain feature and a sentiment, but may not always yield accurate results, as the associated dependency patterns do not cover all the sentiments and shifters that express the opinion (Ding and Liu 2007). For example, in sentences "It is a great movie, however, it is not", "I do not think that this movie is great" and "I am not sure whether this movie is good or not", the dependency relations can be used to identify the underlined opinion phrases in order to associate domain features with their sentiments. However, the dependency relations cannot be used to accurately indicate the polarity score because they do not take into account the negation shifters.

The sentiment aggregation function as presented in Equation 5.1 is based on determining the final polarity score for each extracted domain feature {f_1,….,f_m) in a sentence *s* via aggregating the multiplicative inverse of the sentiment score *ss* of each extracted sentiment {$se_1,\ldots,se_n$} within the sentence *s* and the distance *dist* (se_j,f_i) between the extracted domain feature f_i and the extracted sentiment se_j. The score values +1 and -1 are assigned to positive and negative sentiments respectively. The domain feature f_i is assigned the final calculated score fcs_i as well as is assigned the polarity level (i.e. very positive, positive, neutral, negative, and very negative) using the condition below:

1. Very Positive: IF ($fcs_i > 0.5$ AND $fcs_i \leq 1$)
2. Positive: IF ($fcs_i > 0$ AND $fcs_i \leq 0.5$)
3. Neutral: IF ($fcs_i = 0$)
4. Negative: IF ($fcs_i > -0.5$ AND $fcs_i < 0$)
5. Very Negative: IF ($fcs_i \geq -1$ AND $fcs_i \leq -0.5$)

Equation 5.1

$$\text{Score}(f_i,\text{s})=\sum_{se_j \in s} \frac{se_j.ss}{dist(se_j,f_i)}$$

Dealing with negation terms or shifters such as not, no, never, none, nobody, nowhere and neither can be sometimes problematic when these shifters are mentioned without the following (succeeding) sentiments (Ding and Liu 2007). That is because there are not any fixed rules for them. Therefore, they were treated as sentiments by assigning them a negative score value -1 and counting their distance from the specified domain feature, then aggregate them with other scores. Whereas the score of each sentiment that is preceded by a shifter in case they are adjacent such as (not good) was shifted (+1 to -1 or -1 to +1). Then, the sentiment aggregation Equation 5.1 was applied.

5.2.4 Enriching the Semantic Knowledgebase

In this stage, the semantically structured movie-review knowledgebase that was used to bootstrap the domain feature extraction process is further enriched with new semantic information related to the analysed review and the corresponding extracted domain features. Firstly, the review ID and the name of reviewer who wrote the review were inserted into the movie-review knowledgebase. Secondly, new semantic relations are injected into the movie-review knowledgebase for each extracted domain feature that was associated with a sentiment.

Figure 5.5 illustrates a concept map for some of the injected semantic information into the movie-review knowledgebase, which is related to a review about THE ADDICTION movie. The labels in the concept map that contain “The Addition 1995”, “Katie Virant” and “Performance” indicate the movie domain’s key concepts and ground facts that were used to extract domain features, whereas, the rest of the labels indicate to the semantically-tagged information and relations about the analysed review and the extracted domain features such as the polarity level (i.e. very positive, positive, neutral, negative, very negative) of the extracted domain feature, and the sentiment term that was used to describe the domain feature.

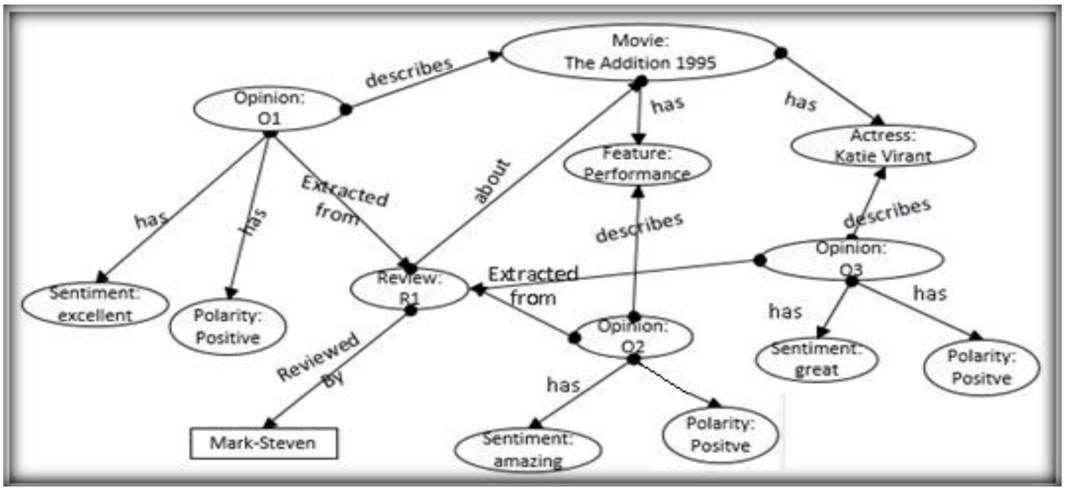


Figure 5.5 A concept map for the injected semantic information into the semantic knowledgebase

The resulting movie-review knowledgebase will be accumulatively enriched with the semantically annotated movie’s features and sentiments extracted from the review, and hence will represent a valuable resource not only for predicting general opinion about a movie, but also for sophisticated retrieval of opinions associated with a specific movie’s feature. For instance, the movie-review knowledgebase should be able to answer a query about movies with the favourable screenplay, filtered by a specific genre, actor, origin, etc.

5.3 Experimental Evaluation

This section presents the conducted experiments on a movie review dataset as a case study in order to evaluate the performance of the domain feature-sentiment association task.

5.3.1 Datasets

Cornell Movie Review Dataset[9] was used for the experiments, and this dataset has been widely used in the sentiment analysis literature (Mukras and Carroll 2004, Allison 2008, Li and Liu 2012). The dataset contains 1770 movie reviews and their corresponding numerical rating for 3-class classification [0, 1, and 2 — essentially "negative", "middling", and "positive", respectively] and for 4-class classification [0, 1, 2, and 3 — essentially "negative", "middling", "positive", and "very positive", respectively]. A total of 475 sentences containing 9301 words were selected from the downloaded dataset, and then from the selected sentences, 107 domain feature-sentiment pairs were manually extracted. The manually identified domain feature-sentiment pairs baseline were used to evaluate the obtained domain feature-sentiment pairs via the novel Domain Feature-Sentiment Association algorithm.

5.3.2 Experimentation Methodology

Using the domain feature extraction phase, sentiments were extracted from movie reviews and then modified to take into account any preceding shifters that might modify their scores. Thereafter, the filtered domain features were associated with their corresponding sentiments using the proposed Domain Feature-Sentiment Association algorithm, and then their polarities were counted. Further, the movie-review knowledgebase was expanded with the obtained new semantic information and relations that belong to the processed movie reviews and the extracted domain features from them.

[9] http://www.cs.cornell.edu/people/pabo/movie-review-data

5.3.3 Experimental Results of Domain Feature-Sentiment Association Task

In this experiment, the proposed Domain Feature-Sentiment Association algorithm was evaluated against feature-sentiment pairs baseline. As described in section 5.2.2, the proposed Domain Feature-Sentiment Association algorithm associates the extracted filtered domain features with their corresponding extracted sentiments (domain feature-sentiment pairs) using dependency pattern rules, which is similar to the approach published in (Agarwal, et al. 2015a, Agarwal, et al. 2015b).

The novelty of the Domain Feature-Sentiment Association algorithm is that it discards the associated domain feature-sentiment pairs that hold descriptive statements (e.g. horrific scene, first movie) using the generated sentiment lexicons for domain features, and it retains the associated domain feature-sentiment pairs that hold subjective statements (e.g. amazing performance, the beauty of the script). Hence, two experiments were performed on the same selected sentences from the downloaded reviews that contain the baseline of associated domain feature-sentiment pairs. In the first experiment, the domain features-sentiment pairs were obtained using dependency pattern rules and without performing the filtering process, whereas in the second experiment, the domain features-sentiment pairs were obtained using the proposed Domain Feature-Sentiment Association algorithm in which the dependency pattern rules are used and the filtering process is performed.

Equation 5.2 and Equation 5.3 were used to compute the Precision and Recall of the associated domain feature-sentiment pairs within the two experiments. In Equation 5.2 and Equation 5.3, DFSPs stands for Domain Feature-Sentiment Pairs.

Equation 5.2

$$\textbf{Precision=} \quad \frac{|\{\text{relevant DFSPs}\} \cap \{\text{retrieved DFSPs}\}|}{|\{\text{retrieved DFSPs}\}|}$$

Equation 5.3

$$\textbf{Recall}= \quad \frac{|\{\text{relevant DFSPs}\} \cap \{\text{retrieved DFSPs}\}|}{|\{\text{relevant DFSPs}\}|}$$

The results shown in Table 5.3 indicate that the domain feature-sentiment pairs associated by the proposed Domain Feature-Sentiment Association algorithm achieved the highest precision value (84%), whereas the associated domain feature-sentiment pairs using dependency pattern rules and without applying filtering process obtained a precision value of 51%. This is due to the fact that using dependency pattern rules results in associating all domain features with their corresponding sentiment whether they present descriptive or subjective opinion phrases, whereas in the proposed Domain Feature-Sentiment Association algorithm such descriptive opinion phrases were filtered using the generated sentiment lexicons for domain features.

Table 5.3 Precision of the domain features-sentiment association

Approach	**The Proposed Domain Feature-Sentiment Association algorithm**	**Dependency Pattern Rules**
Precision	84%	51%

This research also evaluated the advantages of utilising public Linked Open Data sources on the domain feature-sentiment association task. Hence, two experiments were carried out using the same selected sentences from the downloaded movie reviews that contain the baseline associated domain feature-sentiment pairs. The first experiment is based on evaluating the performance of the proposed Domain Feature-Sentiment Association algorithm when the associated domain features are the domain's key concepts and synonyms only (i.e. KB-EXP1). The second experiment is based on evaluating the performance of the proposed Domain Feature-Sentiment Association algorithm when the associated domain features are the domain's key concepts and synonyms in addition to the relevant ground facts that were gathered from Linked Open Data resources (i.e. KBLOD-EXP2).

The obtained results presented in Table 5.4 evidenced that the recall of KB-EXP1 and KBLOD-EXP2 experiments was 69% and 73% respectively, which indicates that the number of extracted opinion phrases (associated domain feature-sentiment pairs) was increased in KBLOD-EXP2 experiment. The improved Recall in the experiment KBLOD-EXP2 demonstrates the benefit of populating the movie-review knowledgebase with ground facts from Linked Open Data resources which increased the number of the matched domain features and subsequently the number of the extracted opinions. Therefore, it can be concluded that populating the domain knowledgebase using Linked Open Data resources can enhance both domain feature extraction and feature-sentiment association processes.

Table 5.4 Recall of domain feature-sentiment association

Experiment	EXP1	EXP2
Recall	69%	73%

5.3.4 Limitations of Domain Feature-Sentiment Association Task

Detailed analysis of the results were presented in our paper that published in International Conference on Knowledge Engineering and Applications (Alfrjani, Osman and Cosma 2017). The analysis revealed that there are some limitations in the output of the association mechanism that affected the performance of domain feature-sentiment association task. We attribute these limitations to the following contributing factors:

- Opinions expressed using the "If condition", for example, consider the narrative from a review "And arguably just as surprising, the good-spirited film is no comedy, even if it does have humorous moments". Here "humorous moments" is an opinion but it was not expressed in the movie (Narayanan, Liu and Choudhary 2009).
- Positive opinions that are rejected at the end of the sentence. For instance, "deeply philosophical movie, which it isn't" (González-Ibánez, Muresan and Wacholder 2011).
- Opinions that are expressed in question style as in "why they weren't given a decent script is the movie's real mystery" (Liu 2012).

Although the richer movie-review knowledgebase supported by Linked Open Data ground facts, as in experiment KBLOD-EXP2, did improve the Recall of domain feature-sentiment association task, there was a few number of false negatives; which happen when dealing with non-explicit sentiments, for instance, "the plot is not especially compelling, but the character interaction is, and that's the real reason to see this motion picture". The proposed Domain Feature-Sentiment

Association algorithm can extract the opinion “the plot is not especially compelling” but it cannot extract the second opinion “but the character interaction is” because the sentiment is not explicit (Huang, Wang and Chen 2017).

5.3.5 Investigating Opinion Classification Based on Feature-Level Sentiment Analysis

In this research, after the completion of enriching the movie-review knowledgebase with new semantic information related to the analysed review and the corresponding extracted domain features such as domain features’ sentiment polarities (as described in section 5.2.4), we deemed worthwhile to investigate whether the calculated features’ sentiment polarities are sufficient to perform opinion classification task on a multi-point scale without further analysis.

