Applied Master Thesis

Mind the gap: A comparison of linguistic vs deep-learning approaches to aspect extraction and aspect category detection

Peter Caine

a thesis submitted in partial fulfilment of the requirements for the degree of

MA Linguistics
(Text Mining)

Vrije Universiteit Amsterdam

Computational Lexicology and Terminology Lab
Department of Language and Communication
Faculty of Humanities

Supervised by: dr. E. Maks
2^{nd} reader: dr. Roser Morante

Submitted: June 30, 2020

Abstract

Aspect Extraction (AE) and Aspect Category Detection (ACD) form two crucial subtasks in Aspect Based Sentiment analysis (ABSA). Since the beginning of this approach to sentiment analysis, two approaches have dominated the area. Early researchers adopted linguistic approaches based on other basic NLP tasks, such as POS-tagging, parsing, rules or rankings mechanism (Marrese-Taylor and Matsuo, 2017). More recently, word embeddings have been employed as the primary features for input to deep learning (DL) neural networks. This thesis reviews studies and selects systems from both approaches to re-implement with the aim of evaluating the output to uncover qualitative differences. It was found that, while performance of DL systems was uniformly better on both tasks, inspection of the output revealed that the output of DL systems could often be unreliable and offer very few implementable strategies to improve performance, while the the simplest linguistic system was much more amenable to targeted strategies. Some strategies are explored for synthesising the strengths of both approaches to boost performance statistics, although questions remain about what can be measured with these metrics and whether pursuit of high scoring systems might ultimately not be the most meaningful aim.

Declaration of Authorship

I, Peter Caine, declare that this thesis, titled *Mind the gap: A comparison of linguistic vs deep-learning approaches to aspect extraction and aspect category detection* and the work presented in it are my own. I confirm that:

- This work was done wholly or mainly while in candidature for a degree degree at this University.
- Where any part of this thesis has previously been submitted for a degree or any other qualification at this University or any other institution, this has been clearly stated.
- Where I have consulted the published work of others, this is always clearly attributed.
- Where I have quoted from the work of others, the source is always given. With the exception of such quotations, this thesis is entirely my own work.
- I have acknowledged all main sources of help.
- Where the thesis is based on work done by myself jointly with others, I have made clear exactly what was done by others and what I have contributed myself.

Date: 16/07/2020

Signed: Peter Caine

Acknowledgments

To my wife, without whom this would have only been a dream;
To the teaching team at the VU for their ambition, hard work and willingness to share their expertise;
To my supervisor, dr. Maks, whose patience never faltered and whose guidance was invaluable in helping my complete my VU journey:

Thank you.

List of Figures

Contents

Chapter 1

Introduction

Research interest in sentiment analysis has taken off over the last 20 years. This has been fuelled variously by desire in political domains to track attitudes towards events in the news (Wiebe et al., 2005), a high level of cross-domain commercial interest, and access to huge quantities of opinionated textual data made available by the rapid adoption of social media technologies. Sentiment analysis also poses a series of interesting Natural Language Processing (NLP) which challenge researchers and stimulated interest (Liu, 2012).

Sentiment analysis presents issues involving a wide range of corollary NLP tasks such as part-of-speech (POS) tagging, word sense disambiguation, anaphora resolution, entity recognition and sarcasm detection among others, leading some researchers to refer to SA as a 'suitcase research problem' (Cambria et al., 2017).

One seemingly insurmountable challenge in sentiment analysis is attempting to determine and analyse algorithmically what Quirk et al.called 'private states', since by definition, private states are not open to objective observation or verification (Pang and Lee, 2008). Indeed, while early work on subjectivity showed that human annotators could achieve consistently high levels of inter-annotator agreement when annotating texts at the document level with tasks involving binary classification, when it came to more fine-grained analyses at the level of word or phrase, inter-annotator agreement was much lower. Disagreement was subject to several factors including the tag set used, part of speech of the word under analysis, identification of subjective element boundaries, and annotators own biases (Wiebe et al., 2001). It would thus seem optimistic to expect machines to perform well beyond a coarse-grained, document or sentiment level when human annotators diverge in their analysis.

We can understand why a coarse-grained, document or sentence level analysis might be considered insufficient by considering the following sentence from Liu (2012):

> although the service is not that great, I still love this restaurant

A sentence level analysis shows that the use of the subordinate conjunction creates emphasis on the clause containing the positive sentiment, thus imbuing the sentence with an overall positive sentiment. Yet simply labelling a sentence like this in binary terms (positive, opinionated, non-flame) leaves valuable information unaccounted for,

and much of the text unstructured. We can reasonably suppose that if we were to analyse any text of a length which exceeds one or two sentences, we would likely find a range of subjective ideas variously evaluating multiple targets. More preferable, then, would be a finer analysis where each of the concepts at which sentiment is directed is extracted together with its associated sentiment. An approach to sentiment analysis that does just that is known as Aspect-Based Sentiment Analysis (ABSA).

Much of the literature on this approach credits Hu and Liu (2004) with making the first concerted efforts to address sentiment analysis at this level. Rather than assume that documents address only one target for subjective analysis, they addressed the possibility of multiple possible targets within sentences. ABSA has since gone on to become an established task within sentiment analysis with its own sub-tasks, namely: aspect (term) extraction (AE), aspect polarity classification, aspect category detection (ACD) and category polarity classification (Pontiki et al., 2014). As an example, consider the below sentence:

> It has great sushi and even better wait staff.

We can evaluate the sentence as containing multiple aspects ('sushi' and 'wait staff') both with positive sentiment polarity (great, better). We can further generalise these aspects into categories '(food' and 'service'), with associated positive sentiment polarities inherited from the finer-grained aspects. Once performed, ABSA makes information in the text available for query or summarising.

While sentiment polarity analysis is integral to sentiment analysis, the sub-tasks of AE and ACD are valuable in and of themselves. Without knowing how people feel about a particular target of sentiment, it is also of value to determine what the sentiment is directed at; what it is that people value sufficiently to warrant critique or praise. Extracting the targets alone, however, is insufficient, since the range of facets related to particular aspects as well as the variance inherent in language renders the number of references to targets unwieldy. Thus, ACD also plays an important role in making sense of the range and variation of lexical terms.

The dual tasks of extraction and abstraction (generalisation) pose complex challenges and are viable as standalone tasks in their own right. The focus of this thesis is limited to these sub-tasks only; AE and ACD.

Approaches to ABSA typically fall into three categories: linguistic, statistical, and unsupervised (Marrese-Taylor and Matsuo, 2017). While initial approaches were almost entirely reliant on linguistic techniques such as POS tagging and parsing (Hu and Liu, 2004; Popescu and Etzioni, 2007; Raju et al., 2009; Qiu et al., 2011; Wagner et al., 2014; Kiritchenko et al., 2014; Toh and Wang, 2014), when evaluating state of the art work at the time of this writing, linguistic approaches have generally been supplanted by unsupervised, deep-learning (DL) approaches (Wang et al., 2016; Xu et al., 2018, 2019; Sun et al., 2019)which have all but dispensed with feature engineering. This would seem to indicate that a revolution has taken place in NLP where previously ubiquitous linguistic approaches have been entirely superseded by neural network techniques.

Yet, as Cambria et al. (2018)point out, machine learning is not without issues. Most systems require large quantities of domain-dependent training data, lack transparency, and can produce inconsistent results, and that machine learning is in essence "merely probabilistic ... only useful to make a 'good guess' based on past experience". They go on to quote Noam Chomsky's assertion that:

> "you do not get discoveries in the sciences by taking huge amounts of data, throwing them into a computer and doing statistical analysis of them: that's not the way you understand things, you have to have theoretical insights".

In the realm of text mining, theoretical insights are necessarily linguistic, however examination of literature regarding ABSA reveals that little attention is paid to linguistic approaches when researchers implement DL systems, and even authors who adopt a hybrid approach, do so in ways which suggest they are not trying to leverage strengths of one approach to compensate for weaknesses of the other.

The purpose of this thesis is to explore approaches which are based on insights from linguistic theory with modern, deep learning approaches to two of the sub-tasks associated with ABSA, namely aspect extraction (AE), and aspect category detection (ACD). The intention is to analyse the results of the contrasting approaches with a view to gaining insights into the strengths and weaknesses of the different approaches, and to explore the potential for strategic synthesis.

1.1 Research Question

The main research question is thus:

> *To what extent do results of linguistic-based text mining techniques differ from deep learning approaches for automatic extraction and categorisation of aspects, and what can the two approaches offer one another?*

This can be broken down into the following sub-questions:

1. What are common linguistic and DL approaches to AE and ACD?
2. How does performance of the two approaches compare?
3. To what extent does the output of linguistic approaches qualitatively differ to that of DL approaches?
4. What does the output suggest as areas where the different approaches could complement one another?

1.2 Outline

In the following sections, background information on key terms and expressions, metrics and datasets central to the concepts under review will be first introduced before Research Question 1 is addressed in the chapter on relevant literature, which will seek to document the most common techniques and evaluate the systems. The selection and modifications of appropriate systems on an established dataset will form the methodology section, and the results of their implementation will be used to address Research

Question 2. An in-depth error analysis will provide the foundation for the discussion section which will attempt to explore research questions 3 and 4. Any insights, conclusions and recommendations will be shared in the final section.

Chapter 2

Background information

This section introduces some key terms, expressions and background information required to access the remainder of the thesis. This will include definitions of the subtasks aspect extraction and aspect category detection as used in this writing; datasets which are referred to subsequently are also covered in order to put performance metrics into context.

2.1 Key terms and expressions

While the term, sentiment analysis, has been employed thus far to situate this thesis, an observation should be made regarding a term which is often used interchangeably with the term, sentiment analysis: 'opinion mining'. To accurately document the origins of the research area, it is appropriate to note a distinction. Paltoglou and Thelwall (2017) record that the term, sentiment analysis, as originally conceived was first used in a paper aiming to track market sentiment, and the phrase was quickly co-opted by the NLP community used when referring to determination of the polarity of a text. In contrast, the term, opinion mining, has its roots in a paper on web search and product reviews and was originally used to refer to the analysis of product attributes (Dave et al., 2003). For this reason, while the distinction may be subtle, it is perhaps more appropriate to refer to ABSA as an **opinion mining** technique.

The concept of ABSA as formally conceived is based on the idea of being able to decompose entities into a set of components, each with its own set of attributes (Zhang and Liu, 2014). An **entity** refers to a product, service, person, event, organisation, or topic. This definition at first glance seems broad, but given the nature of the domains in which aspect extraction can potentially be employed, this is unavoidable. Compare the following reviews from two domains:

> A comfortable flight, no issues. Meal was lovely, good choice of entertainment premium seating was spacious and comfortable, even with the seat in front of you fully down . Cabin crew had a positive attitude, served with a smile.

- **Domain:** air travel
- **Entity:** flight

- **Components:** Meal; entertainment; seating; cabin crew.
- **Attributes:** (cabin crew) attitude

> The battery life has not decreased since I bought it, so i'm thrilled with that. The price and features more than met my needs.

- **Domain:** electronics
- **Entity:** laptop
- **Components:** battery; price; features.
- **Attributes:** (battery) life

Entities can refer to both tangibles (laptops) and intangibles (flights), and their components may or may not include attributes. In practice, components and attributes are treated together and referred to as **'aspects'** of the entity, thus an **aspect expression** or **aspect term** is any textual representation (word or phrase) that indicates the component or attribute.

Aspects can be both *implicit* and *explicit*. In the following example from Hu and Liu (2004), we can infer a reference to two aspects (weight and size) which have no concrete representation in the text. Weight and size are thus implicit aspects.

> "While light, it will not easily fit in pockets."

Aspects can be indicated by a range of words and expressions, but it is commonly accepted that explicit aspects are realised by nouns and noun phrases, while implicit aspects can take many forms, but commonly adjectival or adverbial in nature as in example above. This thesis is only concerned with extraction of explicit aspects.

For the purposes of this paper, when using the word, entity, we refer to the highest-level, general concept within a domain, and all explicit components and attributes which manifest as nouns or noun phrases will be aspects.

Thus, as a formal definition for this paper, **aspect (term) extraction** (AE) can be said to involve the identification of explicit nouns and noun phrases describing components or attributes of a general entity within a specific domain.

It is common for there to be a high degree of variation in noun phrases referencing identical aspects. Considering the number of ways that different kinds of food might be referenced, naming menu items alone would render an extremely long list; add colloquial expressions, regional dish variations and combinations and descriptive adjectives ('cold', 'spicy', 'Asian'), and the number of permutations becomes unwieldy. By grouping these aspects together into categories, it is possible to make this variation more manageable, thereby making processed texts more structured.

ACD involves the attempt to aggregate similar explicit aspects into a limited number of common, more generalised, and commonly agreed upon categories. In this way,

similar items which vary in the way they are described can become directly comparable.

NLP tasks can be conceived of in a variety of ways; as binary classification problems, categorical classification problems, sequence labelling problems, sequence to vector (encoder) sequence to sequence (transducer) and so on. While sentiment analysis can be treated in all of these ways, ACD lends itself to a multi-class, document labelling task, whereas AE is generally treated as a sequence labelling task.

A sequence labelling task is a task in which each token in a review receives a class label. Aspect extraction for example can be conceived of by thinking of a word in a sequence as being in one of two states, either part of an aspect or not part of an aspect. It is useful to denote boundaries where new aspects begin so that adjacent, separate aspects are not confused and thought of as a single aspect. A common labelling system for this is IOB (B= Beginning, I = Inside and O= Outside); any tokens which are not considered to be part of an aspect are labelled O, whereas single tokens – aspects represented by one token are marked B. For multi-word aspects, the first token is labelled 'B' and subsequent tokens are labelled 'I'. In this manner, a label for each token is produced.

2.2 Performance metrics

The following literature review chapter introduces and describes previously published research covering approaches to the sub-tasks of AE and ACD. Throughout the section, regular reference to three system performance metrics will be made, namely: precision, recall and f1. While precision, recall and f1 are useful metrics to evaluate and compare systems, the way they are used and referenced through the literature is not uniform and does raise some issues. Quoting results for partial matches instead of exact matches (regarding boundaries of aspect terms) can skew results heavily, yet very few authors discuss which method they adopt when publishing results. Some researchers only quote f1 statistics, which is useful for a quick and dirty comparison with competing systems, but not helpful in detailed analysis of underlying biases of the systems.

Many systems described compare their system performance with that of earlier published work. However, a degree of scepticism is warranted with published performance statistics. One study which sought to replicate 'well-known algorithms for syntactic centric aspect-based opinion mining' (Marrese-Taylor and Matsuo, 2017) Their study covered two of the systems introduced in the literature review below (Hu and Liu, 2004; Qiu et al., 2011). However, their attempts failed to replicate performance by a significant margin in both cases. Studies like this which fall well short of being able to replicate system results mean not only that the systems studied are of questionable validity, but the subsequent systems which claim to have repeated those results in order to compare with their own system performance are also called into question.

2.3 Datasets

In order to be able to put system results into context, it is important to keep in mind that different systems use different datasets, annotated according to non-uniform guide-

lines. This means that results are often not directly comparable, and metrics given are just an indication of performance.

Most of the datasets in the systems reviewed use subjective texts taken from one of two domains: electronics or restaurants. That is not to say that these are the only domains on which ABSA is useful, but this does provide a common backdrop against which to compare systems. Of the systems described below, there are largely three datasets which are prevalent: (Hu and Liu, 2004)[1], SemEval 2014 (Pontiki et al., 2014), SemEval 2016 (Pontiki et al., 2015, 2016)[2].