The evaluation was conducted on the downloaded dataset that contains 1770 movie reviews via retrieving from the developed movie-review knowledgebase the average domain features’ sentiment polarities for each processed movie reviews, which were then used to calculate the rating class via applying range of classification rules as demonstrated in Table 5.5.

Table 5.5 Classification rules for 2-class, 3-class and 4-class classification

Class	Classification Rules
2-Class	Rating Class = 0 : IF (Average Polarity ≤ -0.1) Rating Class = 1 : IF (Average Polarity >-0.1)
3-Class	Rating Class = 0 : IF (Average Polarity < -0.1 AND Average Polarity >= -1) Rating Class = 1 : IF (Average Polarity <= 0.3 AND Average Polarity >= -0.1) Rating Class = 2 : IF (Average Polarity <= 1 AND Average Polarity > 0.3)
4-Class	Rating Class = 0 : IF (Average Polarity < -0.5 AND Average Polarity >= -1) Rating Class = 1 : IF (Average Polarity <= 0 AND Average Polarity >= -0.5) Rating Class = 2 : IF (Average Polarity <= 0.5 AND Average Polarity > 0) Rating Class = 3 : IF (Average Polarity <= 1 AND Average Polarity > 0.5)

The obtained results were compared against the reviews’ numerical ratings on a scale of [0, and 1 — essentially “negative” and “positive” respectively], [0, 1, and 2 — essentially “negative”, “middling”, and “positive”, respectively] and [0, 1, 2, and 3 — essentially “negative”, “middling”, “positive”, and “very positive”, respectively] for 2-class, 3-class and 4-class classification respectively. Equation 5.4, was used to compute the Precision of the calculated reviews’ rating class

Equation 5.4

$$\textbf{Precision}= \frac{|\{\text{Number of reviews of correct calculated rating}\}|}{|\{\text{Total Number of All reviews}\}|}$$

Table 5.6, Table 5.7 and Table 5.8 presents the obtained results for 2-class, 3-class and 4-class classification respectively. The results indicate that classifying reviews using classification rules worked quite well for 2-class classification only with an average 77.4%, whereas, the results were not satisfied for 3-class and 4-class classification with an average 46.3% and 43% respectively.

Table 5.6 Results of 2-class classification using classification rules

Rating Class	**0**	**1**	**Average**
Precision	30.2 %	91.7 %	77.4 %

Table 5.7 Results of 3-class classification using classification rules

Rating Class	**0**	**1**	**2**	**Average**
Precision	30.2 %	64.3 %	39.3 %	46.3 %

Table 5.8 Results of 4-class classification using classification rules

Rating Class	**0**	**1**	**2**	**3**	**Average**
Precision	3.6 %	34 %	72 %	12 %	**43 %**

It is clear that performing opinion classification by aggregating the sentiment polarities of the extracted domain features is not sufficient to consistently get accurate results across all variations. Therefore, we further involved Machine Learning approach for performing opinion classification on a multi-point scale as described in the next chapter.

5.4 Discussion

In this chapter, the Domain Feature-Sentiment Association phase was explained, which relies on utilising domain sentiment lexicons to improve the precision of the associated domain features with their corresponding sentiments via a new Domain Feature-Sentiment Association algorithm. Domain features that are extracted from a textual content might not be have any subjective opinions about them as users maybe describe factual information about the extracted domain features as in “the American movie is my favourite”. Discarding subjective opinions is still challenging for many researchers. In this research, utilising the domain knowledge to create domain’s sentiment lexicon enabled us to eliminate descriptive opinions, and hence to improve up on the state of the art related works that used syntactic parsing techniques (i.e. identify both descriptive and subjective phrases). The generated sentiment lexicon for each group of domain features contains a list of sentiments that can be used only to express subjective opinions for a specific group of domain features. The experimental results demonstrated that the proposed algorithm improved the performance of domain feature-sentiment association task; which answers our research question RQ5 (*Can the domain’s sentiment lexicon contribute to improve the domain feature-sentiment association task?*). However, our analysis of the results revealed that there are some limitations in the output of the association mechanism that affected the performance of domain feature-sentiment association task.

In this research, after the completion of enriching the domain knowledgebase with new semantic information related to the analysed review and the corresponding semantically annotated movie’s features and their corresponding sentiments as well as their polarities, we deemed worthwhile to investigate whether the calculated features’ sentiment polarities are sufficient to perform opinion classification task on a multi-point scale without further analysis. The classification accuracy in the obtained results was not satisfactory as the results indicated that classifying reviews using classification rules worked quite well for 2-class classification only with an average 77.4%, whereas, the results were not satisfied for 3-class and 4-class classification with an average 46.3% and 43% respectively; which answers our research question RQ6 (*Is the aggregation of the domain features’ sentiment polarities based on Semantic Knowledgebase approach sufficient for the accurate classification of the review opinion?*). Therefore, we will further involve Machine Learning approach for performing opinion classification on a multi-point as described in the next chapter.

Chapter 6

6 Multi-point Opinion Classification based on a Hybrid Semantic Knowledgebase-Machine Learning Approach

This chapter addresses the final research question RQ7 (*How can we use Semantic Knowledgebase approach to improve the quality of training features that are then used to build a Machine Learning classifier in order to improve the accuracy of opinion classification on a multi-point scale?*) via discussing the conducted Multi-point Opinion Classification process alongside the experimental evaluation. The multi-point opinion classification process is based on integrating Semantic Knowledgebase approach with Machine Learning approach. The main objective is to determine whether adding additional semantic features (e.g. number of extracted domain features, frequency of the sentiments that were associated with domain features per review, average polarity of each group of domain features, etc.) to a training dataset of statistical features (e.g. frequency of the refined terms in textual reviews) can improve the performance of opinion classification task on a multi-point scale, i.e. solving the rating inference problem on a multi-point scale.

6.1 Related Work on Opinion Classification

This section discusses related literature in opinion classification with a focus on methods for classifying opinions on a multi-point scale.

Opinion classification is the process of classifying the opinions into a binary classification (i.e. whether it is a positive or negative), or a multi-point scale (i.e. classify the polarity of the content at fine-grained level) such as very negative, negative, neutral, positive and very positive (Pang and Lee 2005). The problem of classifying opinions using a multi-point scale (also referred to as the rating inference problem) has been an interesting research area in recent years. Early published research focused on binary classification of the overall polarity of the opinion (Moraes, Valiati and Neto 2013, Poobana and Sashi Rekha 2015). The obtained results of such studies indicated that Machine Learning algorithms outperformed humans on the task of binary classification of opinions (Sebastiani 2002).

More recently, researchers have focused on classifying opinions on multi-point scale rating using Machine Learning algorithms in particular supervised learning algorithms (Vapnik 2013). In general, these approaches are based on training a classifier on a dataset of features that have been extracted from textual contents and the corresponding target outputs (i.e. numeric rating). Then, the built classifier is tested on a dataset of features without the target outputs. Finally, the obtained outputs are compared against the real target outputs in order to evaluate the classifier (Pang and Lee 2008, Prabowo and Thelwall 2009, Tang, Tan and Cheng 2009). Various techniques have been developed to improve the accuracy of the classifier's results as well as decrease the dimensionality of the dataset. The authors in (Lunardi, et al. 2016) proposed an approach for multiclass classification that is based on using Nested dichotomies algorithm to perform successive stages of binary classification processes. The effort in (Asghar 2016) resulted in various multi-class classifiers based on a combination of four types of extracted features (unigrams, bigrams, trigrams and latent semantic indexing) with four types of Machine Learning algorithms which are: Naïve Bayes, Perceptron Neural Networks, Logistic Regression and Linear Support Vector Classifier. The study in (Acampora and Cosma 2015) introduced an innovative computational intelligence framework to predict customer opinions rating, which is based on using Information Retrieval approaches to extract features and then using an integration of Singular Value Decomposition, Dimensionality Reduction, Genetic algorithms and different fuzzy algorithms for opinion classification on a multi-point scale rating. The same authors have presented their updated framework via applying fuzzy C-Means and the adaptive neuro-fuzzy inference algorithms for opinion classification on a multi-point scale rating (Cosma and Acampora 2016).

Until recently, Machine Learning approaches have been commonly applied for the process of opinion classification and are known to deliver outstanding performance, especially when they are trained using an effective dataset of features that have been manually annotated by a human expert who tend to enhance the annotation process with domain background knowledge. However, this can be an extremely time-consuming task as the required size of the training dataset should be sufficiently large to bootstrap the learning algorithms. The authors in (König and Brill 2006, Joshi and Penstein-Rosé 2009, Wu, et al. 2009, Nakagawa, Inui and Kurohashi 2010) stated that a successful improvement of the classified opinions can be achieved when different approaches are combined together. Therefore, a hybrid approach has emerged as an effective approach for enhancing the quality of the used dataset of features to train the classifier in order to improve the opinion classification task.

The study in (Sacharin, Schlegel and Scherer 2012) integrated different approaches to improve the opinion classification task, they utilised Natural Language techniques to process the contents in terms of removing errors, lemmatizing terms, tagging terms with their part of speech tag, then they used Lexicon-Based approach to identify sentiment terms, adverbs, negations and emoticons as a training features. In addition, they used Association Rule Mining approach to modify the score of the identified sentiment terms as well as the identified emoticons. Finally, they used Machine Learning approach to build a classifier based on these created features. The authors in (Vilares, Alonso and Gómez-Rodríguez 2013) classified the opinion of Spanish tweets via a hybrid approach that integrates Machine Learning and Linguistic Knowledge where they trained a supervised classifier on part of speech tags, semantic knowledge and syntactic dependencies features, which were obtained by means of Natural Language techniques. The authors in (MartíN-Valdivia, et al. 2013) developed a combined approach for opinion classification that is based on using SentiWordNet lexicon (Baccianella, Esuli and Sebastiani 2010) to extract features from Spanish movie reviews and using them as a training dataset features for Machine Learning classifier. The work in (Marchand, et al. 2013) was aimed to classify the polarity of tweets contents via using sentiment lexicon to extract the frequent sentiment terms from the pre-processed contents as features for Machine Learning classifier such as Support Vector Machine. The authors in (Balage Filho and Pardo 2013) presented a hybrid approach for opinion classification that is based on implementing a system that extracts the best features from the contents using Association Rule Mining and Lexicon-Based approaches, which are then are used to train a Machine Learning classifier. The researches in (MartíN-Valdivia, et al. 2013, Poria, et al. 2014) have focused on dealing with the ambiguity of the classified contents via combining Lexicon-Based and Machine Learning approaches. In (Roncal and Urizar 2014), for the purpose of opinion classification, a polarity lexicon was used to extract features from the contents, and then a Support Vector Machine classifier was built based on the extracted features. The presented work in (Baca-Gomez, et al. 2016) is based on combining Machine Learning and Lexicon-Based approaches to classify the polarity of Mexican Spanish social media comments such as twitter. They trained a Sequential Minimal Optimization classifier on a dataset of features that were generated using an affective lexicon. The created features are positive emoticons, negative emoticons, negations, adverbs and frequency of very positive, positive, very negative and negative sentiments. They evaluated their method on classifying the contents at different rating inference scales, which are 3-class (positive, neutral, negative) and 5-class (very positive, positive, neutral, negative, very negative). The best obtained results were when the contents are classified at 3-class classifications. In (Tan and Na 2017), the study focused on generating semantic features using Semantic Parsing and Class Association Rule Mining approaches to build a Machine Learning classifier in order to improve the opinion classification task.

The Semantic Knowledgebase approach uses a knowledgebase that represents a shared understanding of the domain of interest, hence, the Semantic Knowledgebase approach can be used to enrich a dataset with semantic features, which can improve the performance of opinion classification task. Semantic Knowledgebase approaches rely mainly on capturing the knowledge background of a chosen domain in order to extract the domain features from reviews. These domain features are then utilised to build a Machine Learning classifier in order to classify the overall opinion of the reviews as positive or negative as in (Polpinij and Ghose 2008, Sulthana and Subburaj 2016). However, there appear to be no studies that investigate the use of Semantic Knowledgebase approaches to produce dataset of semantic features that are then used to build a Machine Learning classifier to classify the opinions on multi-point rating scale.

6.2 Design and Implementation of Multi-point Opinion Classification Phase

This section discusses a novel approach to multi-point opinion classification that is based on a new Opinion Classification algorithm, which builds the Machine Learning classifiers using a combined training dataset of semantic and statistical features.

To improve the performance of multi-point opinion classification, firstly the enriched semantic movie-review is used to retrieve semantic features about the semantically structured opinions (i.e. extracted from the pre-processed reviews). Thereafter, Vector Space Model is deployed to generate statistical features (i.e. contains the frequency of the refined terms in the analysed reviews), which is widely used in the Information Retrieval field (Gravano, García-Molina and Tomasic 1999). Finally, the semantic and statistical features are combined and used to train a Machine Learning classifier and resulting the rating class of the analysed reviews. Figure 6.1 illustrates the architecture of the Multi-point Opinion Classification phase, which comprises the following main components: Generating Semantic Features, Generating Statistical Feature and Training Machine Learning Classifier.

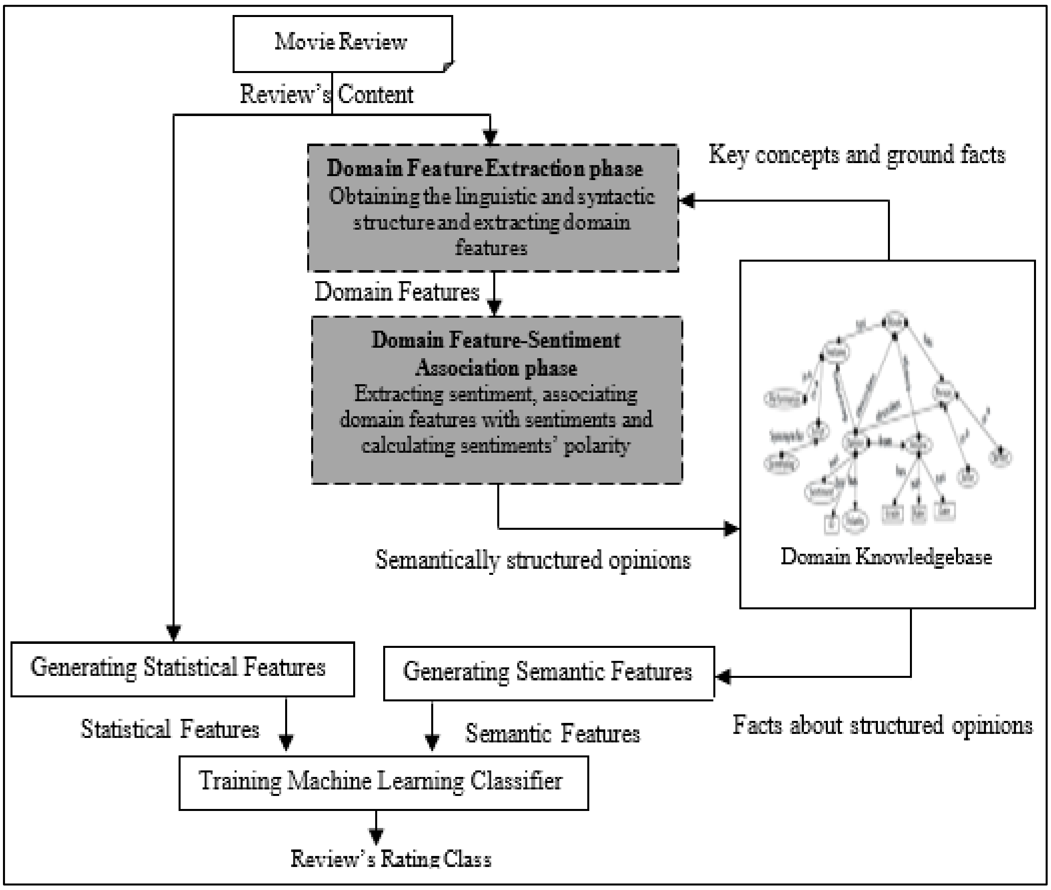

Movie Review
Review's Content
Domain Feature Extraction phase
Obtaining the linguistic and syntactic structure and extracting domain features
Key concepts and ground facts
Domain Features
Domain Feature-Sentiment Association phase
Extracting sentiment, associating domain features with sentiments and calculating sentiments' polarity
Semantically structured opinions
Domain Knowledgebase
Generating Statistical Features
Generating Semantic Features
Facts about structured opinions
Statistical Features
Semantic Features
Training Machine Learning Classifier
Review's Rating Class

Figure 6.1 The architecture of the multi-point opinion classification phase

The proposed Opinion Classification algorithm listed below explains the detailed account of the role of each component of the Multi-point Opinion Classification phase.

Algorithm 4 **Opinion Classification**

```
Input:
Number of Reviews N, List of Reviews' unique identity ID, movie-review Knowledgebase contains key
concepts, synonyms, ground facts and semantic information
----------------------------------------------------------------------------------------------------
                        /* Generating Semantic Features*/
Do for i=1: N,
    /* Number of Extracted Domain features (NEDF)*/
  NEDF= SumDomainFeatureQuery(ID[i], movie-review Knowledgebase)
    /* Number of Positive Sentiments (NPS)*/
  NPS= SumPositiveSentimentsQuery (ID[i], movie-review Knowledgebase)
    /* Number of Negative Sentiments (NNS)*/
  NNS= SumNegativeSentimentsQuery (ID[i], movie-review Knowledgebase)
    /* Frequency of the associated Sentiments (FS)*/
  FS= FrequencySentimnetQuery(ID[i], movie-review Knowledgebase)
    /* Average Polarity for each Group of Domain features (APGDF)*/
  APGDF= AVGPolarityQuery(ID[i], movie-review Knowledgebase)
    /* Insert all the semantic value into a matrix*/
Matrix F= Insert(NEDF, NPS,NNS,FS,APGDF)
 End for
----------------------------------------------------------------------------------------------------
                        /* Generating Statistical Features*/
Do for i=1: N,
ListofTokenisedTerms[i]= Tokenise(Review[i])
ListofFilteredTerms[i]= Filter (ListofTokenisedTerms[i])
ListofStemmedTerms[i]=Stemm(ListofFilteredTerms[i])
End for
Matrix S= CreatingVectorSpaceModel(ListofStemmedTerms)
----------------------------------------------------------------------------------------------------
                        /*Training Machine Learning Classifier */
Matrix FS= Merge(Matrix F, Matrix S)
NormalisedData= Normalise(Matrix FS)
(TrainingData, TestingData)= Spli(NormalisedData)
ClassiferModel= Train(TrainingData, TargetRating)
Reviews-Rating= Test(ClassifierModel, TestingData)
Output: Reviews' Rating
```

6.2.1 Generating Semantic Features

The generated semantic features represent facts of the semantically structured opinions such as number of extracted domain features. Let mxn be a semantic feature by review matrix F_{mxn}= [f_{ij}] where each row *i* holds a semantic value about the extracted domain features from textual reviews, and each column *j* represents a textual review. Hence, each cell f_{ij} of matrix F contains a semantic value at which a domain feature *i* appear in a review *j*. The semantic values contained in matrix F were retrieved from the enriched movie-review knowledgebase, in which each semantic value presents a specific type of information as follows:

1: Number of extracted Domain Features per a review (NDF).

2: Number of Positive Sentiments mentioned in the review (NPS).

3: Number of Negative Sentiments mentioned in the review (NNS).

4: Frequency of each Sentiments that were Associated with Domain Features per review (FSADF).

5: Average Polarity of each group of domain features (AvgP- *i*), for example, there will be an average polarity value = 1 for a grouped domain feature *i* in a review *j* when the grouped domain features *i* (e.g. script, story, screenplay) were extracted from a review *j* and associated with their corresponding sentiments (e.g. "the beauty of the script", "lovely story", "the screenplay was fantastic"), and their calculated polarity values are +1, +1 and +1.

Although the polarity value for each extracted domain feature can be obtained via running a query on each domain feature individually, a single query was performed on each group of domain features. Grouping the domain features is based on the structure of the modelled domain key concepts in the movie-review knowledgebase. For example, the movie's features "staring, writer, editor, etc." are specified as a person, hence, instead of performing an individual query for each of them, one query (as shown in Figure 6.2) is applied for these movie's features in order to combine their polarities and derive the average value.

The aim of grouping the polarity value of domain features is to reduce the number of zeros values in the matrix as a technique for improving the quality of matrices passed into the classifier. Prior to grouping the polarity value of domain features, the matrices were Sparse, meaning that most of their elements were zero values. Users often express their opinions on certain domain features and focus less on other domain features and this resulted in Sparse matrices. Sparse training matrices can have impact on the performance of the Machine Learning classifier because they do not contain sufficient data for training the classifier (i.e. have many zeros). Hence, minimising the zero values would improve the quality of the training data and as a consequence will improve the performance of the classifier (Xu, et al. 2017).

The value of the average polarity is presented as a fuzzy value using the conditions below, where the maximum average polarity is 1 and the minimum average polarity is -1 due to the fact that the score of sentiments (i.e. associated with domain features) is 1 or -1 for positive and negative sentiments respectively.

- 1 → Strongly Negative: IF (polarity ≥ -1 AND polarity ≤ -0.5)
- 2 → Negative: IF (polarity > -0.5 AND polarity < 0)
- 3 → Neutral: IF (polarity = 0)
- 4 → Positive: IF (polarity > 0 AND polarity ≤ 0.5)
- 5 → Strongly Positive: IF (polarity > 0.5 AND polarity ≤ 1)

```
Prefix owl:<http://www.movie-review-ontology.owl#>
PREFIX rdfs:<http://www.w3.org/2000/01/rdf-schema#>
PREFIX rdf:<http://www.w3.org/1999/02/22-rdf-syntax-ns#>

SELECT     ?review  AVG(?polarity)

WHERE
{
?review owl:hasOpinion ?opinion .
?opinion rdf:type owl:Opinion .
?opinion owl:describesFeature ?K .
{?K  rdf:type owl:Writer } UNION
{?K  rdf:type owl:Editor } UNION
{?K  rdf:type owl:Staring } UNION
{?K  rdf:type owl:Director } UNION
{?K  rdf:type owl: Cinematographer } .
?opinion owl:hasPolarityValue ?polarity .
}
GROUP BY ?review
```

Figure 6.2 Example of querying the average polarity of a group of domain features

Table 6.1 presents the generated semantic feature matrix from few examples of movie reviews that are listed below. The matrix presents the total number of extracted domain features, positive sentiments and negative sentiments; the frequency of sentiments that are used to express subjective opinions in each review; and the average polarity of each group of domain features (e.g. story and script are grouped together).

- Review1: I liked this movie … The beauty of the script… horrific scene.
- Review2: This movie is great ….the performance is amazing.
- Review3: I hate this movie … the performance is very bad.

- Review4: The story is not great … the acting is amazing.
- Review5: This is a nice film … the acting is great.

Table 6.1 Example of a generated semantic features matrix

Review Feature	Review1	Review2	Review3	Review4	Review5
NDF	3	2	2	2	2
NPS	2	2	0	1	2
NNS	1	0	2	1	0
FSADF-like	1	0	0	0	0
FSADF-beauty	1	0	0	0	0
FSADF-great	0	1	0	1	1
FSADF-amazing	0	1	0	1	0
FSADF-hate	0	0	1	0	0
FSADF -bad	0	0	1	0	0
FSADF -nice	0	0	0	0	1
AvgP-Movie	5	5	1	0	0
AvgP-Performance	0	5	1	5	5
AvgP-Script	5	0	0	1	0

6.2.2 Generating Statistical Features

The generated statistical features represent the frequency of the refined terms in textual reviews. Let mxn be a statistical feature by review matrix $S_{mxn}= [s_{ij}]$ where each row *i* holds the frequency of the refined term in textual reviews, and each column *j* represents a textual review. Hence, each cell s_{ij} of S contains the frequency value (i.e. 0 for the absence or 1 for the presence) at which a term *i* appears in a review *j*. The statistical values contained in Matrix S were generated from the textual reviews via : (1) tokenising each review's contents into list of tokens, (2) filtering the list of tokens by removing stop words, punctuations marks, semicolons, colons, numbers, tokens with length equal to one, tokens contain numbers and tokens that occur in only one review,

(3) stemming the list of filtered tokens by formatting each token to its root and converting each token to lowercase letters, and (4) creating a Vector Space Model that represents the frequency of each refined token across all reviews. Vector Space Model is an algebraic model for representing text documents as vectors of identifiers (e.g. index terms), which was developed by the authors in (Salton, Wong and Yang 1975). Table 6.2 presents the generated statistical feature matrix from few examples of movie reviews that are listed below. The matrix presents the frequency value of the refined terms that occur in more than one review.

- Review1: I liked this movie … The beauty of the script… horrific scene.
- Review2: This movie is great ….the performance is amazing.
- Review3: I hate this movie … the performance is very bad.
- Review4: The story is not great … the acting is amazing.
- Review5: This is a nice film … the acting is great.