2.3.1 Hu and Liu dataset

This dataset [3] comprises 100 reviews each on 2 digital cameras, 1 DVD player, 1 mp3 player and 1 mobile phone scraped from two websites. This data was annotated by the two authors who tagged both explicit (the majority) and implicit features (aspects) and using a numerical value for sentiment. The authors noted that they found some difficulty in annotating sentiment, but little difficulty in annotating aspects. Aspect categories were not annotated in this dataset.

2.3.2 SemEval 2014 dataset

ABSA was introduced for the first time for SemEval 2014 (Task 4). It is divided into two distinct datasets: one for laptops and one for restaurants. The datasets are fragments of real-world reviews adapted from Ganu et al. (2009), which was originally annotated solely for categories. There were originally six categories, however these were subsequently re-annotated as 5 categories since the annotators found two categories, miscellaneous and anecdotes, were found to be difficult to distinguish and were merged into one. This category became the default label; i.e. if an item was to be labelled as an aspect, yet the term/phrase did not fit the other four categories, the aspect would be annotated as 'anecdote/miscellaneous'.

Annotations for aspects were subsequently added by two annotators: a graduate student and a linguist. In areas of disagreement, they would either decide collaboratively or by recourse to a third expert annotator. Areas of disagreement mostly stemmed from multi-word aspect term boundaries, or distinctions between the aspect term compared to the target entity. Since the definition of aspect as 'a component or attribute of an entity' excludes the entity itself, this was a crucial distinction. An example of this latter issue is given below:

> This place is awesome (reference to the restaurant as entity)
> Cozy place and good pizza (place referencing the ambience as aspect category, rather than as entity).

Curiously, verbs or verbals ('words formed from a verb, but functioning as a different part of speech e.g., gerunds and participles') were also considered to be aspects (see

[1] Available at: `https://www.cs.uic.edu/~liub/FBS/sentiment-analysis.html`
[2] All SemEval datasets referenced can be found at: META-SHARE (`http://www.metashare.org/`)
[3] Available: `https://www.cs.uic.edu/~liub/FBS/sentiment-analysis.html`

examples below). Pronouns were not annotated at all (considered to be implicit aspects.

i. Fresh , delicious , and reasonably **priced**
ii. It is pretty sweet when you want **gaming** on the laptop

A separate test-set was constructed from scratch using the same annotation guidelines[4].

A notable difference between the annotations for aspect terms versus aspect categories is that aspect terms were required to be explicitly realised in the text, and even pronominal reference to an entity violated the annotation guidelines. However when it came to aspect categories, this was not the case, and categories could be (and often were) implied as demonstrated in this instruction in the annotation guidelines:

> "Notice that aspect categories may not necessarily occur as terms in the sentence e.g., the sentence "Anybody who likes this place must be from a different planet, where greasy, dry, tasteless and complimentary" discusses the aspect category food, without mentioning particular aspect terms related to the food."
>
> (Quote taken from the guidelines referenced in footnote 4)

2.3.3 SemEval 2016 dataset

In its second year, 2015, the ABSA task (Task 12) was expanded to include reviews from the hotel domains, although this constituted an out-of-domain task since no training data was provided. The training and test sets for both laptops and restaurants was also modified, with the following amendments:

- Full reviews were used instead of review fragments. The full review was to be taken into consideration when identifying targets rather than that at sentence level.
- A change to the definition of aspect category was made. A category was now defined as a combination of both an entity (one of food, drink, service, ambiance, location, restaurant) and an attribute type (general, prices, quality, style & options, miscellaneous). The task was to extract the aspect category (the entity and attribute pair), aspect term and opinion at once, presented in one tuple.
- Implicit reference to an entity referred to through pronouns meant that the tuple was still extracted, but the entity was to be assigned the value of 'Null'.
- References to entities outside of those under review (e.g. other restaurants) were considered to be 'out of scope' and not annotated.

This time an experienced linguist was the main annotator and a second linguist inspected the annotations. Problems still persisted with the assignment of some category labels. These were resolved in collaboration (Pontiki et al., 2015) again with a third

[4]Annotation guidelines available at: http://alt.qcri.org/semeval2014/task4/data/uploads/

expert called in when agreement could not be reached. In 2016, the task (Task 5) was expanded to include multiple languages, but the English language datasets remained the same as in 2015(Pontiki et al., 2016).

The changes implemented meant that, while ostensibly the same task was being conducted, systems found it much harder to perform according to the new parameters, and state-of-the-art performance metrics were around ten points lower as a result.

2.3.4 Other datasets

Raju et al. (2009) used four datasets each from a separate domain (cell phone, microwave, watch and portable DVD player) containing 25 product descriptions collected from www.amazon.com website. Zhai et al. (2010) attempted to demonstrate the generality of their method and used a broad range of domains for their datasets (hometheater, insurance, mattress, car and vacuum) obtained from an opinion mining company.
In both cases above, it is assumed that the authors performed that annotation themselves, although no further information about the annotation process was provided. These datasets do not feature strongly in this thesis.

Chapter 3

Literature review

This chapter will seek to address Research Question 1; common approaches to AE and ACD and systems that use them. First, AE systems exploiting purely linguistic systems will be reviewed. This is followed by a short section on types of neural networks referenced in the following section on deep learning based systems. The final sections will cover linguistic and deep learning approaches to the ACD subtask.

3.1 Aspect Extraction (AE) systems and approaches

3.1.1 Systems using linguistic information

With explicit aspects conceptualised in grammatical terms (nouns and noun phrases) it would seem that the task of extraction of nouns and noun phrases would be technically little more than a matter of identifying nouns and noun phrases in a text. Hu and Liu's (2004) seminal paper, however, explains that extracting based solely on parts-of-speech (POS), a strategy previously employed in terminology finding, produces too many non-terms (negatively affecting precision). They employed instead a technique known as association mining to identify frequent itemsets (unordered collocations) exploiting the observation that despite variance inherent in unstructured texts, when discussing aspects relating to a common domain entity, expressions tend to converge. The itemsets discovered through this process, they claim are likely to be product features (aspects).

According to the authors, their approach also suffers from issues with precision, so two pruning techniques are employed post-extraction to improve this metric: compactness pruning (based on order of words in phrases) and redundancy pruning, single words that are likely only part of phrases and not aspects unto themselves (such as the word 'life' in 'battery life'). They use known sentiment-carrying adjectives to identify potential aspect-carrying sentences, and extract aspects if they are found to be a member of the frequent itemset. If there are no frequent features (noun phrases in the itemset) then the nearest proximal noun is labelled as a feature. They claim that after all steps are performed including pruning, that 80% of all features were identified (recall = 0.8) and of the features that the system identified, 72% were correctly identified (precision = 0.72).

Later research by Popescu and Etzioni (2007) extended Hu and Liu's research using

Pointwise Mutual Information (PMI[1]) claiming to make significant precision improvements on Hu and Liu's results. PMI is employed twice in their explicit aspect extraction method: once to compare extracted features to meronomy indicators of the product class (e.g. class comes with + meronym; class is equipped with + meronym; meronym + of class), and once again to compare against general word co-occurrence probabilities using the internet as a general corpus. This, they claim, contributed to a 21% increase in precision on Hu and Liu's results above, albeit with a marginal decrease in recall.

The constituency-based approach of Raju et al. (2009) began with extracting all noun phrases then pruning the results based on the presence of determiners or frequency comparison with a general corpus in a step they call pre-processing. They cluster the noun-phrases before performing statistical operations to extract the aspect; they assume the right-most word in a noun phrase to be the head and extract that as the aspect. They report of 92% precision and 62% recall (p = 0.92; r = 0.62) however these are for partial matches. Statistics for full matches are rather less impressive with precision dropping precipitously (p = 0.52). Worryingly, as they experimented with corpora of various sizes, their results for full match precision drops as the corpus grows in size.

In addition to constituency-based analyses, dependency parsing has also been leveraged to identify and extract aspects. The main insight is that since dependencies show relations between words, and we know that opinion words often modify these nouns in particular dependency relations, that we could use known opinion words plus their heads to identify features. This concept was extended by Qiu et al. (2011) who used a given seed set of opinion words to identify target aspects. Extending Hu and Liu's insight that language on aspects tends to converge and repeat, their system assumed that once an aspect had been extracted it could be used to identify other potential opinion words by analysing subsequent encounters with the term in the corpus for novel opinion words. Once the set of opinion seed words had expanded, subsequent iterations over the dataset could further extract target aspects which could once again be sued to expand the set of opinion seed words, and so on. This method of using known opinion words to identify aspects exploiting dependency information of the extracted aspects to identify new opinion words is called Double Propagation (DP).

They employed a method of pruning based on the observation that within clauses, unless a conjunction was present (such as 'and'), there was likely to be only one aspect and any other identified nouns were likely to be noise. They once again used the frequency information from Hu and Liu to discard nouns that were less frequent (assuming them not to be aspects). Unlike a previous attempt to extract aspects using dependency information which tried to extract noun phrases simultaneously (Wu et al., 2009), Qiu et al.'s approach extracted only single nouns initially and expanded their results to noun phrases by looking for particular clause constructions (combinations of 2 consecutive nouns and 1 adjective) in a post-processing step. Published results were p = 0.88, r = 0.83.

The DP concept was further extended by Liu et al. (2013), who observed that by

[1]PMI is a measurement of co-occurrence for two terms, comparing the frequency of co-occurence with what we would expect if they were independent (Jurafsky and Martin, 2008)

pruning general words (e.g. 'things', 'place', 'day' identified in WordNet) which were extracted by default by the DP algorithm, they were able to increase precision without affecting recall. Their implementation of DP differed slightly to Qiu et al. (different parser and a different pruning method which increased recall scores). They recorded the following results: p = 0.88; r= 0.87.

One final linguistic system to introduce is that of that of Toh and Wang (2014), who performed a study using a diverse range of linguistic information. Among the features were: the word itself, a name list extracted from a gazetteer, part-of-speech tag, dependency head and the POS of the head word, particular dependency relations, word clusters trained from a range of domain-related corpora and they also used a modified version of the DP algorithm to create another feature list with a CRF[2] classifier. Wang's results show recall and precision scores of 83% and 85%, respectively. They performed a subsequent ablation study removing one feature at a time to explore the contribution of each feature to the task. They discovered the omission of POS and dependency relation information to be detrimental when other features were in place, thus confirming their centrality to the task.

In the following section a brief overview of Neural Network-based systems and their applications to NLP is given, followed by a brief description of a systems which employ these techniques for AE.

3.1.2 Deep learning and neural network models

Deep Learning is predicated on the idea of neural networks; modules of computational units which take some numerical input either from a previous group of units, a layer, or from input, and after performing some mathematical operation (usually a weighted sum of the inputs together with a static 'bias' value), produces output which can be the output of the system or input for the next module layer. A network with multiple layers is considered to be 'deep' since, once it has been initiated, it is left to its own devices.

The idea of neural networks is not new. Their origins lie in ideas of artificial neurons that go back to 1940's (Jurafsky and Martin, 2019) While traditional machine learning techniques required extensive feature engineering to transform input data into a vector representation suitable for a machine to be able to identify patterns, deep learning allows machines to be fed raw data from which suitable representations could be learned LeCun et al. (2015).

Back-propagation algorithms (Rumelhart et al., 1986) update the network iteratively during learning which lead to updated weights, modifying the calculations, producing new outputs, and the process repeats for a predefined number of iterations (epochs). Crucially, this back-propagation can eliminate the need for feature extraction (Goldberg, 2016).

In effect, what each layer does is output some abstraction of the input, creating an abstracted representation. Ideally, information which is relevant to determining some predictive pattern is amplified while information in the input which is irrelevant is su-

[2]which computes probability for a sequence over a set of tag sequences (Jurafsky and Martin, 2019)

pressed (Miller et al.).

Although the purpose of this paper is not to provide an in-depth coverage of Neural Network (NN) architectures a brief description of the systems referenced is useful to understand the motivation for selecting different architectures in systems described. What follows is a description of a fully connected NN to serve as a baseline for comparison and introduce issues in ABSA that the variants subsequently introduced (CNN, (Bi-)RNN, (Bi-)LSTM) attempt to address. As such, descriptions will be generally light, with their relevance to the task of ABSA the primary focus.

Fully connected feed forward models

A 'plain vanilla' neural network, is a network in which all neurons in one layer can pass output to every neuron in the next layer. Each neuron performs a calculation over the inputs using the weights passed as input and outputs the result of that calculation to each neuron in the next layer. Thus, two layers each with 10 neurons perform 100 (102) weights calculations. A non-linear function is typically applied to provide a 'squashed' output (an output usually in the range -1 to 1), which is passed as input for the next layer. The output layer typically has the same number of computational units as the number of classes that are to be predicted, which in the case of an IOB, sequence-labelling task would be three. Outputs of systems tend to be real number vectors which need to be interpreted as probability distributions over these classes, i.e. converted to a set of numbers between 0 and 1 which sum to 1 (Jurafsky and Martin, 2019). What is most commonly used in the systems described below is the softmax activation function.

Each layer between the input layer and the output layer is considered to be a 'hidden layer'. During training, output of these calculation is compared with desired output, loss is computed , back-propagation updates the weights during training to help condition the network to the desired output whereupon the process is repeated for multiple epochs.

Neural networks do not (generally) accept input of variable size, which is problematic for NLP since documents, sentences and phrases have no fixed length. One solution known as the continuous bag of words (CBOW) model is to project inputs to a projection layer where their vectors are averaged to produce fixed-sized representations for the input (Mikolov et al., 2013).This approach tries to capture all relevant aspects at once, and as with BoW, information encoded in the order of information is lost. This problem with this is clear when we consider that the following sentences (from Jurafsky and Martin (2019)) would be treated as equivalent.

> "it was not good, it was actually quite bad"
> "it was not bad, it was actually quite good"

Convolutional neural network

A solution proposed by LeCun et al. (1995) to issues with standard fully connected neural networks with regards to data rich input like images and speech recognition was

the use of Convolutional Neural Networks (CNN). While not initially applied to NLP, issues such as the above described indifference to positional together with other issues such as the cumbersome number of weights generated, overfitting to the training data, lack of built-in invariance (ability to adjust to variation in the input) were relevant, and a CNN approach was adapted to NLP by researchers such as Collobert et al. (2011).

CNN work by sliding a fixed-size multidimensional 'windows' (filters) over groups of input such as pixels in an image transforming the group into a single vector representation. The filter proceeds stepwise over the rest of the input performing the same generalising function. In contrast to image processing applications which use a 2-dimensional convolution, in NLP, 1-dimensional windows can 'slide' over representations for groups of words (say 5 words at a time). These windows are typically combined in a 'pooling' layer which seeks to maximise the important features from a number of windows into a more abstracted vector representation. Max-pooling offers the twin benefits of amplifying relevant features and ignoring irrelevant features and producing a fixed-length output.

By processing groups of words sequentially in an overlapping pattern allows some information about local word order, as an automatically constructed n-gram, to be captured. If information within a short distance prove useful as features for the classification task, then CNN's are particularly useful (Goldberg, 2016). It is for this reason that CNNs lends themselves well to a task like AE which is a sub-sentence-level task, often identified by short-range dependency relationships with opinion words.

However, while local dependency relations can capture many aspect terms, not all information is local. An adaptation of the FFNN, the Recurrent Neural Network (RNN) is employed which offers the benefits of being able to capture longer range relations as well being able to accommodate arbitrary fixed-length inputs.

(Bi-directional) recurrent neural network

As mentioned above, a standard FFNN requires input of uniform length (dimensionality) and information is passed one way (forward) through the network. In an RNN, by contrast, the output of the hidden layer is split, with one passed to the next hidden layer as before, while the other is recycled back into the same layer as a new input to be processed together with additional input. This allows a kind of memory of previous input to stay within the processing unit Jurafsky and Martin (2019) enabling the system to process new information while evaluating all previous information going back to the beginning of the sequence since every time a new token is processed, information from preceding computations can be taken into account. There are different kinds of RNN, the most common being the Elman-type RNN, which takes as memory input information from the hidden layer of the previous time step; and the Jordan-type RNN, which takes information from the hidden layer of the previous time step as memory input component Liu et al. (2015).

RNNs typically function as transducers, producing an output for each input (Goldberg, 2016). This capacity of producing sequence to sequence vector representations suit sequence labelling tasks like AE. RNN's also offer more flexibility than this, however, and they have been put to creative use. Since variable length input is a possibility,

it is possible to take the final output to represent the entire sequence of inputs (as a sequence to vector encoder), which can be used at document level of sentiment analysis tasks also (like ACD). These outputs can then be used as inputs to other NN architecture such as a FFNN on a down-stream task, performance on that task providing the information used to update all the weights.

The memory effect allows memory of previously processed information, but some researchers realised that processing sequences from both beginning to end and a separate RNN processing from end to beginning of a given sequence would allow information to be taken into account from both the beginning of the sequence as well as the end. These bi-directional RNNs produce two vector representations which can be combined in a variety of ways to capture information across the whole sequence (Jurafsky and Martin, 2019).

These solutions come at a cost, however. In practice, RNNs suffer from an issue of compounding multiplication of the weights as the sequence processing continues and errors are back-propagated through time. This constant multiplication can quickly mean that very small floats decrease exponentially, approaching zero (vanish)(Bengio et al., 1994) while floats larger than 1 increase exponentially (explode) in a process that is somewhat analogous to magnification of small genetic issues through multiplication of normally recessive genes in cases of inbreeding. Neurons which output zero can no longer contribute to computations, and neurons which do so are 'dead' for the remainder of the training.

Subsequent models sought to overcome these limitations. One solution coming from a new type of architecture.

(Bi-directional) Long short term memory

The memory echo of the cycling output in RNN results in repeated multiplications in training. Ideally the memory effect would be more selective, retaining only that which is important to prediction on the task, while somehow able to forget that which is not useful in training. One solution is called a Long-Short Term Memory network (LSTM). LSTM's accomplish the selective memory task via the use of an extra 'context' layer and the use of 'gates'. Gates are essentially mathematical functions that simulate logical gates outputting a 1 (that allows information to pass) or 0 which inhibits the information (Goldberg, 2016). The LSTM block is a complex unit which consists of a memory cell with a self-connection (as with the RNN), together with an input gate to selectively control the input signal, an output gate to control the effect of the neuron activation, and a forget gate allowing the neuron to reset its state through the self-connection (Liu et al., 2015).

The LSTM architecture allows subtle changes to be made in training making back-propagation look crude by example. The concept has also been extended like the bi-directional architecture for RNN. Bi-LSTM allows information from previous tokens and information from following tokens to be taken into consideration when performing computations, but as is becoming a pattern in the evolution of these networks, the complexity of the system creates undesirable outcomes such as computational expense and lack of transparency. Further innovations have taken place with a more simpli-

fied architecture (GRU), and more powerfully, attention mechanisms in transformer architecture which will be briefly introduced in the next section.

3.1.3 Deep learning systems (and their input)

Traditional methods of vectorising linguistic input rely on one-hot encoding, the creation of an n-dimensional vector with zeroes representing each linguistic feature. For NLP, each word in the vocabulary could be used as a feature, however this created high dimensional sparse vectors (vectors of mostly zeroes up to thousands of dimensions long) which do not keep information about word order, reducing documents and corpora as bags of words (BoW). This loss of syntactic information can be counteracted with the use of n-grams to some extent; however this causes the vectors to have exponentially more dimensions which are even more sparse. Unfortunately, deep neural nets do not perform well with large sparce vectors (Goldberg, 2016), so one conceptual jump which accelerated the use of Neural Networks in NLP was the introduction dense vectors which project words into much smaller vector spaces, word embeddings.

Depending on the way the embeddings are trained, both syntactic and semantic information can be captured, allowing not only the reduction of representation of words, but also the reduced need for linguistic feature engineering. One of the first word embedding models was that of Bengio et al. (2003); numerous other models soon followed (Mikolov et al., 2013; Pennington et al., 2014). Two techniques are common when training embeddings: a continuous bag-of-words based model (CBOW) that infers word context and the skip-gram model that predicts words from neighbouring words. Resultant embeddings are a manifestation of the observation that words that are similar in meaning occur in similar contexts, lending empirical support to Firth's distributional hypothesis (1957).

There are many variants of embeddings trained in different ways or trained on different corpora. These include Collobert et al.'s SENNA embeddings (2011) trained on Wikipedia text, Stanford University's GloVe embeddings trained on word co-occurrence, Facebook's fastText using a combination of skip-gram and character n-grams to assist in out of vocabulary representations Trusca et al. (2020), and even sentiment-specific embeddings (SSWE) (Do et al., 2019).

With not only dense vector representations with the capacity to encode both semantic and syntactic information made commonly available to NLP researchers, but also the tools with which to custom train embeddings also available for use in domain-specific corpora, the stage was set for a sea change in NLP approaches to common tasks to be able to add semantic information.

Xu et al.'s 2018 aspect extraction system used general embeddings in conjunction with custom trained word embeddings trained on a domain-specific corpus. 100-dimensional, domain-specific embeddings were concatenated with 300-dimensional Word2vec embeddings to represent each word in reviews. This served as input for a 4-layer convolutional neural network with a fully connected output layer with a softmax activation function. The combined effect of general embeddings pretrained on billions of words (thus offering great generalisability) coupled with the additional information from training custom embeddings within a domain meant that this simple CNN system (which

did not include max pooling layers), even when trained on a relatively small amount of data, gave state-of-the-art (SOTA) performance (f1 = 74.37 for the SemEval 2016 restaurant dataset).

Very recently as of the time of writing, a new type of network architecture, transformers, has emerged. Transformers utilise attention mechanisms, where different 'heads' attend to a variety of information in the input, a method of training involving masking parts of the input and an innovative way of encoding positional information which allow long-range dependencies to be captured saw instant improvements on a range of NLP tasks (Vaswani et al., 2017). Originally designed to mask subsequent information in the sequence to only attend to previously occurring input in the sequence, this was considered a weakness for some language tasks like question and answering and Devlin et al. (2018) extended the idea to use bi-directional information with random masking of select tokens during training, together with an additional task, next sentence prediction, conditioned to be sensitive to 'context' both to the left and to the right in training. These new contextualised pre-trained Bidirectional Encoder Representations from Transformers (BERT) embeddings showed immediate SOTA results on eleven separate NLP tasks at the time of its publication. While ABSA was not among them, it did not take long for researchers to implement these for AE.

Xu et al. (2019) employed pre-trained BERT representations in their AE system. Tokenised text, vectorised using the BERT model was output directly to a fully connected layer with a softmax function in a FFNN system that could hardly be simpler. Given the success of standard BERT embeddings in other NLP tasks, the authors were initially disappointed in the performance on the review task (f1 = 0.74). Further fine tuning on 700,000 reviews in a related dataset[3] saw this addition of domain knowledge increase performance (f1 = 0.78) on a the SemEval 2016 dataset.

These simple systems showcase the power of dense contextual representations alone. Not all systems keep linguistic and DL techniques apart. There have been several attempts to combine these approaches for AE.

3.1.4 Hybrid systems

Liu et al. (2015) included a simplified set of linguistic features including 4-dimensional POS tags (noun, adjective, verb, adverb), and BIO tagged chunk information as binary features (NP, VP, PP, ADJP, ADVP) for linguistic information and compared a range of NN architectures (Elmann-type RNN, Jordan-type RNN, Bi-directional RNN and LSTM) with a range of embeddings (Google, SENNA, and embeddings custom trained on Amazon review data). Their best testing system on the SemEval 2014 dataset scored 0.82 (f1) using custom trained embeddings on Amazon reviews in Bi-directional RNN. This was slightly below the state of the art at the time.

Poria et al. (2016) combined custom embeddings trained on Amazon opinionated reviews combined with 6-dimensional POS encoded vectors (noun, verb, adjective, adverb, preposition, conjunction), and five linguistic rules. The embedding and vectorised POS information was run on a CNN network and the rules separately, and all terms

[3] `https://www.yelp.com/dataset/challenge`

marked as aspects by either of the two systems were combined and labelled as aspects. Their results showed that the POS information added to word embeddings increased recall from 84% to 85% while precision stayed the same (87%). Inclusion of the rules saw recall increase to 86% and precision increase to 88%. They reported that removal of stop words (high frequency words which contribute little semantic value, such as 'a', 'we'), and a rule which identified a target aspect in a dependency relation with a known opinion word, and another rule (extracting noun complements of a copula verbs as aspects) to contribute the most to results. They also employed this system on Hu and Liu's dataset recording precision and recall scores of 90% and 86% respectively.

Toh and Su (2016) performed an ablation study on both subtasks of AE and ACD. For AE they used a classical approach of engineered features and a CRF classifier. Their innovation was to use the output of a separately trained Bi-RNN as an additional feature in the CRF classifier. The list of features used was extensive and included both the word and bigram, frequent aspects and frequent words that occurred in aspects identified in the training data, dependency heads of each token, custom trained word embeddings (using both Word2Vec and GloVe training tools) from related unlabelled corpora (Amazon and Yelp); embedding clusters (K-means) and an implementation of Qiu et al.'s DP as candidate aspects. Each additional feature contributing a few percentage points to the system performance. The largest performance gains were provided by the additional feature of the Bi-RNN probabilities trained on the same embeddings, the name lists extracted from training and word clusters. Theirs was the leading system on the SemEval 2016 dataset at the time of publication (p = 0.75; r = 0.69).

Most recently, Ray and Chakrabarti (2019) sought to combine a CNN classifier with POS information and several syntactic rules. They also use the Word2vec training tool to create embeddings as the sole input for a CNN with 2 convolutional layers and 2 max pooling layers trained on the SemEval 2014 dataset. As with Poria et al. above. They ran each system separately and combined the results of both processes reporting an increase of p=0.78 and r = 0.85 to p=0.8 and r = 0.86 when adding output from the rules to the output of the CNN on the SemEval 2014 restaurant dataset.

As mentioned in 2.1, despite Hu and Liu's claim that users tend to use the same terms when referring to aspects (2004), aspect terms are prone to variation. This is in part due to the number of aspects to be evaluated, and in part due to the inherent variation in language use. One common application of ABSA is for potential consumers to be able to evaluate comparable products before making a purchase. It is beneficial to be able to group similar aspects together to be able to make direct comparisons between related entities. This task and systems employing both linguistic and DL approaches will be introduce in the section below.

3.2 Aspect Category Detection (ACD) systems and approaches

According to the guidelines laid out in the SemEval 2014 task, the definition of ACD involve identifying "a predefined set of aspect categories ... [which] do not necessarily occur as terms in the given sentence." (`http://alt.qcri.org/semeval2014/task4/`).

Whereas AE is concerned with identifying explicit lexical references, ACD is a task concerned with abstraction of concepts to a more generalised concept into which they can be subsumed. When it comes to ACD in practice, statistical models such as Latent Dirichlet Allocation (LDA) dominate (e.g. Lin and He, 2009.; Guo et al., 2010). Topic modelling methods usually perform both aspect extraction and categorization simultaneously as individual terms are clustered in a document collection. However statistical techniques are beyond the scope of this paper and the rest of this section will be devoted to the few examples of non-statistical, linguistic and deep-learning approaches.

3.2.1 Systems which rely on linguistic information

Several linguistic techniques involve the use of WordNet to group aspects together based on similarity metrics. WordNet is "an on-line lexical reference system ... [in which] ... English nouns, verbs, and adjectives are organized into synonym sets, each representing one underlying lexical concept" (Miller et al.). In contrast to a dictionary, lexemes are organised according to meaning. In addition to organising information in terms of synonymy and antonymy, crucially for the task of clustering aspects, nouns are also organised in terms of hyponymy/hypernymy (showing hierarchical relations between general word terms and specific word terms such as brother is a kind of relative and relative is a kind of person) and meronymy/holonymy (showing part, whole relations such as a relative is a part of a family; a leg is part of a person) (Miller, 1993).

Organisation of (predominantly) nouns in this way allows for calculations to be made on similarity of words by using hierarchical relations and measuring the paths between synsets in imaginative ways such as the 'distance' from both words to a common subsumer, the depth in the path from the global root entity, the relation of the depth to the maximum depth of the taxonomy, the number of hyponyms available and the information content shared (Meng et al., 2013).

Carenini et al. (2005) developed a system for category detection that combined the output of Hu and Liu's (2004) unsupervised association mining technique used as input for a supervised learning system with user-defined features to create 'crude features'. Classification was refined by using WordNet lexical distances, and (lemmatised) string matching. Their results involved a custom dataset, user defined features and metrics which do not allow for direct comparison with competing systems (placement distance and redundancy reduction scores).

Zhai et al. (2010)exploit lexical similarity in WordNet and the fact that aspects in the same category often: share words (e.g. battery and battery power) as 'soft constraints' which could be relaxed during training. They coupled this with Expectation-Maximization (EM[4]) algorithm and a naive Bayes classifier. Their system required a small number of user defined seeds (one per category) for the system to begin. The classifier assigns all labelled and unlabelled examples to a class. Once again, custom datasets were used involving five domains: Home theater, Insurance, Mattress, Car and Vacuum. A comparison of their approach with other systems including K-means

[4]EM trains a classifier with labelled documents first, and uses the classifier to assign probabilistic labels to unlabelled documents then trains a new classifier using all the maximizing the likelihood of all data (Nigam et al., 2000)

clustering and LDA variants showed their approach to be more effective however, once again metrics used do not allow for direct comparison with competing systems (entropy and purity).

Brun et al.'s (2014) system employed output from a custom parser and a bag of words to assign aspect categories at the sentence level. The parser output included tokenization, morphosyntactic analysis, POS tagging, Named Entity Detection, chunking and, extraction of dependency relations annotated with 'deep syntactic functions' in which a predicate is linked with its 'deep subject (SUBJ-N), its deep object (OBJ-N), and modifiers' and some semantic encoding. A list of domain words taken from the training data and extended with WordNet synonyms and food terms from Wikipedia, their frequencies calculated and reweighted if they contained an on-list word, were features used in a logistic regression classifier. A threshold was set (¿0.25) to help filter multiple categories per sentence. Results on the SemEval, 2014 dataset were competitive (p =0.83; r = 0.81). This system was later extended to a multi-step approach which involved explicit aspects which had been detected in a CRF classifier being first classified. The aspect terms and information regarding their syntactic dependencies were assigned a label due to probability distributions determined by comparison to training corpora. A second step involved assigning a class label to the sentence to cover sentences without targets using the same criteria and a different threshold (f1 = 0.69).

One non-statistical system successfully employed on a SemEval 2014 dataset is that of Kiritchenko et al. (2014). Their system was feature-heavy and included the use of five binary one vs-all Support Vector Machine (SVM) classifiers (one for each predetermined categories in the dataset). Their feature set involved the use of n-grams, stemmed n-grams, character n-grams, word clusters (Brown et al., 1992), trained on Yelp restaurant data as well as Owoputi et al.'s publicly available clusters[5] trained on 56 million tweets (2013), and lexicon features. Lexicon features here are defined as the cumulative scores of all terms in each sentence of the review for each of the five aspect categories as annotated in the Yelp Restaurant Word–Aspect Association Lexicon[6] . Their system was the highest scoring system at time of publication (p=0.91; r = 0.86).