Table 6.2 Example of a generated statistical features matrix

Review / **Term**	**Review1**	**Review2**	**Review3**	**Review4**	**Review5**
movie	1	1	1	0	0
great	0	1	0	1	1
performance	0	1	1	0	0
amazing	0	1	0	1	0
acting	0	0	0	1	1

6.2.3 Training the Machine Learning Classifier

The matrix F and S are merged together to produce a new matrix FS, which is then normalised by deploying feature scaling (i.e. each column) and instance scaling (i.e. each raw) and passed to Machine Learning classifiers such as Support Vector Machine and Naïve Bayes in order to result the rating inference for each review.

Support Vector Machine that was first introduced by the authors in (Vapnik 1995, Vapnik 1998b, Vapnik 1998a) is a supervised learning algorithm that is used with kernel functions to classify linear and nonlinear data. Support Vector Machine is useful for finding the best surface to separate the negative samples from the positive samples. Support Vector Machine is very productive in review classification compared to other classifiers such as Naïve Bayes and Maximum Entropy. Support Vector Machine is used to solve opinion binary classification task with fewer classification errors via finding the best decision boundary between classes that has the maximum margin hyperplane. When passing the training dataset to Support Vector Machine, a classifier for this data set is generated, which it is used to conclude facts of the provided testing data. Support Vector Machine is also used to solve opinion multi-class classification problem using one-against-all or one-against-one approaches.

Naïve Bayes that was first introduced by the authors in (Domingos and Pazzani 1997) is a supervised learning algorithm that is based on performing Bayes' theorem with the "naive" assumption of independence between features (i.e. each pair of features). Naive Bayes classifiers are highly scalable, requiring a number of parameters linear in the number of variables (features/predictors) in a learning problem. Naive Bayes classifier assigns a new observation to the most probable class, assuming the features are conditionally independent given the class value. Despite its simplicity, Naïve Bayes can classify the data faster than other sophisticated classifiers.

6.3 Experimental Evaluation

This section presents the conducted experiments on a movie review dataset as a case study in order to evaluate the performance of the proposed Hybrid Semantic Knowledgebase-Machine Learning approach for improving the performance of opinion classification on a multi-point scale. In particular, evaluating whether adding additional semantic features to a dataset of statistical features that are then used to build a classifier can result in higher classification accuracy or not.

6.3.1 Datasets

Cornell Movie Review Dataset[10] was used for the experiments, and this dataset has been widely used in the sentiment analysis literature (Mukras and Carroll 2004, Allison 2008, Li and Liu 2012). The dataset contains 1770 movie reviews and their corresponding numerical rating for 3-class classification [0, 1, and 2 — essentially “negative”, “middling”, and “positive”, respectively] and for 4-class classification [0, 1, 2, and 3 — essentially “negative”, “middling”, “positive”, and “very positive”, respectively].

Table 6.3 presents the characteristics of the chosen dataset. The numerical ratings of the chosen dataset will be used as reviews’ rating baseline to evaluate the obtained reviews’ rating via the proposed via the proposed Hybrid Semantic Knowledgebase-Machine Learning approach.

Table 6.3 Dataset characteristics

	4-class classification	**3-class classification**
Rating	**Count**	**Count**
0	191	413
1	526	648
2	766	709
3	287	-
Total	1770	

6.3.2 Experimentation Methodology

The semantic features were generated from the semantically constructed movie-review knowledgebase that had been enriched with the obtained new semantic information and relations, which belong to the processed movie reviews. Then, the semantic features were merged with the statistical features that were generated via standard Vector Space Model. Finally, the new Semantic-Statistical features were normalized by deploying feature scaling (i.e. each column) and instance scaling (i.e. each raw).

After that, the normalized Semantic-Statistical features were used to build different Machine Learning classifiers such as Support Vector Machine, Naive Bayes, K-Nearest Neighbors, Random Forest, etc. to classify each review at numerical rating

[10] http://www.cs.cornell.edu/people/pabo/movie-review-data

scale. The best results were obtained by Support Vector Machine and Naïve Bayes classifiers (Samal, Behera and Panda 2017). The one-against-all approach is used to build a Support Vector Machine and Naïve Bayes multi-classifier model. The process is based on building K binary classifiers, where K represents the number of classes. Then, training each cth binary classifier on all samples that are related to the cth class to be positive labels, and the negative labels are the rest of samples that are belong to the remaining classes. Finally, the cross validation technique was applied on each of the built Support Vector Machine and Naïve Bayes model to find the best kernel function and parameters for them. In this experiment, both classifiers were tuned using the linear kernel function because the best results were obtained when using the linear kernel function.

6.3.3 Experimental Results of the Opinion Classification Task

For the evaluation, the reviews were classified using three (Statistical, Semantic, and Statistical-Semantic) datasets. The Statistical dataset is generated using standard Vector Space Model; it contains the frequency number of each extracted word per review, which is computed by assigning zero for the absence of the word and one for the presence of the word. The Semantic dataset contains the valuable semantic information about the extracted domain features, which was retrieved from the enriched movie-review knowledgebase. The Statistical-Semantic dataset is a result of merging the Statistical and Semantic datasets.

Each dataset was input into the Support Vector Machine and Naïve Bayes classifiers, and classification performance was evaluated for 3-class and 4-class classification task. The obtained results were compared against the reviews' numerical ratings on a scale of [0, 1, and 2 — essentially "negative", "middling", and "positive", respectively] and [0, 1, 2, and 3 — essentially "negative", "middling", "positive", and "very positive", respectively] for 3-class and 4-class classification respectively.

Equation 6.1, Equation 6.2 and Equation 6.3 were used to compute Precision, Recall and F-measure respectively for evaluating classification performance.

Equation 6.1

$$\textbf{Precision}= \frac{|\{\text{relevant reviews}\} \cap \{\text{retrieved reviews}\}|}{|\{\text{retrieved reviews}\}|}$$

Equation 6.2

$$\textbf{Recall}= \frac{|\{\text{relevant reviews}\} \cap \{\text{retrieved reviews}\}|}{|\{\text{relevant reviews}\}|}$$

Equation 6.3

$$\textbf{F-measure}= \frac{2 * \text{Precision} * \text{Recall}}{\text{Precision} + \text{Recall}}$$

Table 6.4 and Table 6.5 present the obtained results from two classifiers Support Vector Machine and Naïve Bayes using the three datasets (Statistical, Semantic, and Statistical-Semantic) for 3-class and 4-class classification respectively. The results indicate that the performance of both classifiers improved when they were trained using the Statistical-Semantic dataset as opposed to using the other datasets.

Table 6.4 Results of 3-class classification task

3-Class Classification One V One method - Cross Validation K=10										
Classifier	**Rating Class**	**Statistical Dataset**			**Semantic Dataset**			**Statistical-Semantic Dataset**		
		P	**R**	**F**	**P**	**R**	**F**	**P**	**R**	**F**
Support Vector Machine	0	74	54	63	72	38	50	75	54	63
	1	57	65	61	51	67	58	59	69	64
	2	75	77	76	72	70	71	78	79	79
	Average	**68**	**67**	**67**	64	62	61	**71**	**70**	**70**
Naïve Bayes	0	67	67	67	62	47	53	70	66	68
	1	59	62	60	51	63	56	60	67	64
	2	75	71	73	72	67	70	79	73	76
	Average	**67**	**67**	**67**	62	61	61	**70**	**69**	**70**

Table 6.5 Results of 4-class classification

4-Class Classification One V One method, Cross Validation K=10										
Classifier	**Rating Class**	**Statistical Dataset**			**Semantic Dataset**			**Statistical-Semantic Dataset**		
		P	**R**	**F**	**P**	**R**	**F**	**P**	**R**	**F**
Support Vector Machine	0	56	42	48	54	26	35	80	23	35
	1	57	55	56	53	51	52	56	59	57
	2	64	75	69	57	75	65	62	82	71
	3	70	52	59	66	39	49	83	41	55
	Average	**62**	**62**	**62**	57	57	55	**65**	62	60

Naïve Bayes	0	66	49	56	52	22	31	68	44	53
	1	54	65	59	51	49	50	56	70	62
	2	67	59	62	56	74	64	70	63	66
	3	56	61	59	65	36	46	61	61	61
	Average	**61**	**60**	**60**	55	55	53	**64**	**63**	**63**

Table 6.6 and Table 6.7 present the accuracy of the classified 1770 reviews by Support Vector Machine and Naïve Bayes classifiers for 3-class and 4-class classification for the three datasets with respect to the number of features for each dataset, which are 1322, 716 and 2038 for the Statistical, Semantic and Statistical-Semantic datasets respectively. Comparing the results across the various datasets when using the Support Vector Machine and Naïve Bayes classifiers, maximum classification accuracy was consistently achieved by the Support Vector Machine classifier for 3-class classification. In particular, accuracy using Support Vector Machine was 0.5%, 0.7%, and 0.4% higher for the Statistical, Semantic, and Statistical-Semantic datasets respectively, when using the Support Vector Machine as opposed to when using the Naïve Bayes classifier. With respect to for 4-class classification, accuracy using Support Vector Machine was 2.1% and 1.7%, higher for the Statistical and Semantic datasets respectively, and 0.5% lower for Statistical-Semantic dataset, when using the Support Vector Machine as opposed to when using the Naïve Bayes classifier.

Table 6.6 The accuracy of the classified reviews at 3-class classification by support vector machine and naïve bayes classifiers for the three datasets that have different number of features

Classifier	**Accuracy**	**Statistical Dataset**	**Semantic Dataset**	**Statistical-Semantic Dataset**
Support Vector Machine	Correctly Classified	67.6%	62%	70.1%
	Incorrectly Classified	32.3%	37.9%	29.8%
Naïve Bayes	Correctly Classified	67.1%	61.3%	69.7%
	Incorrectly Classified	32.8%	38.6%	30.2%

Table 6.7 The accuracy of the classified reviews at 4-class classification by support vector machine and naïve bayes classifiers for the three datasets that have different number of features

Classifier	**Accuracy**	**Statistical Dataset**	**Semantic Dataset**	**Statistical-Semantic Dataset**
Support Vector	Correctly	62.5%	57.1%	62.7%

Machine	Classified			
	Incorrectly Classified	37.4%	42.8%	37.2%
Naïve Bayes	Correctly Classified	60.4%	55.4%	63.2%
	Incorrectly Classified	39.5%	44.5%	36.7%

The obtained results indicated that the coverage of the semantic features in its own is not sufficient to get accurate results, hence, we only compared the percentage improvement in the accuracy of both classifiers Support Vector Machine and Naïve Bayes for 3-class and 4-class classification respectively when using the Statistical-Semantic dataset against the Statistical dataset. The obtained results in Table 6.8 and Table 6.9 evidenced that there was a noticeable improvement of both classifiers on each of the precision, recall and f-measure of the classified reviews. For example, the improvement was from +2% to +3% for 3-class classification and from +0% to +%3 for 4-class classification. Hence, complementing the Statistical dataset with the Semantic dataset enhanced the quality of the training data and resulted in improving the performance of opinion classification task on a multi-point scale.

Table 6.8 The percentage improvement of classifiers for 3-class classification when using the statistical-semantic dataset against statistical dataset

Classifier	P	R	F
Support Vector Machine	+3%	+3%	+3%
Naïve Bayes	+3%	+2%	+3%

Table 6.9 The percentage improvement of classifiers for 4-class classification when using the statistical-semantic dataset against statistical and semantic dataset

Classifier	P	R	F
Support Vector Machine	+3%	+0%	+0%
Naïve Bayes	+3%	+3%	+3%

6.4 Discussion

In this chapter, the Classification process of the proposed Hybrid Semantic Knowledgebase-Machine Learning approach was explained to address our research question RQ7 (*How can we use Semantic Knowledgebase approach to improve the* *quality of training features that are then used to build a Machine Learning classifier in order to improve the accuracy of opinion classification on a multi-point scale?);* which basically relies on building a Machine Learning classifier using a dataset of combined semantic and statistical features that was generated via a new Opinion Classification Extraction algorithm for the intension of improving the performance of opinion classification task on a multi-point scale. The Vector Space Model was used to generate the statistical features that represent the frequency of the refined terms in textual reviews. SPARQL queries were implemented to retrieve from the developed semantic knowledgebase the semantic features that represent facts about the semantically structured opinions about domain features. Machine Learning approaches have been commonly applied for the process of opinion classification and are known to deliver outstanding performance, especially when they are trained using an effective dataset of features that have been manually annotated by a human expert who tend to enhance the annotation process with domain background knowledge. The Semantic Knowledgebase approach uses a knowledgebase that represents a shared understanding of the domain of interest to provide a deep understanding of the structure and knowledge of the content to correctly extract domain features and their relevant sentiments and then determine the polarity of each sentiment (i.e. opinions). The experimental results for the opinion classification task demonstrated that the proposed Opinion Classification algorithm enhanced the classification accuracy on a multi-point scale, which answers the hypothesis of whether adding additional semantic features to dataset of statistical features can result in higher classification accuracy, as opposed to using a statistical dataset containing the frequencies of features.