3.2.2 Deep Learning systems

Although the ability to train embeddings on custom corpora is a relatively simple process, it is not uncommon for systems to take pre-trained word embeddings as sole input. One example is Xue et al.'s (2017) multi-task neural approach (MTNA). Word embeddings for each word in a sentence are fed into a deep neural network involving first a Bi-LSTM (concatenating the two outputs) which is further used for two separate computations. One computation takes the Bi-LSTM output as input for a CNN. One output of this is used for an AE task in combination with the BiLSTM output. Another takes the output of the Bi-LSTM as input to a max pooling layer to be concatenated with the output of the CNN (also having undergone pooling) to be used as an aspect category detection task (see figure 3.1)

[5]Clusters available at: `http://www.cs.cmu.edu/~ark/TweetNLP/`
[6]available at: `http://www.saifmohammad.com/WebPages/lexicons.html`

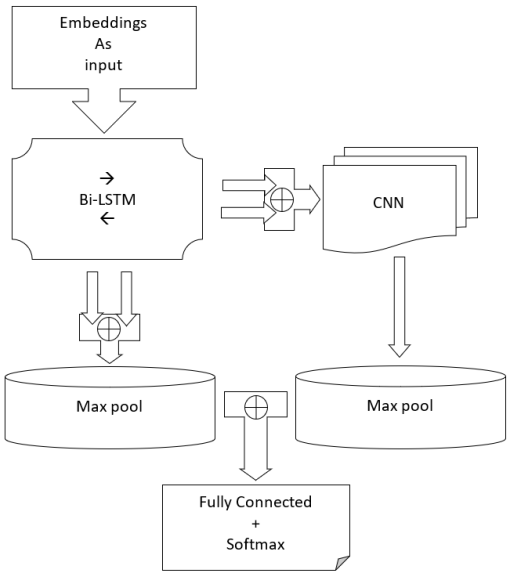

Figure 3.1: Diagram of basic structure of MTNA-s

In the original design as conceived, the dual tasks (AE and ACD) not only utilise the same inputs (embeddings) and configuration of neural network (Bi-LSTM and CNN), the final loss function of each task is a weighted sum of the loss functions for both tasks. Only f1 scores were published, but on the SemEval 2014 dataset, results appeared competitive (f1 = 89% for ACD using the shared loss function, and f1 = 0.88 ACD task alone without recourse to the AE task information.

At the time of writing, state-of-the-art performance has been achieved by a BERT-based embeddings system. Sun et al.'s(2019) system, leveraging the ability for BERT to accept sentence pairs as input, involved a novel use of an auxiliary sentence which converted AE and ACD into a sentence pair classification task. They use question answering (QA) and natural language inference (NLI) in their approach for two methods. During training for the QA method, sentences are constructed such as: What do you think of the category (e.g. price) of it?; for the NLI method, a simple pseudo-sentence is generated " aspect – category" these were used in combination with the sentence level input for each review . Three binary classifiers were added: e.g. aspect – category – positive; aspect – category – negative; aspect – category – none. In testing on the SemEval 2014 dataset, this auxiliary sentence approach outperformed all previous submissions (p=0.94; r=0.91) with the configuration employing NLI method performing best.

3.2.3 Hybrid systems

As mentioned above, Toh and Su (2016) performed ACD as well as AE in their hybrid approach. However, rather than using RNN output as features for a CRF classifier as they did with the AE task, they instead opted to use the output from a CNN (with max pooling, to a fully connected layer + softmax activation) word embeddings, a name list, head and embedding clusters were used as features. This output was then used as

an additional feature combined with features for word and bigrams, dependency heads and word clusters in a series of FFNN binary classifiers (one for each category). They reported results of p = 0.72, r = 0.74 for their classifiers which was state-of-the-art at the time of publication for the SemEval 2016 dataset.

3.3 Summary of systems

Research Question 1 asked what the linguistic-based and DL-based approaches to the tasks AE and ACD commonly entailed. This chapter has covered a wide range of systems. Below is a summary of all systems reviewed for both categories.

System	Approach	Dataset	Results		
			P	R	f1
Hu and Liu (2004)	L - Association mining + frequency + pruning	Custom dataset (electronics)	0.72	0.8	0.76
Popescu and Etzioni (2005)	L - Association mining + frequency + PMI	Hu and Liu (2004)	0.94	0.77	0.85
Raju, Pingali & Varma's (2009)	L - NP extraction + pruning	Custom dataset (electronics)	0.52[7]	0.63	0.51
Qiu et al. (2011)*	L - Sentiment Lexicon + rules (DP) + pruning	Hu and Liu (2004)	0.88	0.83	0.85
Liu et al. (2013).	L - Rules (DP) + WordNet (general word) pruning	Hu and Liu (2004)	0.88	0.87	0.87
Toh & Wang (2014)	L - Word, gazetteer, POS, head and head POS, clusters, rules (DP) + CRF	SemEval 2014	0.85	0.83	0.84
Xu et al. (2018)*	DL - Embeddings + CNN	SemEval 2016	-	-	0.74
Xu et al. (2019)	DL - BERT + SoftMax	SemEval 2016	-	-	0.78
Liu, Joty and Meng (2015)	H - POS, chunk + Embeddings + Bi-RNN	SemEval 2014	0.83	.80	0.82
Poria et al. (2016)	POS + rules + embeddings + CNN	SemEval 2014	0.88	0.86	0.87
		Hu and Liu (2004)	0.9	0.86	0.88
Toh and Su (2016)	Word, training aspects, dependency heads + embeddings + clusters + DP extracted aspects + RNN (as feature) + CRF	SemEval 2016	0.75	0.69	0.72
Ray and Chakrabarti (2019)	Rules + embeddings + CNN	SemEval 2014	0.85	0.86	0.85

Table 3.1: A summary of AE systems reviewed in this chapter: the approach, main features, classifier, dataset and published performance metrics are indicated where given. (L = system based purely on linguistic information; DL = deep learning - based system; H = hybrid system; systems used marked with *)

3.3.1 Error analysis in research papers

The systems described above are largely representative of the majority of systems used in the tasks of AE and ACD. In the course of conducting the review, what is notable is the absence of a detailed error analysis of the output of the systems. A detailed error analysis would allow researchers to be strategically select techniques or approaches to

[7]Statistics shown are for full matches – partial match precision = 0.92

System	Approach	Dataset	Results		
			P	R	f1
Carenini, Ng & Zwart (2005)	L - Association Mining + user-defined features + WordNet	Hu and Liu (2004) custom annotated to add categories	-	-	-
Zhai et al. (2010)	L - WordNet (soft constraint) + seeds + EM + Naïve Bayes	Custom dataset (household)	-	-	-
Kiritchenko et al. (2014)*	L - n-grams + character n-grams + lexicon features + SVM	SemEval 2014	0.91	0.86	0.88
Brun, Popa and Roux's (2014)	L - BoW, deep syntax + corpus frequencies + Log Reg.	SemEval 2014	0.83	0.81	0.82
Xue et al.'s (2017)*	DL - Embeddings + BiLSTM + CNN	SemEval 2014	-	-	.89
Sun, Huang & Qiu (2019)	DL - BERT + auxiliary sentence	SemEval 2014	0.94	0.91	0.92
Toh and Su (2016)	H - Word, heads + clusters + CNN (as feature) + FF	SemEval 2016	0.72	0.74	0.73

Table 3.2: A summary of ACD systems reviewed in this chapter: the approach, main features, classifier, dataset and published performance metrics are indicated where given. (L = system based purely on linguistic information; DL = deep learning - based system; H = hybrid system; systems used marked with *)

deal with the shortcomings of particular systems, providing knowledge of where systems fall short. However, many researchers seem content to compare their results to baseline results or other systems without mentioning inspection of the output at all (Hu and Liu, 2004; Popescu and Etzioni, 2007; Zhai et al., 2010; Toh and Wang, 2014,?; Liu et al., 2015; Ray and Chakrabarti, 2019). Some researchers do mention issues but stick to general coments (Xu et al., 2019; Raju et al., 2009; Poria et al., 2016) with very few citing specific issues in the output (Xu et al., 2018; Liu et al., 2013).

While there are many likely reasons for the omission of such information, such as space saving concerns; lack of generalisable errors worth reporting; the focus being on submission for the competition where errors are not immediately relevant, for example, for systems seeking to build on their predecessors or researchers looking to implement an alternative approach, this type of information (the kind of information missed by the system) is vital to be able to strategically motivate decisions in terms of addressing the kind of data missed by other approaches, and the strengths of the new approach that could compensate for it.

This lack of strategy is apparent in hybrid systems that seek to combine Linguistic and DL approaches which generally do not justify the selection of linguistic information, rules or the type(s) of DL network model selected, nor do they try to synthesise the systems in a targeted way; rather the output of the two approaches appear to be combined in a crude manner rather than with the intention of optimising for the strengths of one approach or seeking to ameliorate weaknesses in another approach.

Even ablation studies which have the opportunity to not only compare metrics, but also to compare output, disappointingly either do not report the results of inspection, or perhaps have not performed this fine-grained analysis. The degree of hedging used in the following quote by one ablation study involving a combined approach, indicates the latter.

> "As these two features are also used in the CNN system, it **may** be redundant to include them again in the multiclass classification system. In addition, the neural network features **may** have become the dominant features during training, affecting the usefulness of other features... Further investigation **may** be needed to identify better ways of combining the different machine learning systems together"

(Toh and Su, 2016).

Thus, the question remains as to whether there are significant differences between outputs of systems that adopt different approaches and whether these differences can be combined in ways which would complement weaknesses. This can only be done through a detailed analysis of the output. Thus, the following sections will describe a study which seeks to simulate the output of contrasting approaches to the tasks of AE and ACD in order to inspect results for differences, or potential for a strategic combination of approaches which would address specific issues in its counterpart.

To perform this study, systems must be selected and re-constructed to the extent that representative output can be produced. The next section will describe motivation for system selection and any modifications were made to the systems as originally described. This will be followed by a detailed account of the results of an error analysis from each system in turn.

Chapter 4

Method

The aim of this study was to compare output that is representative of contrasting approaches on the tasks of AE and ACD. The procedure involved first selecting appropriate systems from existing work. The intention was not to replicate the system results as they appeared in the original papers, since not all resources were readily available, levels of detail in descriptions of systems vary and, as mentioned in section 2.2, replications from these descriptions can often prove problematic (Marrese-Taylor and Matsuo, 2017). The intention instead is to produce output typical of that produced by the approach, such that this is available to inspection and comparison with a contrasting approach on the same task. In order that a high-quality error analysis could be performed, a minimal number of systems were required. As the intention is to compare two approaches (linguistic and deep learning approaches) for two subtasks of ABSA (AE and ACD), a total of four systems were minimally required. A number of factors were considered in system selection.

4.1 Dataset

In order that output be directly comparable, it was necessary to use a common dataset for all systems. This would facilitate the task of comparing the output making the output directly comparable. Since the most frequently described dataset in the section above for both tasks was the SemEval 2014 dataset (Pontiki et al., 2014)it was a natural first choice for inspection.

Two domains were used in the SemEval 2014 dataset, the restaurants domain, and a laptop domain. According to the authors systems performed (+10%) better on the restaurants domain due to the laptop domain involving more entities, complex concepts, and a greater degree of implicit reference (Pontiki et al., 2015). To facilitate the process of analysis, the restaurants domain was deemed most suitable. Table 4.1 shows the key statistical information for this dataset.

The structure of the dataset showed a relatively simple (xml) structure which could readily be adapted to both tasks, i.e. it could be flexibly adapted for both a sequence labelling task as well as a document labelling task with the offset of the terms denoted by a from index value to index value annotation (see Figure 4.1 for sample structure).

As indicated (in section 2.3.2): the category labels in the dataset are largely self-explanatory with the exception of the label anecdotes/miscellaneous, which ultimately became the default 'other' category. Other annotation points of note are that only explicit aspect terms are to be annotated while category labels can be implicit or explicit, and at least one category is mandatory for each review. Figure 4.1 shows an example of an explicit category; figure 4.2 shows an implicit category (portions 'of food' being implied), and 4.3 shows an example of a review with no explicit term and the default category label, anecdotes/miscellaneous, applied.

```
<sentence id="1609">
    <text>Service was quick.</text>
    <aspectTerms>
        <aspectTerm term="Service" polarity="positive" from="0" to="7"/>
    </aspectTerms>
    <aspectCategories>
        <aspectCategory category="service" polarity="positive"/>
    </aspectCategories>
</sentence>
```

Figure 4.1: Xml annotation structure in which two attributes ("from and "to") that indicate its start and end offset in the text

```
<sentence id="1579">
    <text>And really large portions.</text>
    <aspectTerms>
        <aspectTerm term="portions" polarity="positive" from="17" to="25"/>
    </aspectTerms>
    <aspectCategories>
        <aspectCategory category="food" polarity="positive"/>
    </aspectCategories>
</sentence>
```

Figure 4.2: Dataset annotation sample with aspect term and implicit category.

```
<sentence id="2707">
    <text>Go inside and you won't want to leave.</text>
    <aspectCategories>
        <aspectCategory category="anecdotes/miscellaneous" polarity="positive"/>
    </aspectCategories>
</sentence>
```

Figure 4.3: Dataset annotation sample without aspect term and default category (anecdotes/miscellaneous) applied.

4.2 Systems

4.2.1 Criteria for system selection

The intention of this study is to closely inspect the output of systems; a small number of systems allows a more thorough analysis of the output. Thus, a total of four systems

	Training dataset	**Test dataset**
Sentences	3041	800
Average sentence length	14	14
One-word aspect terms	2786	818
Multi-word aspect terms	907	316
Total aspects	3693	1134
Categories	Food, Ambience, Price, Service, Anecdotes/Miscellaneous	
Number of category labels	3392	1025
Food	33% - (1232)	41% - (418)
Anecdotes/Misc	30% - (1132)	23% - (234)
Service	16% – (597)	17% – (172)
Ambience	12% - (431)	12% - (118)
Price	9% - (321)	8% – (83)

Table 4.1: SemEval 2014 dataset key information

were to be selected. One linguistic system for both tasks and one DL system for both tasks. Factors considered for system selection included:

- Simplicity of set up

A complex design would provide too many possible variations and potential deviation from output as intended. A second issue would be the proficiency requirements for getting complex systems to operate. As such only systems which were deemed feasibly within the scope of ability to reproduce were considered.

- Access to resources (software or hardware requirements; lexicons; corpora required for training)

Systems which rely on resources which are no longer available or not publicly available were not selected by default.

- Purity of approach

In order to be able to evaluate the output of the systems separately, it is necessary for systems to use only one or other the approaches, ruling out hybrid systems.

- Level of detail in system descriptions

While this is not a replication study the intention is to closely model the approach based on existing systems. To be able to approximate these systems, detailed descriptions are required.

- Performance

It would be preferable to choose good performing systems over systems that did not perform well originally.

- Popularity

Techniques or approaches which have been used repeatedly by researchers would indicate that they are robust, and lead to results which are representative, or perhaps have theoretically interesting premises.

In short, a system would be preferable if it is simple, representative, and involves readily available resources.

4.2.2 Systems selected - Aspect Extraction

The DP linguistic approach of Qiu et al. (2011) fulfils many of the criteria above. It is often cited in the literature, suggesting that it is representative of rule-based approaches; it is detailed in terms of the rules employed and how the algorithm is applied (see appendix A); the seed lexicon is publicly available.

The DL system as described in Xu et al. (2018) is a good candidate for the DL approach. The system as described lacks complexity, and more importantly the researcher has made the resources and code available[1]. The system is also described to have performed well.

4.2.3 Systems selected - Aspect Category Detection

The linguistic approach of Kiritchenko et al. (2014) while complex, is one of the only high performing purely linguistic approaches reviewed and many of the resources used are freely available.

The DL approach of Xue et al. (2017) uses publicly available GloVe embeddings[2] as sole features for input. While system design is relatively complex and the full system design as originally intended leverages loss calculations from simultaneously performing multiple tasks (AE and ACD), the paper does publish high performing results when only performing the task of ACD.

Since descriptions of systems are not always as detailed as required for accurate reconstruction and exact resources not always available, some modifications to systems were inevitable. What follows is a description of the systems employed and areas where they (are likely to have) deviated from the original.

4.2.4 Modifications to systems

For each system, some deviations from the original design were necessary. These details are described and motivated.

AE - linguistic system

The most radical departures from system design as described were with Qiu et al.'s system. These modifications were implemented due to poor system performance on the

[1] `https://github.com/howardhsu/DE-CNN`
[2] `https://nlp.stanford.edu/projects/glove/`

System	Precision	Recall	F1
System as originally intended	0.48	0.40	0.43
Addition of rule a	0.63	0.43	0.51
Addition of rule a + b	0.64	0.59	0.61
Addition of rule a, b, + c	0.65	0.66	0.66

Table 4.2: performance of Qiu et al.'s DP system with and without the addition of other rules

SemEval 2014 dataset, which, even using a relaxed scoring system (using a binary 'I' or 'O' classification system as opposed to a stricter IOB system) only achieved an f1 of 0.43 with both precision and recall below 50%. Since recall was so low, it was decided at this stage to forgo the post-processing frequency-based pruning step.

Since the authors were clear about their technique only exploiting dependency relations with opinion adjectives the rules were expanded to be able to exploit other dependency relations. These included: One rule (a) from Ray and Chakrabarti (2019) another rule (b) from Poria et al. (2016) both exploiting dependency relations with verbs. A second rule (c) from Ray and Chakrabarti also made use of dependency relationships between nouns.

(a) If verb n has a direct object and a noun p, then label p as an aspect.

(b) If a noun h is a complement of a copular [sic] verb, then mark h as an explicit aspect.

(c) If a term is labelled as an aspect by the previous rules and there is a noun-noun compound relationship with another word, then composed of both of them and marked as an aspect. For example, "Battery Life" "Battery" or "Life" is marked as an aspect, then the whole expression is labelled as an aspect.

The addition of these rules saw gains across all metrics (see Table 4.2).

Other changes included: Substitution of MiniPar (no longer available at the link provided) for CoreNLP for pre-processing (tokenisation, POS tagging and dependency parsing).

Qiu et al. performed a post-processing expansion of their single-word aspects to aspect phrases by looking for nouns and adjectives within a given proximity to the target word. Experiments were conducted with extraction of both consecutive nouns and combinations of adjectives and nouns (using the NLTK regex chunk parser[3]) however, it was discovered that the inclusion of adjectives returned significantly lower precision numbers which lowered overall performance and only consecutive nouns were finally used to expand the terms.

Aspect (term) Extraction - Deep Learning system

While the code and resources for Xu et al's system (2017) was made available, it was designed to be used on the SemEval 2016 dataset. Due to the wide-ranging changes

[3] `https://www.nltk.org/_modules/nltk/chunk/regexp.html`

implemented after SemEval 2014 for definitions of aspect categories, the xml structure of the dataset was significantly different to that of the SemEval 2014 dataset.

Although extracting information from a different xml structure did not pose a problem, making all the necessary modifications in the original author's code proved complex. Several attempts were made, but the process proved sufficiently challenging that after several attempts, it was determined that it would be simpler to use the resources provided and rebuild the CNN from scratch. The original code was consulted for information related to various parameters such as dropout value, activation functions, optimizer, loss functions, batch sizes and number of epochs. However, the resulting code was significantly different to the published author's original, and is better considered as a radical simplification. Accuracy statistics during training were very high (0.9959) indicating that the modifications were not detrimental.

Aspect Category Detection - Linguistic system

The list of features published for the ACD task in Kiritchenko et al.'s system (described at the end of section 3.2.1) was long (n-grams, stemmed n-grams, character n-grams, word clusters both custom and pre-trained and lexicon features). While most of these features could be implemented, an attempt to build clusters based on the Yelp dataset using Percy Liang's implementation of the Brown clustering algorithm[4] proved too computationally slow to complete (approximately 7 days of processing time was required for the dataset consisting of 80,492 reviews). Other implementations of Brown's clustering algorithm are available, although these were not explored. As such, only the publicly available tweet clusters were employed.

Kiritchenko et al.'s system used five classifiers, one per aspect category. However, if their classifiers did not return a label for a text, they would implement a post-processing step. In this step, any texts which had not been assigned a label were assigned (a) label(s) if a certain threshold probability (0.4) was attained for a class given a document (review). It was not clear however what metrics were used to calculate this posterior probability. For this reason, this post-processing step was omitted. Since Kiritchenko et al. published results for various combinations of features, including the omission of the features described above, a direct comparison could still be made.

ACD - DL system

Xue et al.'s description of their system was highly detailed in some areas while lacking in other areas. The system as described consisted of 200-eimensional GloVe embeddings, a bi-LSTM layer, and CNN layer(s). Details of the CNN were a little confusing since reference was made to a one-dimensional CNN layer with 'a set of kernels with different widths' (widths of 3, 4 and 5). It was not clear how the outputs of these layers were handled. Some experimentation using single CNN layers of various widths compared using the output of several CNN layers of various widths which were concatenated before being max pooled showed that a single layer with width 5 outperformed all other attempts. Since this was the simplest configuration as well as being the highest

[4]Percy Liang's implementation of the Brown clustering algorithm available at:`https://github.com/percyliang/brown-cluster`

performing, the final configuration only used one CNN layer.

Details on the bi-LSTM setup were hard to discern, since, according to the author hyper parameters (batch size, dropout rate, dimensions of the LSTM cells and weight in loss function) were optimised individually for each of the five binary classifiers. This was likely done to optimise results, and since peak performance was not the intention of the current research, some default hyper-parameter values were selected for the NN model (dropout 0.2; activation – tanh, units – 300) for all five classifiers. These were trained for 20 epochs.

Chapter 5

Results

A summary of results of the performance of the systems described above compared with the original system published results can be seen in table 5.1

System	Original published results (with dataset)	Results of reconstructed systems (with dataset)
AE – Linguistic (Qiu et al., 2011)	p=0.88 r=0.83 f1=0.85 Hu and Liu	p=0.66 r=0.64 f1=0.65 SemEval 2014
AE – Deep Learning (Xu et al., 2018)	- - f1=0.74 SemEval 2016	p=0.92 r=0.83 f1=0.87 SemEval 2014
ACD – Linguistic (Kiritchenko et al., 2014[1])	p=0.88 r=0.80 f1=0.84 SemEval 2014	p=0.86 r=0.83 f1=0.84 SemEval 2014
ACD – Deep Learning (Xue et al., 2017)	- - f1= 0.89 SemEval 2014	p=0.89 r=0.84 f1=0.86 SemEval 2014

Table 5.1: comparison of performance of re-implementation of selected AE and ACD systems with original systems as published with datasets

At this stage the answer to Research Question 2 can be addressed, which sought to discover how the performance of the approaches compared with one another on each task. From the results shown in table 5.1, it is clear that the DL approach outperformed its linguistic counterpart on every metric in both tasks.

In the following section, the output of the systems, for each task is inspected for differences. Errors that occurred within a significant subset of the output are analysed and differences described and categorised.

5.1 Error analysis

The purpose of the error analysis is to investigate the extent to which the output of the two approaches qualitatively differs. For this, the results on the testset was used (consisting of 800 reviews).

[1]Statistics quoted were for results that did not use post processing or clusters

5.1.1 Aspect Extraction Task

For the AE task, the worst performing system (Qiu et al's DP system) committed a significant number of errors in comparison to the competing DL system. When selecting reviews for error analysis, it was not sufficient to extract a given number of errors (e.g. 50 errors) for each system as this would not be representative of the disproportionate imbalance of errors. It was instead decided that a sizable portion of all the reviews should be pulled to ensure good coverage of errors and proportion of errors for all systems. The process involved some trial and error, initially by extracting a certain percentage of reviews and looking at general error types (introduced below), evaluating frequency of errors compared with the statistics for overall performance and continuing until the balance of errors reflected this ratio. A more detailed analysis was subsequently performed to determine whether any new types of errors would emerge by incrementally expanding the selection. This continued until no new subcategories of errors emerged.

For the AE task, 120 reviews (constituting 15% of the dataset) were inspected for errors (see appendix B).

The first level of analysis grouped errors into two major classes:

- **Type A:** words (tokens) which should have been identified as aspects according to the gold labels, but were not. These are errors that affect recall.
- **Type B:** words (tokens) which should not have been identified as aspects according to the gold data, but they were mislabelled. These are errors affecting precision.

Linguistic approach

Type A: For the linguistic approach, there were 42 type A errors. Many of these can be group into the following, more specific sub-categories:

Type A: Boundary error

The aspect term is a phrase and part of the phrase has been correctly identified as an aspect by the system, but not the entire phrase (a partial match).

In sentence 7 below, the system identified 'tuna roll' as an aspect term expression; however the word, 'spicy', wasn't identified as part of the aspect term resulting in a partial match. Since spicy is (assumed) to be part of the name of the dish, according to annotation guidelines, it is part of the full aspect term - this is in contrast to the subjective term, BEST, which is not to be annotated as part of the aspect term.

sentence 7: BEST spicy tuna roll.

14 of these type A errors can be classified as are boundary issues.

Type A: Parsing error

The error derives from a problem with the parser (POS tagger or dependency-relation parser). An error at this level will mean that rules which rely on these features will be error prone.

In sentence 0 below, the system has not identified the word 'bread' as an aspect term. This is due to the fact that the POS tagger labelled the phrase 'top-notch' as a noun phrase (ADJ + N) instead of an adjectival phrase. The word bread, thus, is not parsed as being in a predicative relation with the adjective complement, and thus not identified as an aspect.

sentence 0: The bread is top notch as well.

5 issues result from incorrect parsing

Type A: Resource issue

This category of error results as a weakness of a lexicon approach. Because a lexicon is used in order to identify features and opinion words, any opinion words that do not feature in the lexicon or are not a recurring feature of the text will cause aspects to be missed. Words that are incorrectly spelled may be omitted, or words that have a specific connotation within the domain may be misinterpreted as non-opinion words, and were not included in the seed dictionary.

In sentence 20 below, the system has not identified the word 'convienent' as an aspect term. This is attributable to the misspelling of the word convenient not being included in the opinion seed lexicon. Thus the word 'parking' was ignored.

sentence 20: and the convienent parking at Chelsea Piers.

2 issues are attributable to use of lexicon.

Type A: Coding issue

This error results from the target aspect being unidentifiable due to not being coded for.

In the example below (sentence 34), the aspect terms are 'appetizers' and 'entrees'. However the context in which it occurs is not part of the rule set employed to extract aspects. While the word, 'food' was identified, the parenthetical aspects introduced with the structure, 'from ... to ..' was a rule that was not included in the algorithm and could not have been returned.

sentence 34: the food (from appetizers to entrees) was delectable

16 issues result from coding.

Type A: Annotation-related error

An admittedly controversial inclusion to error subcategories are those that may be attributable to inconsistent annotation or subtleties of the annotation guidelines which are not obvious. This type of error is subjective, and an inevitable effect of the annotation process in which annotators can deliberate and refine the annotation process amongst themselves. However, as indicated in the example below, it is hard to reconcile the decision by the annotators to label the word, 'menu' as an aspect term, since 'rose roll' is the subjective target of the review fragment, and the reference to the menu appears to be an informative inclusion suggesting that the customer would have to actively request the food item, and thus not a target of the subjective expression.

sentence 8: Try the rose roll (not on menu)

3 issues are (potentially) attributable to annotation.

Noun Phrases of the type ADJ N in which adjectives like 'spicy', 'fresh', or 'Asian' were annotated as part of the aspect. Since these were not coded for, they form both boundary errors and coding issues. They have been accounted for as boundary issues only in the statistics provided.

In sum, type A errors constituted 44% of all errors, the bulk of which related to boundary and coding issues (see table 5.2).

Type B errors

There were 52 type B (precision) errors in the inspected output of the linguistic AE system: Errors here have been categorised as:

Type B: References to the entity

In the definition of an aspect as given in the 2014 SemEval task, references to the entity (restaurant, place) are not permitted as aspect terms (discussed in section 2.3.2. However, Qiu and Liu's system was not designed with this distinction in mind and as such indiscriminately returned nouns and noun phrases depending on their relationships with other words rather than their semantic content.

In the example given below, the word, 'restaurant' has been extracted as an aspect term due to its dependency relation with the adjective, 'great'. Technically the word restaurant is the entity, whereas an aspect term can only refer to an attribute/component of an entity according to the annotation guidelines.

sentence 45: this is a great new restaurant

A significant number of the errors inspected are of this type.

Type B: References to time

Many references to time were extracted as aspect terms. This could also be reclassified as a coding error, since in the original DP system, these were dealt with in a post-processing pruning step which was not performed in this implementation. However, references to time were so frequent, they can be thought of as a distinct (sub)class of their own.

In the example below, the word, 'evening' has been classified by the system as an aspect term due to its relation with the adjective, 'unforgettable'. However, this is of course not an aspect of the restaurant entity.

sentence 12: the top - notch food and live entertainment sold us on a unforgettable evening

In total there were 8 instances of this type.

Type B: Parsing errors

As above

In sentence 0 below, the system has identified the word 'Esp' as an aspect term. This is due to the fact that the POS tagger labelled the word as a noun rather than an adverbial, thus creating a series of three nouns which were identified by the algorithm as one noun phrase.

sentence 9: Esp lychee martini.

There were ten of this type of error.

Type B: Annotation-related issues

Decisions made in the annotation process to include certain terms such as verbals were explicit in the annotation guidelines, while others seem to have been part of some implicit criteria. Making it such that new rules could not be intuitively inferred and can only be identified post-hoc.

In the three examples below, each time, the system returned the word, 'experience', as a target aspect, but these are not annotated as such in the dataset. This is suggestive of a systematic approach by the annotators to have predetermined that an experience cannot be considered an attribute of a restaurant. This is not obvious from the guidelines.

sentence 35: You will be very happy with the experience
sentence 47: Had a great experience at Trio
sentence 60: The whole experience was satisfying

Other examples include the word, 'choice' and 'reviews'.

As can be seen in table 5.2 type B errors formed the majority of errors (56%), of which a quarter of are attributable to references to the entity (25%), and almost a fifth of errors are attributable to references to time (21%).

Table 5.2 shows a summary of errors for the AE linguistic approach. Common to both type A and type B errors are issues which result from parsing. These result in cumulatively 16% of errors. The largest overall cause of errors derive from coding. This can be almost wholly attributed to the fact that the DP system pre-dates the SemEval 2014 task and was not designed with the annotations in mind. The references to time, the restaurant entity and type B annotation-related errors might also be included in this category, since they also derive from the task definition and annotation procedures.