Chapter 7

7 Dynamics of the Population and Interrogation of the Problem Domain Knowledgebase

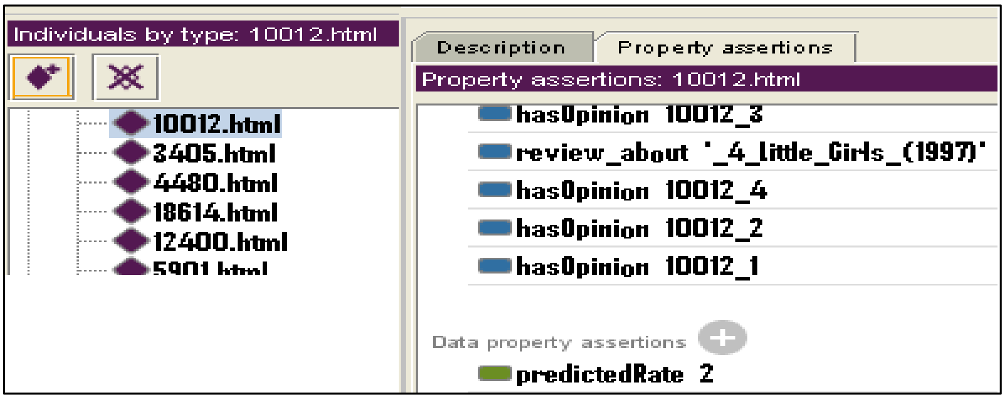
This chapter illustrates the process of updating the developed movie-review knowledgebase (i.e. enriched with semantic information in addition to its comprehensive knowledge) with the classification results (i.e. the calculated rating class) for the pre-processed movie reviews, which were obtained after the completion of the Classification process of the proposed Hybrid Semantic Knowledgebase-Machine Learning approach (as described in chapter 6). Thereafter, this chapter demonstrates the usability of the developed movie-review knowledgebase for sophisticated interrogation of opinions and for recommending a specific movie.

7.1 Inserting the Classification Results into the Domain Knowledgebase

The developed movie-review knowledgebase was accumulatively enriched with the semantically annotated movie's features and sentiments extracted from the review (i.e. semantic information) as explained in section 5.2.4; the semantic information were then used to produce an enriched dataset of semantic features for the purpose of enhancing the opinion classification task on a multi-point scale as demonstrated in section 6.2.1. The further step is to insert the obtained classification results (i.e. the calculated rating class) into the developed movie-review knowledgebase as follows:

- Getting the prediction rate from the obtained classification results for each review.
- Performing a SPARQL Construct Query that inserts the obtained prediction rate for each movie review using the relation "review predictedRate predicted-rate", where "predictedRate" is a datatype relation and "predicted-rate" is a datatype value. An example of inserting the predicted rate (e.g. 2) for the review (e.g. Review1) is "Review1 predictedRate 2". Figure 7.1 presents a snapshot of the inserted prediction rate into the movie-review knowledgebase for a review about 4

Little Girls (1997) movie.



Individuals by type: 10012.html

- 10012.html
- 3405.html
- 4480.html
- 18614.html
- 12400.html
- 5901.html

Property assertions: 10012.html

- hasOpinion 10012_3
- review_about '_4_Little_Girls_(1997)'
- hasOpinion 10012_4
- hasOpinion 10012_2
- hasOpinion 10012_1

Data property assertions

- predictedRate 2

Figure 7.1 A snapshot of inserted prediction rate into movie-review knowledgebase for a review about 4 little girls (1997) movie.

7.2 Interrogation of Opinions from the movie-review knowledgebase

The semantically structured movie-review knowledgebase can be further used to infer valuable semantic information about the main domain concepts (such as movie) as well as the expressed opinions on its constituent features. For example, it is possible to compute the overall opinions about a movie across multiple reviews as well as for the cinematic features (actors, script, sound effects, etc.). In addition, the movie-review knowledgebase should be able to answer fairly complex queries such as a query about movies with the favorable screenplay (i.e. opinionated domain features), filtered by non-opinionated domain features such as genre, actor, origin, etc.

We demonstrate the usability of the developed movie-review knowledgebase for sophisticated interrogation of opinions and for recommending a specific movie using (i.e. prediction) through the following examples of queries with answers.

Query1: Overall opinions about movies across multiple reviews

In this SPARQL query, the recommender function retrieves the overall average of the predicted rate about each movie across all movie reviews.

```
Prefix owl:<http://www.movie-review-ontology.owl#>
PREFIX rdf:<http://www.w3.org/1999/02/22-rdf-syntax-ns#>

Select   (?movie as ?c1) (AVG(?predictedrate) as ?c2)
```

```
WHERE
{
?movie rdf:type owl:Movie .
?review owl:review_about ?movie .
?review owl:predictedRate ?predictedrate .
}
 GROUP BY ?movie
```

Movie	Average Predicted Rate
Erleuchtung_garantiert_(2000)	2.0
Haunting_The_(1999)	0.0
Stalingrad_(1993)	2.0
Gone_in_Sixty_Seconds_(2000)	0.0
Soldier_(1998)	0.0
Just_Cause_(1995)	1.0
Wisconsin_Death_Trip_(1999)	2.0
Ref_The_(1994)	0.0
Instinct_(1999)	1.0
Mission:_Impossible_(1996)	1.0
Junior_(1994)	0.0
Clear_and_Present_Danger_(1994)	2.0
Remember_the_Titans_(2000)	2.0
Crocodile_Dundee_(1986)	2.0
Mercury_Rising_(1998)	2.0
Wo_hu_cang_long_(2000)	1.0

Query2: All movies that have a (very positive/positive/neutral/negative/very negative) domain feature such as (screenplay, actor, script, etc.)

This request queries the movie-review knowledgebase to get all movies that have very positive opinions on a specific domain feature (in this case “screenplay”).

```
Prefix owl:<http://www.movie-review-ontology.owl#>
Prefix rdf:<http://www.w3.org/1999/02/22-rdf-syntax-ns#>

SELECT distinct ?movie

WHERE
{
     ?movie rdf:type owl:Movie .
     ?review rdf:type owl:Review .
     ?review owl:review_about ?movie .
     ?opinion rdf:type owl:Opinion .
     ?review owl:hasOpinion ?opinion .
     ?opinion owl:hasPolarity  owl:Very_Positive .
     ?opinion owl:describesFeature  ?screenplay .
     ?screenplay  rdf:type owl:Screenplay .
}
```

Movie
Flipper_(1996)
Heidi_Fleiss:_Hollywood_Madam_(1995)_(TV)
Godfather:_Part_II_The_(1974)
Soldier_(1998)
Atlantis:_The_Lost_Empire_(2001)
Schindler's_List_(1993)
To_Gillian_on_Her_37th_Birthday_(1996)
Earth_(1998)
Last_Supper_The_(1995)……
……

Query3: Name of (star, writer, editor, etc.) that has (very positive/positive/neutral/negative/ very negative) polarity across all reviews

In this query, names of people who are related to a movie (in this case "star") and have a very positive polarity are retrieved.

```
Prefix owl:<http://www.movie-review-ontology.owl#>
PREFIX rdf:<http://www.w3.org/1999/02/22-rdf-syntax-ns#>
SELECT  distinct ?star
WHERE
{
?opinion rdf:type owl:Opinion .
?opinion owl:describesMovieRelatedPeople  ?star .
?star  rdf:type owl:Starring .
?opinion owl:hasPolarity  owl:Very_Positive .
}
```

Star
Kevin_Spacey
Christopher_Eccleston
Martin_Lawrence
James_Caan
Fionnula_Flanagan
Nandita_Das
Cameron_Diaz
Vincent_Price
Linda_Fiorentino
Lee_Remick
……
……

Query4: Opinion Phrases that expressed on a domain feature (screenplay, actor, script, etc.) across all reviews

In this query, we retrieve all opinion phrases (e.g. the beauty of script) that were expressed on a specific domain feature (in this case "set design").

```
Prefix owl:<http://www.movie-review-ontology.owl#>
PREFIX rdfs:<http://www.w3.org/2000/01/rdf-schema#>
PREFIX rdf:<http://www.w3.org/1999/02/22-rdf-syntax-ns#>
SELECT   distinct (?phrase)
WHERE
{
?opinion rdf:type owl:Opinion .
?opinion owl:describesFeature  ?set_design .
?set_design  rdf:type owl:Set_Design .
?opinion owl:hasPhrase ?phrase .
}
```

Phrase
warm_Set_Design
evocative_Set_Design
creepy_Set_Design
stylish_Set_Design
lush_Set_Design
terrific_Set_Design
richly_Set_Design
even_worse_Set_Design
intentionally_cheap_Set_Design
imaginative_Set_Design
sumptuous_Set_Design

Query5: All movies that have two or three (very positive/positive/neutral/negative/ very negative) domain features

This request queries the movie-review knowledgebase to get all movies that have positive opinions on two or more specific domain features (in this case “performance and starring”).

```
Prefix owl:<http://www.movie-review-ontology.owl#>
PREFIX rdf:<http://www.w3.org/1999/02/22-rdf-syntax-ns#>
SELECT   (?movie)
WHERE
{

?opinion owl:describesFeature  ?feature .
{?feature  rdf:type owl:Performance} UNION {?feature  rdf:type
owl:Starring} .
?opinion owl:hasPolarity  owl:Positive .
?movie rdf:type owl:Movie .
?review rdf:type owl:Review .
?review owl:review_about ?movie .
?review owl:hasOpinion ?opinion .
}
GROUP BY ?movie
```

Movie
Blow_(2001)
Flirting_with_Disaster_(1996)
Life_(1999)
Godfather:_Part_II_The_(1974)
Dunston_Checks_In_(1996)
Browning_Version_The_(1994)
Titanic_(1997)
Map_of_the_World_A_(1999)
Just_Cause_(1995)
Dinosaur_(2000)
Marvin's_Room_(1996)

Query6: All movies that their language are (English, American, etc.) and have a (very positive/positive/neutral/negative/very negative) domain feature such as (screenplay, actor, script, etc.)

This query presents a combination of using opinionated domain feature with non-opinionated domain features such as getting all English movies that have very positive opinions on a specific domain feature (in this case “direction”).

```
Prefix owl:<http://www.movie-review-ontology.owl#>
PREFIX rdf:<http://www.w3.org/1999/02/22-rdf-syntax-ns#>
SELECT   (?movie)
WHERE
{
?opinion rdf:type owl:Opinion .
?opinion owl:describesFeature  ?feature .
?feature  rdf:type owl:Direction .
?opinion owl:hasPolarity  owl:Very_Positive .
?movie rdf:type owl:Movie .
?review rdf:type owl:Review .
?review owl:review_about ?movie .
?review owl:hasOpinion ?opinion .
?movie owl:hasLanguage owl:English .
}
GROUP BY ?movie
```

Movie
Air_Force_One_(1997)
General_The_(1998)
Crucible_The_(1996)
Independence_Day_(1996)
To_Kill_a_Mockingbird_(1962)

7.3 Chapter Summary

In this chapter, we illustrated the process of updating the developed knowledgebase with the classification results, and then we demonstrated the advantage of the semantic modeling of the movie-review knowledgebase (i.e. enriched with semantic information in addition to its comprehensive knowledge) for inferring valuable semantic information about the domain as well as the expressed opinions on the domain features. In addition, we demonstrated the usability of the developed movie-review knowledgebase for recommending a specific movie using the inserted classification results.

Chapter 8

8 Conclusion and Future Work

8.1 Overview

Online Opinions that have been published on blogs, forums, and social networks play an important role in supporting consumers make decisions about purchasing products or services. In addition, customers' opinions allow companies to understand the strengths and limitations of their products and services and improve upon these. The challenge is that online opinions are predominantly expressed in natural language text, and hence opinion mining tools are required to facilitate the effective extraction and analysis of opinions from unstructured text. In this research, we introduced a new hybrid approach that will semantically extract and analyse opinions from unstructured online reviews by integrating Semantic Knowledgebase and Machine Learning approaches to improve the actionable intelligence extraction and analysis of opinions from unstructured domain reviews.

This approach comprises several stages, in which each stage was developed to improve opinion mining challenges at domain feature level. In the initial stage, we constructed a semantic knowledgebase that contains comprehensive knowledge of the problem domain. Constructing a semantic knowledgebase starts with modelling the domain knowledge into a domain model that can represent and associate generic information about the domain, opinions as well as its reviews. The domain model was then translated into a formal ontology that represents the schemata for populating the domain knowledgebase with structured information. The semantic structure of the domain knowledgebase provides for obtaining data from other public sources that use similar standards for data structuring such as Linked Open Datasets, which can be used, for instance, to populate the domain knowledgebase with dynamic ground facts about the problem domain, which is considered valuable for the process of opinion mining at domain feature level.

In the second stage, we developed and implemented the domain feature extraction process to extract domain features from movie reviews. Linked Open Data resources such as DBpedia and Internet Movie Database were utilised to populate the constructed semantic domain knowledgebase with structured relevant ground facts about each processed domain review via performing composed SPARQL Construct queries. A set of Natural Language Processing components were built to obtain the linguistic and syntactic structure of the textual review such as tokenising, tagging, lemmatising the review content as well as determine the dependency relation between them. To extract the domain features, the populated semantic domain knowledgebase was utilised to identify domain's key concepts and their synonyms and ground facts from the processed reviews. The identification was based on linking between the root of each word in the pre-processed reviews and the conceptualised terms in the semantically structured domain knowledgebase via implementing GATE's Onto Root Gazetteer and hand crafted JAPE rules. The domain feature extraction process has performed better with the produced semantic domain knowledgebase that has more comprehensive coverage than similar reported works. However, the characteristic of the problem domain (e.g. movie reviews) affected the performance of the domain feature extraction as we observed that reviewers tend to mention the full name of people (e.g. Spike Lee) at the first time of expressing opinions on them, and then only single names (e.g. Lee) or pronouns (e.g. s/he) are mentioned to express opinions. Moreover, we observed that movie reviews contain opinions on movie's features such as (movie names and names of stars, writers, editors, etc.) that belong to the target movie as well as to other movies that are sometimes discussed in the review.

Therefore, Co-referencing resolution process was deployed to identify the orthographic and pronominal relations between the identified domain features and single names and pronouns to further identify non-explicit domain features via implementing hand-crafted JAPE rules with GATE's ANNIE Transducer and GATE's Co-referencing components. The conducted evaluation showed that the performance of domain feature extraction task was further improved after deploying co-reference resolution for non-explicit domain features. Furthermore, the relevant semantically structured ground facts about the target domain review were exploited to discard irrelevant domain features via performing SPARQL's ASK query. The conducted evaluation showed that the accuracy of the domain feature extraction process was further improved by consulting the semantic knowledgebase to filter out irrelevant domain features.

In the third stage, we developed and implemented the domain feature-sentiment association process to associate the extracted domain features with their corresponding features. Sentiment lexicon was used to extract sentiment words from the pre-processed reviews. Following the identification of sentiments, any adjacent shifters (negation or adverb) were taken into account to moderate the sentiment's score accordingly. To associate the extracted domain features with the extracted sentiments, a set of dependency pattern rules was implemented based on the syntactical structure of the content to identify patterns that contain both domain feature and sentiment, which were then associated together. The performance of the domain feature-sentiment association process was not satisfactory due to the fact that using dependency pattern rules results in associating all domain features with their corresponding sentiment whether they present descriptive or subjective opinion phrases.

Therefore, a sentiment lexicon for each group of domain features was generated, which contains a list of sentiments that can be used only to express subjective opinions for a specific group of domain features. The generated domain sentiment lexicons were used to discard the identified patterns that contain descriptive opinions. Further evaluation demonstrated that analysing the subjectivity of opinion phrases improved the performance of domain feature-sentiment association process.

In the fourth stage, the semantically structured domain knowledgebase that was used to bootstrap the domain feature extraction process was further enriched with new semantic information related to the analysed review and the corresponding semantically annotated movie's features and their corresponding sentiments as well as their polarities. The resulting domain knowledgebase represents a valuable resource not only for predicting general opinion about a domain, but also for sophisticated retrieval of opinions associated with a specific domain feature. In this research, after the completion of enriching the domain knowledgebase, we deemed worthwhile to investigate whether the calculated features' sentiment polarities are sufficient to perform opinion classification task on a multi-point scale without further analysis. The classification accuracy in the obtained results was not satisfactory, hence we decided to investigate the deployment of Machine Learning approaches for performing opinion classification on a multi-point scale.

In the fifth stage, a novel hybrid Semantic Knowledgebase-Machine Learning approach was developed for classifying the overall opinion of the reviews on a multi-point scale. It is based on combining statistical features with semantic features for bootstrapping the Machine Learning opinion classifiers. The Vector Space Model was used to generate the statistical features that represent the frequency of the refined terms in textual reviews. SPARQL queries were implemented to retrieve from the developed semantic knowledgebase the semantic features that represent facts about the semantically structured opinions about domain features. The experimental results for the opinion classification task demonstrated that the proposed approach enhanced the classification on a multi-point scale, which answers the hypothesis of whether complementing the dataset of statistical features with semantic knowledge-based semantic features can result in an improved classification accuracy.

The final stage in this research focused on updating the developed movie-review knowledgebase with the obtained classification results (i.e. the calculated rating class) for the pre-processed reviews. Thereafter, complex SPARQL queries were used to evaluate the usability of the developed domain knowledgebase for sophisticated interrogation of opinions and for the recommender functions. The knowledgebase response demonstrated that the movie-review knowledgebase was able to answer fairly complex queries such as a query about movies with the favourable screenplay (i.e. opinionated domain features), filtered by non-opinionated domain features such as genre, actor, origin, etc.

8.2 Thesis Contributions

The main aim of this research, “Exploiting Domain Knowledge to Enhance Opinion Mining using A Hybrid Semantic Knowledgebase-Machine Learning approach”, has been fulfilled by successfully addressing the research and development challenges of a novel Hybrid Semantic Knowledgebase-Machine Learning approach as detailed in the previous chapters. Below we revisit how this work responded to the research and development challenges documented at the start of the PhD research investigation.

RQ1. How can the semantic modelling of the domain knowledge further contribute to improving the opinion mining at domain feature level, in particular to the domain feature extraction and opinion classification tasks?

In our research, the required domain knowledge represents the domain's environment that contains the problem domain's key concepts and synonyms and ground facts, as well as the relation between them. The semantic modelling of domain knowledge provided for the comprehensive representation of the problem domain, which facilitated identifying domain features from movie review as well as eased the connection with other related domains such as reviews and opinions for opinion mining process. In addition, the semantic structure of the knowledgebase based on the semantic modelling provided us for obtaining dynamic ground facts about the problem domain from other public sources that use similar standards for data structuring such as Linked Open Datasets. Moreover, the semantic modelling of the domain knowledgebase facilitated the inference of valuable semantic information about the main domain concepts (such as movie) as well as the expressed opinions on its constituent features that in turn enhanced the accuracy of the opinion classification task. Furthermore, the semantic modelling improved the usability of the developed knowledgebase for sophisticated interrogation of opinions and for recommending a specific domain.

RQ2. Can the domain knowledge improve the precision and recall of the feature extraction task?

In this study, we used a Semantic Knowledgebase approach to extract domain features from movie reviews. A new Domain Feature Extraction algorithm was introduced for extracting domain features from movie reviews. The main objective of our work is to utilise a comprehensive domain knowledgebase and populate it with domain's ground facts that are obtained from Linked Open Data resources in order to provide deep understanding of the free-textual contents, which is envisaged to improve the performance of domain feature extraction task. The comprehensive domain knowledgebase was utilised to link between its conceptualised knowledge (domain's key concepts and their synonyms and ground facts) and the lemmatised words in the review. Synonym words are matched to their key concepts in the domain knowledgebase. For example, the word (movie) and synonym words (film, show and picture) are matched to the same key concept (MOVIE) in the movie-review knowledgebase. Words that represent ground facts such as movie names, names of stars, writers, and editors are matched to the same individuals in the movie-review knowledgebase. Hence, exploiting the domain knowledgebase for domain feature extraction helped to overcome the limitation of extracting domain features from textual reviews using the other approaches such as Association Rule Mining, where the extracted domain features tend to be frequent domain features, whereas infrequent domain features are ignored. In addition, some of the extracted nouns and noun phrases may not be domain features even if these occur more frequently in the textual contents. The developed domain knowledgebase-based Semantic Knowledgebase approach also improves on Machine Learning approach, where training datasets need to be manually annotated by human experts in order to deliver significant results, which can be an extremely time-consuming task as the required size of the training dataset should be sufficiently large to bootstrap the learning algorithms.

RQ3. How can the semantically structured public datasets be exploited to improve the performance of domain feature extraction task?

Movie review contains opinions on the target movie and its features (movie name and names of stars, writers, editors, etc.), but sometimes can contain opinions about other movies. Hence, the extracted domain features by Semantic Knowledgebase approach from movie reviews might not necessarily be relevant to the target movie. For example, the sentence "Matt Damon, who seemed relatively lost in THE RAINMAKER, this time he delivers a brilliant and complex performance" and the sentence "The HOME ALONE's star, Macaulay Culkin, is missing from the latest episode, HOME ALONE 3" are both extracted from a review about a movie "HOME ALONE 3". As obvious from both sentences, the movie "THE RAINMAKER" and the star "Macaulay Culkin" certainly are not relevant to the movie "HOME ALONE 3". The related state-of-the-art approaches have not considered eliminating such non-relevant domain features, which can reduce the precision of the extracted domain features. In this research, we addressed this challenge by investigating, where possible, each matched domain feature against the relevant semantically structured ground facts by performing SPARQL’s ASK Queries over the developed movie-review knowledgebase, which was populated utilising Linked Open Data resources.

RQ4. Given the fact that the target domain feature is presented by a single name or pronoun (i.e. termed non-explicit domain features), how can the semantically constructed knowledgebase be utilised with co-reference resolution to extract non-explicit domain features to further improve the domain feature extraction task?

Most reviewers tend to mention the full name of people at the first time of expressing opinions on them, and then only single names or pronoun are mentioned to express opinions. In this research, using the semantic domain knowledge enabled us to match both full names and single names. However, the matched single names referred to the target full names within the reviews as well as to the other full names within the semantic knowledgebase. This is due to the fact that for example two full names can have the same single name (e.g. Ahmed Ali, Ali Salem). Therefore, we looked further and used co-reference resolution to connect single names and pronouns with their referred full names. As the full names already matched by the semantic knowledgebase, their specification (i.e. the full name, their object relation, etc.) were inherited to the referred single names and pronouns. Identifying such non-explicit features was essential to enhance domain feature extraction task. However, the co-reference resolution was unable to deal with the terms “this and it” when they are used to refer to a movie name, which has affected slightly the success of identifying non-explicit domain features.

RQ5. Can the domain’s sentiment lexicon contribute to improve the domain feature-sentiment association task?

Domain features that are extracted from a textual content might not be have any subjective opinions about them as users maybe describe factual information about the extracted domain features as in “the American movie is my favourite”. Discarding subjective opinions is still challenging for many researchers. In this research, utilising the domain knowledge to create domain’s sentiment lexicon enabled us to eliminate descriptive opinions, and hence to improve up on the state of the art related works that used syntactic parsing techniques (i.e. identify both descriptive and subjective phrases). However, our analysis of the results revealed that there are some limitations in the output of the association mechanism that affected the performance of domain feature-sentiment association task.

RQ6. Is the aggregation of the domain features' sentiment polarities based on Semantic Knowledgebase approach sufficient for the accurate classification of the review opinion?

No, the conducted results indicate that classifying reviews using classification rules worked quite well for 2-class classification only with an average 77.4%, whereas, the results were not satisfied for 3-class and 4-class classification with an average 46.3% and 43% respectively. Therefore, we further involved Machine Learning approach for performing opinion classification on a multi-point.

RQ7. How can we use Semantic Knowledgebase approach to improve the quality of training features that are then used to build a Machine Learning classifier in order to improve the accuracy of opinion classification on a multi-point scale?

Machine Learning approaches have been commonly applied for the process of opinion classification and are known to deliver outstanding performance, especially when they are trained using an effective dataset of features that have been manually annotated by a human expert who tend to enhance the annotation process with domain background knowledge. The Semantic Knowledgebase approach uses a knowledgebase that represents a shared understanding of the domain of interest to provide a deep understanding of the structure and knowledge of the content to correctly extract domain features and their relevant sentiments and then determine the polarity of each sentiment (i.e. opinions). In this research, we introduced a Hybrid Semantic Knowledgebase-Machine Learning approach that based on integrating the advantages of Semantic Knowledgebase approaches with the advantages of Machine Learning approaches. To integrate, we semantically constructed a domain knowledgebase and populated it with relevant ground facts from structured public dataset. After the semantic domain knowledgebase facilitated the extraction of domain features from reviews, we enriched it with semantically structured information about the extracted domain features and the analysed reviews. Thereafter, we produced semantic features about the semantically structured opinions from the semantic domain knowledgebase, which are then added to statistical features for training Machine Learning classifier to classify the opinions on multi-point rating scale. The experimental results for the opinion classification task demonstrated that the proposed approach enhanced the classification on a multi-point scale.