Type A errors (recall)		**Type B errors (precision)**	
Coding issues	16 (17%)	Reference to entity	13 (14%)
Boundary issues	14 (15%)	Parsing errors	10 (11%)
Parsing issues	5 (5%)	Reference to time	8 (8%)
Annotation issues	3 (3%)	Annotation errors	6 (6%)
Lexicon issues	2 (2%)		
Miscellaneous	2 (2%)	Miscellaneous	16 (17%)
Total errors	42 (44%)	**Total errors**	53 (56%)

Table 5.2: Error breakdown for linguistic Aspect Extraction system. Percentages of total errors (type A and type B errors combined) given in parentheses

Deep Learning approach

In general, the total number of errors was far below the linguistic-based Double Propogation system and many were not possible to further categorise. Table 5.3 shows general statistics for the DL system.

Type A errors

The majority of the DL errors were type A of which only 8 could be further grouped.

These were boundary errors (in which part of the aspect term was labelled but not the whole aspect resulting in a partial match).

Four of these were issues with adpositions as part of noun phrases such as sentence 18 in the example below in which 'of the' was omitted from the aspect term. Others included sentences 22 and 29. Two of these involved punctuation, such as sentence 47 below in which the nouns were labelled as aspect terms but the forward slashes were not.

sentence 18: The portions of the food that came out were mediocre.
sentence 47: I would highly recommend the portobello/gorgonzola/sausage appetizer

The remaining errors were more difficult to categorise: Although 3 involved the word 'taste' (sentence 85 is given as an example below); the rest were not subject to further categorisation.

sentence 85: I was pleasantly surprised at the taste

Type B errors

In general, the deep learning approach committed proportionally fewer Type B errors (see table 5.3).

3 of these involved verbals two of which were from sentence 42 given below (tasting, seasoned)

sentence 42: If you want good tasting well seasoned latin food

2 of these involve errors which correspond to annotation-related issues identified in the linguistic system above (sentence 27: 'choice', sentence 60: 'experience'), seemingly predetermined as non-aspects in the annotation process.Both examples are given below.

sentence 27: We were pleasantly surprised with our choice
sentence 60: The whole experience was satisfying

The remaining errors were miscellaneous.

Table 5.3 shows a summary of errors from the DL approach. In general, type A errors were the most prevalent, most of which stemmed from boundary issues (partial matches). Nearly half of all errors of both types were idiosyncratic errors which were difficult to categorise (miscellaneous: 48%).

type A errors (recall)		**Type B errors (precision)**	
Boundary issues	8 (32%)	Verbals	3 (12%)
Miscellaneous	8 (32%)	Annotation errors	2 (8%)
		Miscellaneous	4 (16%)
Total errors	16 (64%)	**Total errors**	9 (36%)

Table 5.3: Summary of Aspect Extraction deep learning system errors. Percentages of total errors given in parentheses.

Summary of Aspect Extraction errors for both systems

Table 5.4 shows a summary of the error analysis for both systems. From table 5.4, we can see that, in terms of ratio of types of errors, on the whole, the DL approach committed proportionally more type A (recall) errors than the linguistic approach, which was more balanced in terms of types of errors committed (although there were more Type B errors than type A).

	DL	**Linguistic**
Type A (missed targets)	16 (64%)	42 (44%)
Type B (misidentified as targets)	9 (36%)	53 (56%)
Total errors	25	95

Table 5.4: Summary of Error Analysis for both systems on the AE task. Percentages in parentheses are within the respective systems

In total, out of the 120 reviews under analysis, there were 76 reviews on which the two systems combined produced errors, of which 16 reviews were common to both systems . Of those 16 reviews, there was only one review which both systems produced different error types. Nine reviews containing errors were unique to the deep learning system of Xu et al., while 79 reviews were unique to linguistic-based system of Qiu et al.

A similar analysis is performed below for errors on the ACD task.

5.1.2 Aspect Category Detection Task

Both systems were consistently high in terms of both precision and recall, and as such errors were less frequent than for the AE task. For this reason, more output could be inspected. A similar process of extracting reviews until it could be satisfactorily determined that there were no new error types and the balance of errors reflected the system performance statistics was followed. In total, 200 reviews were inspected for errors, constituting 25% of test set (see appendix C for full error output).

As a document-level labelling task clearly identifiable causes of errors hard to determine. There may be one category per document assigned or, as is often the case, there may be multiple, and causes must be inferred. As such, it is not possible to provide a detailed breakdown of errors and sub-categories of errors as it was for the sequence labeling, AE task. For this reason, analysis cannot systematically proceed beyond the general distribution of type A and type B.

A general summary of errors is provided in table 5.5, from which it appears that these approaches are almost indistinguishable in terms of performance. Both systems exhibit approximately the same number of errors, and both had a similar distribution of error types in that type A errors are more prevalent by a significant margin across both systems.

	Linguistic	**DL**
Type A (missed targets)	30	26
Type B (misidentified as targets)	11	12
Total errors	41	38

Table 5.5: Summary of Errors for the ACD task for both systems

Table 5.6 shows a more detailed breakdown of errors associated with each label. For both systems, the most common type A errors relate to the dominant class, 'food', followed by unlabelled reviews, with 'price' contributing the fewest errors in both systems. The most common type B errors related to the default, anecdotes/miscellaneous, class label being most commonly misapplied by a considerable margin.

		Linguistic	**DL**
Type A	Unlabelled	9	5
	Ambience	4	4
	Anec/Misc	3	3
	Food	10	7
	Price	2	2
	Service	2	4
Type B	Ambience	2	0
	Anec/Misc	7	8
	Food	1	3
	Price	1	0
	Service	0	1
	Total	41	38

Table 5.6: Breakdown of Errors for the ACD linguistic system

Although difficult to discern from the raw statistics due to the common occurrence of reviews having multiple labels applied, of the 200 reviews inspected, the combined number of reviews containing errors totalled 45. There were only 17 reviews on which both systems produced erroneous output, eleven of which for which the systems produced the exact same errors. For these eleven reviews, where both systems returned identical errors, there was no significant difference in proportion of types of errors (5 type A errors, 6 type B errors). For all the type B errors in this commonly shared set, the same label (anecdotes/ miscellaneous) was applied in each case. These accounted for almost all of the type B errors associated with this label for both classifiers. There was no such pattern evident for type A errors.

The implications of the above error analysis will be discussed in the following section.

Chapter 6

Discussion

The aim of the thesis is to evaluate system output from two approaches (an approach that used purely linguistic information and one involving supervised, deep learning neural networks) to two ABSA subtasks: aspect extraction (AE) and aspect category detection (ACD). To this end, a small study was performed which emulated output from these approaches. This chapter seeks to answer the remaining research questions in light of the results of the study introduced in the previous chapter.

First, some limitations of the study will be mentioned before the remaining research questions are addressed. Strengths and weaknesses of the competing approaches are discussed before conclusions are drawn and recommendations offered.

6.1 Limitations

In order that the analysis can be considered valid, it is necessary that the output of the systems is representative of the respective approaches. However, because access to output from the original systems on which this research was based was not available, the only way that this could be determined was by comparison of performance metrics; the assumption being that comparable performance metrics would suggest that the system output is representative of the approach.

While for the two ACD systems, the results in table 5.1 appear to align well, there is still some room for doubt when it comes to papers which only published an f1 score, as the underlying distribution of precision and recall metrics which would provide more certainty are impossible to deduce. We must assume from the available information that alignment suggests the output is representative.

The results for the two AE systems do not align with the published results. However, this could be attributable to the use of different datasets. Given the idiosyncrasies of annotated datasets in general, systems which were not designed with a specific dataset in mind could not be expected to perform perfectly. Xu et al.'s double embedding system was originally implemented on the SemEval 2016 dataset, and system performance was generally lower for all submissions than those of SemEval 2014 from competing entries; the top system reported for SemEval 2016 scoring no higher than 0.73 (f1), whereas the top system two years prior scored more than 10 points higher (f1 = 84). For this reason we can consider that, since the reconstructed CNN system (marginally)

outperformed the best on the 2016 dataset, and that the system as implemented here also outperformed the best system on the 2014 dataset, the system performs as intended and the output is representative of the approach.

The linguistic AE system based on Qiu et al.'s system design deviated significantly from the system as described in the original publication. This calls into question how representative the output could be. However, it can be said that as all of the original rules and algorithm remained intact, all additional rules only exploit dependency relations and POS labels in the same manner as the DP system, and the same seed opinion lexicon was employed, this should be considered to be similar enough that the output is reasonably representative of the approach.

A second limitation was that, although system selection was motivated, the number of systems was limited to one system per approach per sub-task. The weaknesses or strengths represented by one particular system are not necessarily representative of all systems within that approach.

Finally, there is undoubtedly a gap in coding proficiency between the authors of the original systems compared to the implementation in this thesis. It is not beyond the realms of possibility that errors in the output might relate to implementation of the code as opposed to inherent deficiencies in the approach.

With these limitations in mind, the remaining research questions can now be addressed.

6.2 Qualitative differences

Research Question 3 asked the extent to which the output of the two approaches qualitatively differs.

6.2.1 Aspect Extraction task

Overall, it was clear that the complexity of the linguistic approach introduced more points of failure in the system, as there were more, and more varied issues. A considerable number of resources were required to run the system as intended, including a word tokeniser, a POS-tagger; a dependency parser; a hand-crafted seed opinion lexicon, and the rules were numerous and the algorithm complex. The reliance on pre-processing and a lexicon which was not tailored to the specific domain added several potential points of failure to the process, and a not insignificant number of the errors in the inspected data could be attributed to parsing errors (16%). This is not a problem that the DL system suffered from. That said, although errors were far more prevalent in the linguistic AE system, the fact that they were relatively easy to identify and categorise makes them easier to target when refining the rules/algorithm.

Causes of most of the DL system errors were more difficult to determine. In some instances, individual words would be labelled as aspects in one instance, but not in others for example the word 'experience'. There were also curious instances where 'obvious'

aspects were not identified (sentence 1: 'times' in the aspect term, 'delivery times'; sentence 100: 'taste' - 'in the worst was the taste'). This is suggestive of Cambria et al.'s observation that decision making in machine learning is "merely probabilistic" (2018); these words commonly occurred in the training set, sometimes as aspects, but more commonly not. Given the degree of variance in the output of the system on each of the five runs, it is likely that changes in labeling for these words might account for some degree of this variation.

Boundary issues (partial matches) plagued both systems accounting for 15% of all errors inspected in the DP system and 32% of the errors inspected in the DL system output. When looking at the individual metrics for 'B' (beginning of the aspect term) compared to 'I' (inside the aspect term) for multi-token aspect phrases, there is a considerable drop in recall for the 'I' label which affected the overall performance of the system (appendix D). While it is clear that determination of boundaries of aspect terms was tricky, even for human annotators (Poria et al., 2016), in 10 of the 16 reviews on which both systems produced erroneous output, five were boundary issues involving prepositions or punctuation inside the NP which were labelled as 'O'.

The proportion of type B (mis-classification) errors demonstrates the indiscriminate nature of the rule-based approach. The linguistic system committed more than 5 times the number of misidentified targets as the deep learning-based system. However, the most common type B errors were relatively easy to account for since reference to the entity[1] was not taken into account for when coding the rules. This was not the case with the DL system as there were no clear pattern in the errors. The issue with the mis-identification of verbals is likely due to these being annotated as aspects in the dataset and learned in training.

6.2.2 Aspect Category Detection Task

The ACD task displayed a different pattern of errors. In contrast to the sequence labelling AE task, results for both systems in the document labelling ACD task seem identical, for all intents and purposes. Numbers of errors, types of errors, distribution of errors over labels are difficult to distinguish in both systems. However, since the two systems only shared around a quarter of errors (overlapping errors from the two systems), we might infer that there is something different occurring with the two approaches, although it is difficult to discern what that might be.

There are several factors which make diagnosis of errors particularly difficult. With a document-level labelling task of between one to four labels per document, tracing errors to a particular cause is not trivial since, according to the guidelines, aspect categories can be implicit, which is often the case when applying the default 'anecdotes/miscellaneous' label, and there is no consistent one-to-one relation with individual tokens. As in the case for the AE DL system, the DL approach of Xue et al. was prone to a large degree of variation each time the system was run making errors more resistant to diagnosis. Even Kiritchenko et al.'s linguistic system, which was stable as far as output is concerned was resistant to error analysis due to the complexity of the

[1] the restaurant cannot be an aspect by definition of an aspect being an attribute of an entity (Pontiki et al., 2014)

features used. The high performance of the system could be attributable to a lack of reliance on pre-processing (as opposed to reliance on the efficacy of a separate POS tagging/ dependency parsing system of the DP AE system) but it is more likely due to the use of the lexicon features which scored selected words for each category. The way this lexicon had been constructed is not completely transparent which adds another layer of abstraction to the input, further confounding analysis.

We do see in the statistics for type B errors however, that both systems show bias towards the default label, 'anecdotes/miscellaneous', even though it is not the majority class. This suggests that there is something unusual about the systems' interpretations of this label, perhaps due to a degree of arbitrary decision making on the part of the annotators who reportedly had difficulty categorising the default label (Poria et al., 2014). This may mean that there are unintended patterns in the data regarding this label which is more suggestive of an annotation problem.

The reviews which did not receive a classification did not overlap for the two systems, highlighting the differences in mechanisms between the two approaches, but not indicative of a clue to how errors might be categorised.

6.3 Opportunities for combining approaches

Research Question 4 asks whether there are clues in the output on areas the different approaches might be engineered to complement one another, and indeed there might be.

There are two steps which are near ubiquitous in descriptions of linguistic systems which do not feature prominently in DL systems: pre- and post-processing. One clear candidate for the application of a post-processing rule insertion are the boundary issues in the case of the DL AE system, which were prevalent enough that they could be clustered. A post processing rule[2] to target regular issues would likely improve precision.

Another post-processing possibility is also available to the DL ACD system of Xue et al. Despite difficulty in diagnosing errors on the ACD task, a step with the potential to improve recall performance is to target the unlabelled reviews. In Kiritchenko's original system, a post-processing step was added in which they identified any unlabelled reviews and relabelled them according to a probability calculation. This was omitted in this study since it was not clear how that was performed from the description. However, in the case of the DL system, the default output is a probability score. It would be technically a simple matter to isolate the label probability scores for these reviews and implement a lower threshold for classification for these obviously erroneous classifications.

When it comes to pre-processing, although some of this does take place in DL systems, tokenisation, or pre-training of embeddings, for example, linguistic systems make much broader use of pre-processing (such as stemming or lemmatisation, parsing information regarding dependencies, parts of speech, constituency chunk information,

[2]such as aspects identified in proximity to one another separated by punctuation or a preposition/ preposition + determiner should be expanded

semantic role labels and named entity recognition among others). While we have observed that over-reliance on pre-processing systems can cause problems downstream, in the case of DL systems, the addition of some of this information may add specific additional syntactic information useful to the task.

Do et al. (2019) observed that, in particular, information from POS tagging and word chunks contribute from between 1% to 4% gains on embedding-based systems. In the case of a sequence labelling task, like AE, there is certainly scope for the addition of linguistic information at the token level. It is surprising then that Xu et al.'s DL system did not attempt to include any of this information, and that most or all of the performance is attributable to the information encoded in embeddings. While minimal inputs lighten the load in terms of minimising feature engineering, inclusion of POS information as a pre-processing step might not only lead to performance gains, but may also go some way to tackling a persistent issue in DL systems.

One major element that confounds a targeted post-processing approach in DL systems is the degree of variance in the output. Methods that could address this, even if the output were to be more error prone, might make error analysis more viable as a strategy to identify areas for targeted post-processing (pruning/ expansion).