8.3 Future Work

Some future research works are debated as follows:

- **Investigating the feasibility of applying the Hybrid Semantic Knowledgebase-Machine Learning approach to short text reviews**

Analysis of social media posts especially Twitter has become the most popular sources for conducting researches on sentiment analysis because it is very convenient to collect the activity of users. However, Twitter allows users to view and share limited character messages with the public, which would pose a challenge because the volume and quality of the semantic information within these posts are significantly less than within textual reviews (i.e. represent elaborate reviews written by expert critics). As in this research the analysed domain reviews represent elaborate reviews written by expert critics, hence, a possible area for further research would be investigating whether applying the proposed Hybrid Semantic Knowledgebase-Machine Learning approach is going to be useful for short texts domains (e.g. Twitter Posts) or not.

- **Investigation Applying fuzzy logic algorithms on the semantically structured opinions for multi-class classification**

Fuzzy logic algorithms are used for making decisions based on multiple criteria with complex interlinks between them. Applying fuzzy logic algorithms for opinion mining analysis has been a fertile research area (Howells and Ertugan 2017). The process of Fuzzy logic starts with converting a dataset of a crisp input into fuzzy sets using fuzzy linguistic variables, fuzzy linguistic terms and membership functions. After that, inferencing process is applied on the generated fuzzy sets based on a set of rules such as using "if-then" rules. Finally, the obtained fuzzy is mapped to a crisp output using the membership functions. Hence, a possible area for further research would be investigating applying fuzzy engine that periodically compares the predicted review's rate and the target review’s rate, and produces a confidence score of each classified review.

- **Investigation on developing a SPARQL based Natural Language Query engine**

There is a rich body of work investigating the development of Natural Language query engine based on SPARQL for improving the process of converting the naturel language query into advanced standards queries such as RDF querying language to enhance the user’s interactivity with the system. The main objective is based on processing the user’s natural language text and extracting from it the semantic information, which is then used to retrieve the accurate information from the developed domain knowledgebase (Bouayad-Agha, Casamayor and Wanner 2014, Suryanarayana, et al. 2018). A possible area of further research would be developing a Natural query portal to querying knowledge-based query engine. The idea is based on utilising the enriched domain knowledgebase to generate training dataset present all the inserted semantically structured opinions and generate the target labels for these datasets to be a encoded as structured queries. The Machine Learning classifier will be trained using these training dataset. After that, using our approach to process the user’s natural language query and extract from them the semantic information, which is then used to generate structured opinions. The generated structured opinions will be passed to the trained classifier to obtain the predicted structured RDF queries.

9 References

Acampora, G. and Cosma, G., 2015. A comparison of fuzzy approaches to e-commerce review rating prediction.

Agarwal, A. and Toshniwal, D., 2018. Application of Lexicon Based Approach in Sentiment Analysis for short Tweets. *In: 2018 International Conference on Advances in Computing and Communication Engineering (ICACCE),* IEEE, pp. 189-193.

Agarwal, B., et al., 2015a. Sentiment analysis using common-sense and context information. *Computational Intelligence and Neuroscience,* 2015, 30.

Agarwal, B., et al., 2015b. Concept-level sentiment analysis with dependency-based semantic parsing: a novel approach. *Cognitive Computation,* 7 (4), 487-499.

Alfrjani, R., Osman, T. and and Cosma, G., 2018. A Hybrid Semantic Knowledgebase-Machine Learning Approach for Opinion Mining. *Submitted to "Elsevier Data and Knowledge Engineering Journal" on 26/1/2018,* .

Alfrjani, R., Osman, T. and Cosma, G., 2017. Exploiting domain knowledge and public linked data to extract opinions from reviews. *In: Knowledge Engineering and Applications (ICKEA), 2017 2nd International Conference on,* IEEE, pp. 98-102.

Alfrjani, R., Osman, T. and Cosma, G., 2016. A new approach to ontology-based semantic modelling for opinion mining. *In: Computer Modelling and Simulation (UKSim), 2016 UKSim-AMSS 18th International Conference on,* IEEE, pp. 267-272.

Ali, F., Kim, E.K. and Kim, Y., 2015. Type-2 fuzzy ontology-based opinion mining and information extraction: A proposal to automate the hotel reservation system. *Applied Intelligence,* 42 (3), 481-500.

Allison, B., 2008. Sentiment detection using lexically-based classifiers. *In: Text, speech and dialogue,* Springer, pp. 21-28.

Asghar, N., 2016. Yelp Dataset Challenge: Review Rating Prediction. *arXiv Preprint arXiv:1605.05362,* .

Baca-Gomez, Y.R., et al., 2016. Web Service SWePT: A Hybrid Opinion Mining Approach. *J.Ucs,* 22 (5), 671-690.

Baccianella, S., Esuli, A. and Sebastiani, F., 2010. SentiWordNet 3.0: An Enhanced Lexical Resource for Sentiment Analysis and Opinion Mining. *In: LREC,* pp. 2200-2204.

Balage Filho, P. and Pardo, T., 2013. NILC_USP: A Hybrid System for Sentiment Analysis in Twitter Messages. *In: SemEval@ NAACL-HLT,* pp. 568-572.

Baroni, M. and Vegnaduzzo, S., 2004. Identifying subjective adjectives through web-based mutual information. *In: Proceedings of KONVENS,* pp. 17-24.

Berners-Lee, T., Hendler, J. and Lassila, O., 2001. The semantic web. *Scientific American,* 284 (5), 34-43.

Bespalov, D., et al., 2011. Sentiment classification based on supervised latent n-gram analysis. *In: Proceedings of the 20th ACM international conference on Information and knowledge management,* ACM, pp. 375-382.

Bhatnagar, V., Goyal, M. and Hussain, M.A., 2018. A Novel Aspect Based Framework for Tourism Sector with Improvised Aspect and Opinion Mining Algorithm. *International Journal of Rough Sets and Data Analysis (IJRSDA),* 5 (2), 119-130.

Bizer, C., et al., 2009. DBpedia-A crystallization point for the Web of Data. *Web Semantics: Science, Services and Agents on the World Wide Web,* 7 (3), 154-165.

Bouayad-Agha, N., Casamayor, G. and Wanner, L., 2014. Natural language generation in the context of the semantic web. *Semantic Web,* 5 (6), 493-513.

Cambria, E., et al., 2010. SenticNet: A Publicly Available Semantic Resource for Opinion Mining. *In: AAAI fall symposium: commonsense knowledge,* .

Chakraborty, G. and Pagolu, M.K., 2014. Analysis of unstructured data: Applications of text analytics and sentiment mining. *In: SAS global forum,* pp. 1288-2014.

Chakraborty, K., et al., 2018. Comparative Sentiment Analysis on a Set of Movie Reviews Using Deep Learning Approach. *In: International Conference on Advanced Machine Learning Technologies and Applications,* Springer, pp. 311-318.

Cosma, G. and Acampora, G., 2016. A computational intelligence approach to efficiently predicting review ratings in e-commerce. *Applied Soft Computing,* 44, 153-162.

Cruz, F., et al., 2008. Experiments in sentiment classification of movie reviews in Spanish. *Procesamiento Del Lenguaje Natural,* 41, 79-80.

Dalvi, B., et al., 2015. Automatic gloss finding for a knowledge base using ontological constraints. *In: Proceedings of the Eighth ACM International Conference on Web Search and Data Mining,* ACM, pp. 369-378.

Davidov, D., Tsur, O. and Rappoport, A., 2010. Enhanced sentiment learning using twitter hashtags and smileys. *In: Proceedings of the 23rd international conference on* *computational linguistics: posters,* Association for Computational Linguistics, pp. 241-249.

Deng, Z., Luo, K. and Yu, H., 2014. A study of supervised term weighting scheme for sentiment analysis. *Expert Systems with Applications,* 41 (7), 3506-3513.

Ding, X. and Liu, B., 2007. The utility of linguistic rules in opinion mining. *In: Proceedings of the 30th annual international ACM SIGIR conference on Research and development in information retrieval,* ACM, pp. 811-812.

Domingos, P. and Pazzani, M., 1997. On the optimality of the simple Bayesian classifier under zero-one loss. *Machine Learning,* 29 (2), 103-130.

Eirinaki, M., Pisal, S. and Singh, J., 2012. Feature-based opinion mining and ranking. *Journal of Computer and System Sciences,* 78 (4), 1175-1184.

El-Halees, A. and Al-Asmar, A., 2017. Ontology Based Arabic Opinion Mining. *Journal of Information & Knowledge Management*, 16 (03), 1750028.

Gadekallu, T., et al., 2019. Application of Sentiment Analysis in Movie reviews. *In:* Application of Sentiment Analysis in Movie reviews. *Sentiment Analysis and Knowledge Discovery in Contemporary Business.* IGI Global, 2019, pp. 77-90.

Gartenberg, J., 1989. Glossary of Filmographic Terms, Version II. *Brussels: FIAF,* .

Gezici, G., et al., 2013. SU-Sentilab: A Classification System for Sentiment Analysis in Twitter. *In: SemEval@ NAACL-HLT,* pp. 471-477.

Ghorashi, S.H., et al., 2012. A frequent pattern mining algorithm for feature extraction of customer reviews. *In: IJCSI International Journal of Computer Science Issues,* Citeseer, .

González-Ibánez, R., Muresan, S. and Wacholder, N., 2011. Identifying sarcasm in Twitter: a closer look. *In: Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies: Short Papers-Volume 2,* Association for Computational Linguistics, pp. 581-586.

Gravano, L., García-Molina, H. and Tomasic, A., 1999. GlOSS: text-source discovery over the Internet. *ACM Transactions on Database Systems (TODS),* 24 (2), 229-264.

Greene, S.C., 2007. *Spin: Lexical semantics, transitivity, and the identification of implicit sentiment.* University of Maryland, College Park.

Guarino, N., 1998. Formal ontology and information systems. *In: Proceedings of FOIS,* pp. 81-97.

Hatzivassiloglou, V. and McKeown, K.R., 1997. Predicting the semantic orientation of adjectives. *In: Proceedings of the eighth conference on European chapter of the Association for Computational Linguistics,* Association for Computational Linguistics, pp. 174-181.

Howells, K. and Ertugan, A., 2017. Applying fuzzy logic for sentiment analysis of social media network data in marketing. *Procedia Computer Science,* 120, 664-670.

Hu, M. and Liu, B., 2004. Mining and summarizing customer reviews. *In: Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining,* ACM, pp. 168-177.

Hu, Y., Chen, Y. and Chou, H., 2017. Opinion mining from online hotel reviews–A text summarization approach. *Information Processing & Management,* 53 (2), 436-449.

Huang, H., Wang, J. and Chen, H., 2017. Implicit opinion analysis: Extraction and polarity labelling. *Journal of the Association for Information Science and Technology,* 68 (9), 2076-2087.

Ibrahim, N.F., Wang, X. and Bourne, H., 2017. Exploring the effect of user engagement in online brand communities: Evidence from Twitter. *Computers in Human Behavior,* 72, 321-338.

Jia, X., et al., 2018. Words alignment based on association rules for cross-domain sentiment classification. *Frontiers of Information Technology & Electronic Engineering,* 19 (2), 260-272.

Joshi, M. and Penstein-Rosé, C., 2009. Generalizing dependency features for opinion mining. *In: Proceedings of the ACL-IJCNLP 2009 conference short papers,* Association for Computational Linguistics, pp. 313-316.

Kamps, J., et al., 2004. Using WordNet to Measure Semantic Orientations of Adjectives. *In: LREC,* Citeseer, pp. 1115-1118.

Karami, A., Bennett, L.S. and He, X., 2018. Mining Public Opinion about Economic Issues: Twitter and the US Presidential Election. *arXiv Preprint arXiv:1802.01786,* .

Kessler, J.S. and Nicolov, N., 2009. Targeting Sentiment Expressions through Supervised Ranking of Linguistic Configurations. *In: ICWSM,* .

Khan, A.Z., Atique, M. and Thakare, V., 2015. Combining lexicon-based and learning-based methods for Twitter sentiment analysis. *International Journal of Electronics, Communication and Soft Computing Science & Engineering (IJECSCSE),* , 89.

Kim, S. and Hovy, E., 2004. Determining the sentiment of opinions. *In: Proceedings of the 20th international conference on Computational Linguistics,* Association for Computational Linguistics, pp. 1367.