The implementations of hybrid systems by Poria et al. (2016) and Ray and Chakrabarti (2019) do little to either strategically target errors or deal with system variation; however Toh and Su's (2016) approach showed a novel approach using system output of the unsupervised classifier as a feature for input to a deterministic system which was offset by a range of linguistic features. Although they did not mention the stabilising effect of using CNN output into a more classical system, we can speculate that it may have had a dampening effect on the variation making error analysis a more productive means of improving system performance. This might also be a way of utilising the complexity of the feature rich system of Kiritchenko et al..Combining their features with the probability distribution output of the DL system as an additional feature for a more deterministic system. While this last direction is speculative, it does leave the door open for creative ways of thinking about dealing with the issue of non-deterministic output.

As a proof of concept of the potential for proposed linguistic modifications, the two DL systems were re-run: one with a pre-processing step and one with a post-processing step:

6.3.1 Aspect Extraction – deep learning with pre-processing: addition of POS

Xu et al.'s system was re-run with the simple inclusion of POS tag information. While an average score over five runs did indeed show a 1% increase in f1 (P = 0.92, r = 0.83, f1 = 0.88), this was within margin of system fluctuation previously seen. However, the positive result for such a minor change does lend some credence to the idea that inclusion of more of this type of information would see incremental improvements.

6.3.2 Aspect Category Detection – deep learning with post-processing: addition of rule for null values

Applying a minimum threshold of 0.4 probability (as was the case with Kiritchenko et al.) of the 5 unclassified reviews, 4 of the previously 5 unlabelled reviews in the inspected output received a label: 3 were labelled correctly according to the gold data and one incorrectly (the incorrectly applied label once again being the default 'anecdotes/miscellaneous' label). In the inspected sample, this would decrease type A errors by 4 (16%) and increase type B errors by 1 (8%). Since there were only 26 unlabelled reviews in total in the output for the ACD DL system, following this post-processing method, of the 26 total originally unlabelled reviews, 18 labels were applied: 12 correctly, 6 incorrectly.

In both cases, although not contributing significantly to system performance, the proposed modifications do represent an incremental increase in performance for both systems.

6.4 Reflection on performance

The strengths of the DL systems, their simplicity and high performance, is naturally attractive to researchers, especially in generating interest in their research. In this regard, it is easy to see why embeddings have been embraced by the NLP community, as their availability and power is undeniable. High performance coupled with reduction in costly feature engineering due to an inherent ability to learn idiosyncratic patterns and compensate for corrupt input like spelling mistakes or language innovations, creates a situation when poorer performing, more rigid linguistic approaches have fallen out of favour. However, in forming conclusions about the approaches and their effectiveness, it may be worthwhile to dwell a little on the price of performance. It is clear that the DP system is the worst performing of all systems, but it stands apart from all other systems in ways which make it difficult to dismiss.

Firstly, whereas the DL systems seem to have reduced expensive feature engineering since they rely on freely available embeddings as inputs, ultimately, these are supervised systems and are only as good as the data they are trained on. The annotation process is time consuming and can lead to arbitrary, or non-intuitive, decisions. In the case of the SemEval 2014 dataset, which was adapted from an existing dataset (perhaps with the intention of saving time or effort) we are confronted with a host of such issues:

- The reviews are fragments rather than complete reviews
- Arbitrary decisions were made about:
  - Annotating boundaries and spans of aspects
  - Inclusion of verbals as aspects
  - Exclusion of implicit aspects
  - The use of only 5 categories and their delineation
  - Every review requiring a category (and that these categories fit the predetermined categories above)

These decisions made in the annotation process bring into question how representative of real-world data this dataset can be said to be. Indeed, for the subsequent ABSA

SemEval tasks, many of these decisions were modified or done away with. These issues are not obvious with DL systems who learn the idiosyncrasies with apparent ease, but perhaps that makes them better at masking problems than revealing any truth with lasting value.

Secondly, to implement any of the DL systems on a new domain or dataset acquisition of a new, comparable dataset is required which must be annotated before retraining and retesting can occur. This is costly, time-consuming, and inevitably results in arbitrary annotation decisions.

Thirdly, the variation in output upon each run means that the output cannot really be trusted. It means that the user has to employ probabilistic strategies in how they treat the data, assigning a degree of certainty to the overall output which cannot likewise be assigned to individual decisions.

Like much of the research reviewed, perhaps too much of the focus of this thesis has been centred around performance on a particular dataset. It might be appropriate to take a step back and reflect on the task as it was originally conceived.

When we widen our focus from performance on the structured data in a particular dataset in a given domain and remember that the task of AE and ACD is to take unstructured data from any domain and make its contents available, the strengths of DL systems begin to wane. Motivation to perform well on a given evaluation dataset renders the system niche, almost unusable for other datasets. Even the linguistic ACD system of Kirichenko et al. suffers from being too heavily tuned to the particular dataset by virtue of the lexical features, that it is not immediately employable on other datasets. In this light, high performance metrics begin to lose their allure.

In comparison to these deficiencies, the linguistic aspect extraction system of Qiu et al. requires no training; it can be immediately employed on any dataset in any domain; it is completely transparent in operation, where every classification 'decision' can be traced back to some cause, and as a result, errors can be targeted accurately with the introduction of new rules. For example to address the issue that many of since references to the entity itself (restaurant/ place), or time (evening / years) accounted for around 40% of these errors (and over 22% of overall errors), a simple rule insertion to ignore these words alone would see the proportion of overall errors drop, and the ratio of type A to type B errors skew toward type A errors, resembling the proportion of DL errors.

This amenability to analysis and manipulation, though, is perhaps not as valuable as the knowledge the linguistic community gleans when implementing a system on real world data that is rooted in a theory about how vocabulary is organised, the value of linguistic properties and the boundaries of how meaning can be encoded in language. From this perspective, poor system performance is a signal that a concept is imperfect and absorbing these lessons allows rapid evolution of ideas as solid theories endure while poor ones fall by the wayside.

6.5 Conclusions and Recommendations

This thesis set out to discover how results of methods that employ linguistic information differ from those of deep learning techniques, and we have seen that results differ in a variety of ways, depending on the type of task. It has been demonstrated that in the narrow confines of artificially created datasets that the DL approach excels with apparent ease, although puzzlingly, we learn very little from the process. We have also seen that the more transparent the process, the more amenable the output is to analysis, and linguistic systems that pile on features can produce output that is as opaque as its DL counterpart, or at least requires a more exhaustive analysis. It has further been demonstrated that error inspection can lead to a targeted approach to synthesise techniques from linguistic and DL approaches to enhance performance on both AE and ACD.

However, throughout the process of system implementation, analysis and reflection, it has become apparent that the apparent ease with which DL systems operate masks the fact that while metrics are a good indicator of system performance, they are a poor substitute for real-world applicability. Having errors in output is expected, but more importantly, having errors borne of ideas from linguistic theory provides insight into the state of our own knowledge. This is far more valuable than errors that defy analysis or even identification. In this regard, we are reminded of Chomsky's assertion (1) that understanding derives from theory, and that perhaps a more laudable aim for system development would be to keep in mind the value of creating systems which challenge and further linguistic insight, allowing researchers to roll up their sleeves up to get to the root of a problem.

The main recommendations that flow from this paper are twofold. There is certainly room for exploring innovative ways to combine linguistic and deep learning-based systems to synthesise strengths, and inspiration can be found in research like that of Toh and Su for areas to experiment; but perhaps the main recommendation is that researchers shift focus from optimisation for performance on niche datasets to optimisation for analysis on real world data, which may evolve our understanding. To that end, even the worst performing system based on a linguistic theory has the potential to offer value, albeit one that cannot be measured in terms of precision and recall.

Appendix A

Appendix A - Qiu et al.'s Linguistic Rules and DP Algorithm

RuleID	Observations	output	Examples
$R1_1$	$O \rightarrow$ *O-Dep* $\rightarrow T$ *s.t.* $O \in \{O\}$, *O-Dep* $\in \{MR\}$, $POS(T) \in \{NN\}$	$t = T$	The phone has a good "screen". (*good* $\rightarrow$ *mod* $\rightarrow$ *screen*)
$R1_2$	$O \rightarrow$ *O-Dep* $\rightarrow H \leftarrow$ *T-Dep* $\leftarrow T$ *s.t.* $O \in \{O\}$, *O/T-Dep* $\in \{MR\}$, $POS(T) \in \{NN\}$	$t = T$	"iPod" is the best mp3 player. (*best* $\rightarrow$ *mod* $\rightarrow$ *player* $\leftarrow$ *subj* $\leftarrow$ *iPod*)
$R2_1$	$O \rightarrow$ *O-Dep* $\rightarrow T$ *s.t.* $T \in \{T\}$, *O-Dep* $\in \{MR\}$, $POS(O) \in \{JJ\}$	$o = O$	same as $R1_1$ with screen as the known word and good as the extracted word
$R2_2$	$O \rightarrow$ *O-Dep* $\rightarrow H \leftarrow$ *T-Dep* $\leftarrow T$ *s.t.* $T \in \{T\}$, *O/T-Dep* $\in \{MR\}$, $POS(O) \in \{JJ\}$	$o = O$	same as $R1_2$ with iPod as the known word and best as the extract word
$R3_1$	$T_{i(j)} \rightarrow T_{i(j)}$-*Dep* $\rightarrow T_{j(i)}$ *s.t.* $T_{j(i)} \in \{T\}$,$T_{i(j)}$-*Dep* $\in \{CONJ\}$, $POS(T_{i(j)}) \in \{NN\}$	$t = T_{i(j)}$	Does the player play dvd with audio and "video"? (*video* $\rightarrow$ *conj* $\rightarrow$ *audio*)
$R3_2$	$T_i \rightarrow T_i$-*Dep* $\rightarrow H \leftarrow T_j$-*Dep* $\leftarrow T_j$ *s.t.* $T_i \in \{T\}$,T_i-*Dep*==T_j-*Dep*, $POS(T_j) \in \{NN\}$	$t = T_j$	Canon "G3" has a great lens. (*lens* $\rightarrow$ *obj* $\rightarrow$ *has* $\leftarrow$ *subj* $\leftarrow$ *G3*)
$R4_1$	$O_{i(j)} \rightarrow O_{i(j)}$-*Dep* $\rightarrow O_{j(i)}$ *s.t.* $O_{j(i)} \in \{O\}$, $O_{i(j)}$-*Dep* $\in \{CONJ\}$, $POS(O_{i(j)}) \in \{JJ\}$	$o = O_{i(j)}$	The camera is amazing and "easy" to use. (*easy* $\rightarrow$ *conj* $\rightarrow$ *amazing*)
$R4_2$	$O_i \rightarrow O_i$-*Dep* $\rightarrow H \leftarrow O_j$-*Dep* $\leftarrow O_j$ *s.t.* $O_i \in \{O\}$,O_i-*Dep*==O_j-*Dep*, $POS(O_j) \in \{JJ\}$	$o = O_j$	If you want to buy a sexy, "cool", accessory-available mp3 player, you can choose iPod. (*sexy* $\rightarrow$ *mod* $\rightarrow$ *player* $\leftarrow$ *mod* $\leftarrow$ *cool*)

Figure A.1: Qiu et al.(2011)'s rules used in Double Propagation system

Input: Opinion Word Dictionary $\{O\}$, Review Data R
Output: All Possible Features $\{F\}$, The Expanded Opinion Lexicon $\{O\text{-}Expanded\}$
Function:
1. $\{O\text{-}Expanded\} = \{O\}$
2. $\{F_i\} = \emptyset, \{O_i\} = \emptyset$
3. for each parsed sentence in R
4. if(Extracted features not in $\{F\}$)
5. Extract features $\{F_i\}$ using $R1_1$ and $R1_2$ based on opinion words in $\{O\text{-}Expanded\}$
6. endif
7. if(Extracted opinion words not in $\{O\text{-}Expanded\}$)
8 Extract new opinion words $\{O_i\}$ using $R4_1$ and $R4_2$ based on opinion words in $\{O\text{-}Expanded\}$
9. endif
10. endfor
11. Set $\{F\} = \{F\} + \{F_i\}$, $\{O\text{-}Expanded\} = \{O\text{-}Expanded\} + \{O_i\}$
12. for each parsed sentence in R
13. if(Extracted features not in $\{F\}$)
14. Extract features $\{F'\}$ using $R3_1$ and $R3_2$ based on features in $\{F_i\}$
15. endif
16. if(Extracted opinion words not in $\{O\text{-}Expanded\}$)
17. Extract opinion words $\{O'\}$ using $R2_1$ and $R2_2$ based on features in $\{F_i\}$
18. endif
19. end for
20. Set $\{F_i\} = \{F_i\} + \{F'\}$, $\{O_i\} = \{O_i\} + \{O'\}$
21. Set $\{F\} = \{F\} + \{F'\}$, $\{O\text{-}Expanded\} = \{O\text{-}Expanded\} + \{O'\}$
22. Repeat 2 till size($\{F_i\}$)=0, size($\{O_i\}$)=0

Figure A.2: Qiu et al.(2011)'s Double Propagation (DP) algorithm

Appendix B

Appendix B - Aspect Extraction systems - erroneous output

Table B.1: Errors in 120 reviews from the AE systems (linguistic and DL)

Review	Review text	Linguistic	DL
0	The bread is top notch as well	Type A Type B	
1	the fastest delivery times in the city		Type A
5	I trust the people at Go Sushi		Type A
6	Very decent Japanese food	Type A	
7	BEST spicy tuna roll , great asian salad	Type A Type A	
8	Try the rose roll (not on menu)	Type A	
9	Esp lychee martini	Type B	Type B
10	In fact , this was not a Nicoise salad	Type B	
11	it should n't take ten minutes to get your drinks	Type B	
12	the top - notch food and live entertainment sold us on a unforgettable evening	Type B Type B	
12	Live entertainment	Type A	Type A
14	The sangria 's - watered down	Type A	
16	all at a reasonable price	Type A	
18	The portions of the food	Type A	Type A
19	The two waitress's who looked like they had been sucking on lemons.	Type A Type B	Type A Type B
20	and the convienent parking at Chelsea Piers	Type A	
22	Tart of the day	Type A	Type A

Table B.1: Errors in 120 reviews from the AE systems (linguistic and DL)

Review	Review text	Linguistic	DL
23	I am surprised at the lower reviews; it is definitely better than other places I have tried	Type B Type B	
24	the secret was that we went on a Sunday night	Type B	
26	WE ENDED UP IN LITTLE ITALY IN LATE AFTERNOON	Type B	
27	We were pleasantly surprised with our choice	Type B	Type B
29	Array of sushi	Type A	Type A
31	If you 're craving some serious indian food	Type A	
33	have always had a great time here	Type B	
34	The service was delightful, the garden adorable		Type A
34	the food (from appetizers to entrees) was delectable	Type A Type A	
35	You will be very happy with the experience	Type B	
37	it provides power lunch and dinners	Type B	
38	Two wasted steaks	Type A	
39	The staff should be a bit more friendly	Type B	
40	I absolutely suggest this place	Type B	
42	If you want good tasting well seasoned latin food	Type A	Type B Type B
43	Definitely try the taglierini with truffles	Type A	
45	this is a great new restaurant that has been consistently good every time	Type B Type B	
46	The gnocchi literally melts in your mouth	Type A	
47	Had a great experience at Trio	Type B	
47	Food was tasty and large in portion size		Type A
47	I would highly recommend the portobello / gorgonzola / sausage appetizer and the lobster Risotto	Type A	Type A
48	Entrees include classics like lasagna	Type A	Type A
51	The pizza with soy cheese	Type A	Type A
54	you do not want to miss this place	Type B	
55	The food is top notch	Type B	
56	I 've been coming here on and off for the past five years	Type B	
58	this is still one of my most favorite restaurants	Type B	
58	(kimono shrimp special was excellent)	Type A	
60	The whole experience was satisfying	Type B	Type B
62	The menu is interesting and quite reasonably priced	Type A	