König, A.C. and Brill, E., 2006. Reducing the human overhead in text categorization. *In: Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining,* ACM, pp. 598-603.

Krishnan, H., Elayidom, M.S. and Santhanakrishnan, T., 2017. Sentiment Analysis of tweets for inferring popularity of mobile phones. *International Journal of Computer Applications*, 157 (2).

Li, G. and Liu, F., 2012. Application of a clustering method on sentiment analysis. *Journal of Information Science,* 38 (2), 127-139.

Li, Y., et al., 2015. A holistic model of mining product aspects and associated sentiments from online reviews. *Multimedia Tools and Applications*, 74 (23), 10177-10194.

Liu, B., 2012. Sentiment analysis and opinion mining. *Synthesis Lectures on Human Language Technologies,* 5 (1), 1-167.

Lunardi, A., et al., 2016. Domain-Tailored Multiclass Classification of User Reviews Based on Binary Splits. *In: International Conference on Social Computing and Social Media,* Springer, pp. 298-309.

Ma, B., et al., 2013. An LDA and synonym lexicon based approach to product feature extraction from online consumer product reviews. *Journal of Electronic Commerce Research,* 14 (4), 304.

Manek, A.S., et al., 2017. Aspect term extraction for sentiment analysis in large movie reviews using Gini Index feature selection method and SVM classifier. *World Wide Web,* 20 (2), 135-154.

Marchand, M., et al., 2013. [LVIC-LIMSI]: Using Syntactic Features and Multi-polarity Words for Sentiment Analysis in Twitter. *In: Second Joint Conference on Lexical and Computational Semantics (* SEM), Volume 2: Proceedings of the Seventh International Workshop on Semantic Evaluation (SemEval 2013),* pp. 418-424.

Martínez-Cámara, E., Martín-Valdivia, M.T. and Ureña-López, L.A., 2011. Opinion classification techniques applied to a spanish corpus. *In: International Conference on Application of Natural Language to Information Systems,* Springer, pp. 169-176.

MartíN-Valdivia, M., et al., 2013. Sentiment polarity detection in Spanish reviews combining supervised and unsupervised approaches. *Expert Systems with Applications,* 40 (10), 3934-3942.

Meena, A. and Prabhakar, T., 2007. Sentence level sentiment analysis in the presence of conjuncts using linguistic analysis. *In: ECiR,* Springer, pp. 573-580.

Miao, Q., Li, Q. and Zeng, D., 2010. Mining fine grained opinions by using probabilistic models and domain knowledge. *In: Web Intelligence and Intelligent Agent Technology (WI-IAT), 2010 IEEE/WIC/ACM International Conference on,* IEEE, pp. 358-365.

Moraes, R., Valiati, J.F. and Neto, W.P.G., 2013. Document-level sentiment classification: An empirical comparison between SVM and ANN. *Expert Systems with Applications,* 40 (2), 621-633.

Muhammad, A., Wiratunga, N. and Lothian, R., 2016. Contextual sentiment analysis for social media genres. *Knowledge-Based Systems,* 108, 92-101.

Mukras, R. and Carroll, J., 2004. A comparison of machine learning techniques applied to sentiment classification. *R Mukras in Masters Thesis University of Sussex Falmer Brighton (2004),* .

Mumtaz, D. and Ahuja, B., 2016. A Lexical Approach for Opinion Mining in Twitter. *International Journal of Education and Management Engineering (IJEME),* 6 (4), 20.

Nakagawa, T., Inui, K. and Kurohashi, S., 2010. Dependency tree-based sentiment classification using CRFs with hidden variables. *In: Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics,* Association for Computational Linguistics, pp. 786-794.

Narayanan, R., Liu, B. and Choudhary, A., 2009. Sentiment analysis of conditional sentences. *In: Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing: Volume 1-Volume 1,* Association for Computational Linguistics, pp. 180-189.

Omitola, T., et al., 2014. Linking social, open, and enterprise data. *In: Proceedings of the 4th International Conference on Web Intelligence, Mining and Semantics (WIMS14),* ACM, pp. 41.

Omitola, T., Ríos, S.A. and Breslin, J.G., 2015. *Social semantic web mining.* Morgan & Claypool Publishers.

Pak, A. and Paroubek, P., 2010. Twitter as a corpus for sentiment analysis and opinion mining. *In: LREc,* .

Palanisamy, P., Yadav, V. and Elchuri, H., 2013. Serendio: Simple and Practical lexicon based approach to Sentiment Analysis. *In: proceedings of Second Joint Conference on Lexical and Computational Semantics,* pp. 543-548.

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

Pang, B. and Lee, L., 2005. Seeing stars: Exploiting class relationships for sentiment categorization with respect to rating scales. *In: Proceedings of the 43rd annual meeting on association for computational linguistics,* Association for Computational Linguistics, pp. 115-124.

Pang, B. and Lee, L., 2004. A sentimental education: Sentiment analysis using subjectivity summarization based on minimum cuts. *In: Proceedings of the 42nd annual meeting on Association for Computational Linguistics,* Association for Computational Linguistics, pp. 271.

Pang, B., Lee, L. and Vaithyanathan, S., 2002. Thumbs up?: sentiment classification using machine learning techniques. *In: Proceedings of the ACL-02 conference on Empirical methods in natural language processing-Volume 10,* Association for Computational Linguistics, pp. 79-86.

Penalver-Martinez, I., et al., 2014. Feature-based opinion mining through ontologies. *Expert Systems with Applications,* 41 (13), 5995-6008.

Peralta, V., 2007. *Extraction and Integration of Movielens and Imdb Data,* .

Polpinij, J. and Ghose, A.K., 2008. An ontology-based sentiment classification methodology for online consumer reviews. *In: Proceedings of the 2008 IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology-Volume 01,* IEEE Computer Society, pp. 518-524.

Poobana, S. and Sashi Rekha, K., 2015. Opinion Mining From Text Reviews Using Machine Learning Algorithm. *International Journal of Innovative Research in Computerand Communication Engineering,* 3 (3).

Poria, S., et al., 2014. Sentic patterns: Dependency-based rules for concept-level sentiment analysis. *Knowledge-Based Systems,* 69, 45-63.

Poria, S., et al., 2013. Enhanced SenticNet with affective labels for concept-based opinion mining. *IEEE Intelligent Systems,* 28 (2), 31-38.

Prabowo, R. and Thelwall, M., 2009. Sentiment analysis: A combined approach. *Journal of Informetrics,* 3 (2), 143-157.

Prud, E. and Seaborne, A., 2006. SPARQL query language for RDF.

Qiao, Z., et al., 2017. A domain oriented LDA model for mining product defects from online customer reviews.

Rebolledo, V.L., L'Huillier, G. and Velásquez, J.D., 2010. Web pattern extraction and storage. *In:* Web pattern extraction and storage. *Advanced Techniques in Web Intelligence-I.* Springer, 2010, pp. 49-77.

Riloff, E., Wiebe, J. and Wilson, T., 2003. Learning subjective nouns using extraction pattern bootstrapping. *In: Proceedings of the seventh conference on Natural language learning at HLT-NAACL 2003-Volume 4,* Association for Computational Linguistics, pp. 25-32.

Roncal, I.S.V. and Urizar, X., 2014. Looking for features for supervised tweet polarity classification.

Sacharin, V., Schlegel, K. and Scherer, K., 2012. Geneva Emotion Wheel rating study (Report). Geneva, Switzerland: University of Geneva. *Swiss Center for Affective Sciences, .*

Salton, G., Wong, A. and Yang, C., 1975. A vector space model for automatic indexing. *Communications of the ACM,* 18 (11), 613-620.

Samal, B., Behera, A.K. and Panda, M., 2017. Performance analysis of supervised machine learning techniques for sentiment analysis. *In: Sensing, Signal Processing and Security (ICSSS), 2017 Third International Conference on,* IEEE, pp. 128-133.

Sebastiani, F., 2002. Machine learning in automated text categorization. *ACM Computing Surveys (CSUR),* 34 (1), 1-47.

Shih, W., et al., 2018. Association rule mining of care targets from hospitalized dementia patients from a medical center in Taiwan. *Journal of Statistics and Management Systems,* 21 (7), 1299-1310.

Shridhar, M. and Parmar, M., 2017. Survey on association rule mining and its approaches.

Shubha, S. and Suresh, P., 2017. An efficient machine Learning Bayes Sentiment Classification method based on review comments. *In: Current Trends in Advanced Computing (ICCTAC), 2017 IEEE International Conference on,* IEEE, pp. 1-6.

Somprasertsri, G. and Lalitrojwong, P., 2010. Mining Feature-Opinion in Online Customer Reviews for Opinion Summarization. *J.Ucs,* 16 (6), 938-955.

Sulthana, A.R. and Subburaj, R., 2016. An Improvised Ontology based K-Means Clustering Approach for Classification of Customer Reviews. *Indian Journal of Science and Technology,* 9 (15).

Suryanarayana, D., et al., 2018. Natural Language Query to Formal Syntax for Querying Semantic Web Documents. *In:* Natural Language Query to Formal Syntax for Querying Semantic Web Documents. *Progress in Advanced Computing and Intelligent Engineering.* Springer, 2018, pp. 631-637.

Taboada, M., et al., 2011. Lexicon-based methods for sentiment analysis. *Computational Linguistics,* 37 (2), 267-307.

Tan, S. and Na, J., 2017. Mining Semantic Patterns for Sentiment Analysis of Product Reviews. *In: International Conference on Theory and Practice of Digital Libraries,* Springer, pp. 382-393.

Tang, D., Qin, B. and Liu, T., 2015. Deep learning for sentiment analysis: successful approaches and future challenges. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery,* 5 (6), 292-303.

Tang, H., Tan, S. and Cheng, X., 2009. A survey on sentiment detection of reviews. *Expert Systems with Applications,* 36 (7), 10760-10773.

Turney, P.D., 2002. Thumbs up or thumbs down?: semantic orientation applied to unsupervised classification of reviews. *In: Proceedings of the 40th annual meeting on association for computational linguistics,* Association for Computational Linguistics, pp. 417-424.

Turney, P.D. and Littman, M.L., 2003. Measuring praise and criticism: Inference of semantic orientation from association. *ACM Transactions on Information Systems (TOIS),* 21 (4), 315-346.

Vapnik, V., 1995. The nature of statistical learning theory Springer New York Google Scholar.

Vapnik, V., 2013. *The nature of statistical learning theory.* Springer science & business media.

Vapnik, V., 1998a. *Statistical learning theory. 1998.* Wiley, New York.

Vapnik, V., 1998b. The support vector method of function estimation. *In:* The support vector method of function estimation. *Nonlinear Modeling.* Springer, 1998b, pp. 55-85.

Vilares, D., Alonso, M.A. and Gómez-Rodríguez, C., 2015. A syntactic approach for opinion mining on Spanish reviews. *Natural Language Engineering,* 21 (1), 139-163.

Vilares, D., Alonso, M.Á and Gómez-Rodríguez, C., 2013. Supervised polarity classification of Spanish tweets based on linguistic knowledge. *In: Proceedings of the 2013 ACM symposium on Document engineering,* ACM, pp. 169-172.

Vinodhini, G. and Chandrasekaran, R., 2012. Sentiment analysis and opinion mining: a survey. *International Journal,* 2 (6), 282-292.

Wu, Y., et al., 2009. Phrase dependency parsing for opinion mining. *In: Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing: Volume 3-Volume 3,* Association for Computational Linguistics, pp. 1533-1541.

Xu, J., et al., 2017. Self-taught convolutional neural networks for short text clustering. *Neural Networks,* 88, 22-31.

Yang, C., et al., 2015. Research on the Sentiment analysis of customer reviews based on the ontology of phone. *In: Proc. Int ICEMCT Conf,* .

Yu, H. and Hatzivassiloglou, V., 2003. Towards answering opinion questions: Separating facts from opinions and identifying the polarity of opinion sentences. *In: Proceedings of the 2003 conference on Empirical methods in natural language processing,* Association for Computational Linguistics, pp. 129-136.

Zhao, L. and Li, C., 2009. Ontology Based Opinion Mining for Movie. *In: Knowledge Science, Engineering and Management: Third International Conference, KSEM 2009, Vienna, Austria, November 25-27, 2009, Proceedings,* Springer, pp. 204.

Zhou, L. and Chaovalit, P., 2008. Ontology-supported polarity mining. *Journal of the Association for Information Science and Technology,* 59 (1), 98-110.

Zhuang, L., Jing, F. and Zhu, X., 2006. Movie review mining and summarization. *In: Proceedings of the 15th ACM international conference on Information and knowledge management,* ACM, pp. 43-50.