Table B.1: Errors in 120 reviews from the AE systems (linguistic and DL)

Review	Review text	Linguistic	DL
64	the food was n't cooked fresh	Type A	
64	It was obviously made before hand and then reheated		Type B
66	The spicy mussels are a highlight	Type A	
68	Would not return for the amount we paid		Type B
69	being foodies , we were utterly disappointed with the food	Type B	
81	While it was large and a bit noisy	Type B	
82	Some of the curried casseroles can be a trifle harsh	Type A	
85	I was pleasantly surprised at the taste		Type A
86	Its chosen cuisine make Mare a great choice for seafood lovers	Type A Type B	Type B
87	I never had an orange do nut before	Type A	Type A
88	they really provide a relaxing , laid - back atmosphere	Type B Type B	
89	This particular location certainly uses substandard meats	Type B	
90	I stumbled upon this resteraunt on my way home from the subway	Type B Type B Type B	
91	The Management was less than accomodating	Type A	
92	The ambience is also more laid - back	Type A	
94	mojitos and the service are the best part in there	Type B	
95	Sandwiches , burgers and salads , like the lemon - dressed cobb , are classic successes	Type A Type A Type A Type A	
95	the lemon - dressed cobb		Type A
96	this restaurant is absolutely beautiful	Type B	
100	The worst though was the taste		Type A
103	Stay with the roasted chickens and you 'll be fine	Type B	
105	The steak melted in my mouth	Type A	
107	The food did take a few extra minutes to come but the cute waiters ' jokes and friendliness made up for it	Type A	

Table B.1: Errors in 120 reviews from the AE systems (linguistic and DL)

Review	Review text	Linguistic	DL
108	Most importantly , it is reasonably priced	Type B	
109	The selection of food is excellent I 'm not used to having much choice at restaurants	Type A	
110	Only suggestion is that you skip the dessert	Type B	
110	it was overpriced and fell short on taste	Type A	Type A
112	I have no idea why this restaurant is so often overlooked	Type B Type B	
113	you feel like you 're in the perfect place	Type B	
114	i do nt know what some people who rave about this hot dog are talking about	Type A	Type A
115	it is a hidden delight complete with a quaint bar and good food	Type B	
116	I find that most Kosher restaurants are average to good	Type B	
117	The waiters ALWAYS look angry and even ignore their high - tipping regulars	Type A	
118	a welcome escape from the rest of the SI mall .	Type B Type B	
119	Yes , they 're a bit more expensive then typical , but then again , so is their food .	Type B	
120	Not terrible , but not the restaurant in the reviews of 2002	Type B	

Appendix C

Appendix C: Aspect Category Detection systems - erroneous output

Table C.1: Errors in 200reviews from the ACD systems (linguistic and DL)

R	**Text**	**Gold Labels**	**Ling. system predictions**	**Type**	**DL-system predictions**	**Type**
4	Certainly not the best sushi in New York, however, it is always fresh, and the place is very clean, sterile.	Food ambience			food	Type A
5	I trust the people at Go Sushi, it never disappoints.	anecdotes			food	Type B
11	While there's a decent menu, it shouldn't take ten minutes to get your drinks and 45 for a dessert pizza.	Food service			food	Type A
12	Once we sailed, the top-notch food and live entertainment sold us on a unforgettable evening.	Food ambience	Food	Type A	-	Type A Type A
20	From the beginning, we were met by friendly staff memebers, and the convienent parking at Chelsea Piers made it easy for us to get to the boat.	service anecdotes/ m.	service	Type A		
21	We enjoyed ourselves thoroughly and will be going back for the desserts	Food anecdotes/	- m.	Type A Type A	Food	Type A

Table C.1: Errors in 200reviews from the ACD systems (linguistic and DL)

R	**Text**	**Gold Labels**	**Ling. system predictions**	**Type**	**DL-system predictions**	**Type**
24	Maybe the secret was that we went on a Sunday night and everything was great.	anecdotes/m.	Ambience food	Type B Type B	-	Type A
33	have always had a great time here.	anecdotes/m.	-	Type A		
34	It was pleasantly uncrowded, the service was delightful, the garden adorable, the food (from appetizers to entrees) was delectable.	Ambience Service food	Food Service	Type A	Food Service	Type A
37	How pretentious and inappropriate for MJ Grill to claim that it provides power lunch and dinners!	food	price	Type B		
38	Two wasted steaks – what a crime!	food	-	Type A		
47	Had a great experience at Trio ... staff was pleasant; food was tasty and large in portion size - I would highly recommend the portobello/gorgonzola/sausage appetizer and the lobster risotto.	Service food	food	Type A	food	Type A
50	Meal was very expensive for what you get.	Food price	price	Type A		
52	Good food at the right price, what more can you ask for.	Food price	food	Type A		
57	Great food, great waitstaff, great atmosphere, and best of all GREAT beer!	Food Service ambience			Ambience Food	Type A

Table C.1: Errors in 200reviews from the ACD systems (linguistic and DL)

R	**Text**	**Gold Labels**	**Ling. system predictions**	**Type**	**DL-system predictions**	**Type**
58	this is still one of my most favorite restaurants in the area the food is inexpensive but very good (kimono shrimp special was excellent) and has a great atmosphere.	Food Price ambience	Food ambience	Type A	Food ambience	Type A
60	The whole experience was satisfying.	anecdotes/ m.			-	
68	Would not return for the amount we paid.	price	anecdotes/ m.	Type B		
74	Stay away if you're claustrophobic	ambience	anecdotes/ m.	Type B	anecdotes/ m.	Type B
81	While it was large and a bit noisy, the drinks were fantastic, and the food was superb.	Ambience food	Food	Type A		
83	I wasn't impressed and it wasn't SPICEY????	food	anecdotes/ m.	Type B	anecdotes/ m.	Type B
87	I never had an orange donut before so I gave it a shot.	food	anecdotes/ m.	Type B	anecdotes/ m.	Type B
94	mojitos and the service are the best part in there	Food service	service	Type A		
107	The food did take a few extra minutes to come, but the cute waiters' jokes and friendliness made up for it.	service	-	Type A	-	Type A
110	Only suggestion is that you skip the dessert, it was overpriced and fell short on taste.	Food price			price	Type A
113	From the moment you walk in, you feel like you're in the perfect place.	anecdotes/ m.	ambience	Type B	-	Type A
114	i dont know what some people who rave about this hot dog are talking about.	food	anecdotes/ m.	Type B	anecdotes/ m.	Type B
115	it is a hidden delight complete with a quaint bar and good food.	Food ambience	food	Type A	food	Type A

Table C.1: Errors in 200reviews from the ACD systems (linguistic and DL)

R	**Text**	**Gold Labels**	**Ling. system predictions**	**Type**	**DL-system predictions**	**Type**
116	I find that most Kosher restaurants are average to good, but this has been the best I've eaten so far.	anecdotes/m.			food	Type B
119	Yes, they're a bit more expensive then typical, but then again, so is their food.	Price food			price	Type A
127	As we waited I watched 3 seperate groups of diners discuss how dissapointed they also were.	anecdotes/m.			service	Type B
130	The fettucino alfredo was amazing.	food			-	Type A
134	Even when the chef is not in the house, the food and service are right on target.	Food service			service	Type A
139	I highly reccomend the grand marnier shrimp, it's insanely good.	food	-	Type A		
145	The sushi is cut in blocks bigger than my cell phone.	food	-	Type A		
164	Waiters are very friendly and the pasta is out of this world.	Service food	service	Type A		
169	The sauce may not be for everyone, since it is distintive.	food	-	Type A		
170	My husband and I have been there at least 6 times and we've always been given the highest service and often free desserts.	Service price			service	Type A
172	Curioni's Pizza has been around since the 1920's.	anecdotes/m.			food	Type B
177	There was only one waiter for the whole restaurant upstairs.	service	anecdotes/m.	Type B	anecdotes/m.	Type B
180	this without question is one of the worst hotdogs i have ever had.	food	-	Type A	anecdotes/m.	Type B

Table C.1: Errors in 200reviews from the ACD systems (linguistic and DL)

R	**Text**	**Gold Labels**	**Ling. system predictions**	**Type**	**DL-system predictions**	**Type**
181	The staff is unbelievably friendly, and I dream about their Saag gosht...so good.	Service food			service	Type A
190	I can't stand this dungeon.	ambience	anecdotes/m.	Type B	anecdotes/m.	Type B
194	Delish and made to your liking!	food	-	Type A	anecdotes/m.	Type B
196	For the price you pay for the food here, you'd expect it to be at least on par with other Japanese restaurants.	Price food			price	Type A

Appendix D

Appendix D - Full Performance Reports

D.0.1 Aspect Extraction System reports

Linguistic-based system

	precision	**recall**	**f1-score**	**support**
B	0.53	0.63	0.58	1132
I	0.68	0.26	0.38	571
O	0.94	0.95	0.94	11049
accuracy			0.89	12752
macro avg	0.72	0.61	0.63	12752
weighted avg	0.89	0.89	0.89	12752

Table D.1: Qiu et al. performance metrics using strict IOB labeling

	precision	**recall**	**f1-score**	**support**
I	0.64	0.59	0.61	1703
O	0.94	0.95	0.94	11049
accuracy			0.90	12752
macro avg	0.79	0.77	0.78	12752
weighted avg	0.90	0.90	0.90	12752

Table D.2: Qiu et al. performance metrics using relaxed IO labeling

Deep learning-based system

	precision	**recall**	**f1-score**	**support**
B	0.84	0.87	0.85	1132
I	0.89	0.56	0.69	571
O	0.97	0.99	0.98	11049
accuracy			0.96	12752
macro avg	0.90	0.81	0.84	12752
weighted avg	0.96	0.96	0.96	12752

Table D.3: Xu et al. performance metrics using strict IOB labeling

	precision	**recall**	**f1-score**	**support**
I	0.92	0.83	0.88	1703
O	0.97	0.99	0.98	11049
accuracy			0.97	12752
macro avg	0.95	0.91	0.93	12752
weighted avg	0.97	0.97	0.97	12752

Table D.4: Xu et al. performance metrics using relaxed IO labeling

D.0.2 Aspect Category Detection System reports - Linguistic-based

ambience

	precision	recall	f1-score	support
0	0.95	0.96	0.96	682
1	0.77	0.73	0.75	118
accuracy			0.93	800
macro avg	0.86	0.85	0.85	800
weighted avg	0.93	0.93	0.93	800

anecdotes/misc

	precision	recall	f1-score	support
0	0.89	0.89	0.89	566
1	0.74	0.74	0.74	234
accuracy			0.85	800
macro avg	0.82	0.82	0.82	800
weighted avg	0.85	0.85	0.85	800

food

	precision	recall	f1-score	support
0	0.88	0.92	0.90	382
1	0.93	0.89	0.90	418
accuracy			0.90	800
macro avg	0.90	0.90	0.90	800
weighted avg	0.90	0.90	0.90	800

price

	precision	recall	f1-score	support
0	0.97	0.99	0.98	717
1	0.94	0.76	0.84	83
accuracy			0.97	800
macro avg	0.96	0.88	0.91	800
weighted avg	0.97	0.97	0.97	800

service

	precision	recall	f1-score	support
0	0.97	0.97	0.97	628
1	0.90	0.88	0.89	172
accuracy			0.95	800
macro avg	0.94	0.93	0.93	800
weighted avg	0.95	0.95	0.95	800

Table D.5: Kiritchenko et al. performance metrics 5 x binary classifiers

D.0.3 Aspect Category Detection report - DL-based

ambience

	precision	recall	f1-score	support
0	0.96	0.96	0.96	682
1	0.79	0.77	0.78	118
accuracy			0.94	800
macro avg	0.88	0.87	0.87	800
weighted avg	0.94	0.94	0.94	800

anecdotes/misc

	precision	recall	f1-score	support
0	0.89	0.92	0.90	566
1	0.78	0.71	0.75	234
accuracy			0.86	800
macro avg	0.83	0.82	0.82	800
weighted avg	0.85	0.86	0.86	800

food

	precision	recall	f1-score	support
0	0.92	0.91	0.91	382
1	0.92	0.93	0.92	418
accuracy			0.92	800
macro avg	0.92	0.92	0.92	800
weighted avg	0.92	0.92	0.92	800

price

	precision	recall	f1-score	support
0	0.97	0.99	0.98	717
1	0.94	0.76	0.84	83
accuracy			0.97	800
macro avg	0.96	0.88	0.91	800
weighted avg	0.97	0.97	0.97	800

service

	precision	recall	f1-score	support
0	0.96	0.98	0.97	628
1	0.93	0.87	0.89	172
accuracy			0.96	800
macro avg	0.94	0.92	0.93	800
weighted avg	0.96	0.96	0.96	800

Table D.6: Xue et al. performance metrics 5 x binary classifiers

Bibliography

Y. Bengio, P. Simard, and P. Frasconi. Learning long-term dependencies with gradient descent is difficult. *IEEE transactions on neural networks*, 5(2):157–166, 1994.

Y. Bengio, R. Ducharme, P. Vincent, and C. Jauvin. A neural probabilistic language model. *Journal of machine learning research*, 3(Feb):1137–1155, 2003.

P. F. Brown, V. J. Della Pietra, P. V. Desouza, J. C. Lai, and R. L. Mercer. Class-based n-gram models of natural language. *Computational linguistics*, 18(4):467–480, 1992.

C. Brun, D. N. Popa, and C. Roux. Xrce: Hybrid classification for aspect-based sentiment analysis. In *SemEval@ COLING*, pages 838–842. Citeseer, 2014.

E. Cambria, S. Poria, A. Gelbukh, and M. Thelwall. Sentiment analysis is a big suitcase. *IEEE Intelligent Systems*, 32(6):74–80, 2017.

E. Cambria, S. Poria, D. Hazarika, and K. Kwok. Senticnet 5: Discovering conceptual primitives for sentiment analysis by means of context embeddings. In *Thirty-Second AAAI Conference on Artificial Intelligence*, 2018.

G. Carenini, R. T. Ng, and E. Zwart. Extracting knowledge from evaluative text. In *Proceedings of the 3rd international conference on Knowledge capture*, pages 11–18, 2005.

R. Collobert, J. Weston, L. Bottou, M. Karlen, K. Kavukcuoglu, and P. Kuksa. Natural language processing (almost) from scratch. *Journal of Machine Learning Research*, 12(76):2493–2537, 2011. URL `http://jmlr.org/papers/v12/collobert11a.html`.

K. Dave, S. Lawrence, and D. M. Pennock. Mining the peanut gallery: Opinion extraction and semantic classification of product reviews. In *Proceedings of the 12th international conference on World Wide Web*, pages 519–528, 2003.

J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

H. H. Do, P. Prasad, A. Maag, and A. Alsadoon. Deep learning for aspect-based sentiment analysis: a comparative review. *Expert Systems with Applications*, 118: 272–299, 2019.

J. R. Firth. A synopsis of linguistic theory, 1930-1955. *Studies in linguistic analysis*, 1957.

G. Ganu, N. Elhadad, and A. Marian. Beyond the stars: improving rating predictions using review text content. In *WebDB*, volume 9, pages 1–6. Citeseer, 2009.

Y. Goldberg. A primer on neural network models for natural language processing. *Journal of Artificial Intelligence Research*, 57:345–420, 2016.

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

D. Jurafsky and J. Martin. Speech and language processing (3rd (draft) ed.), 2019.

D. Jurafsky and J. H. Martin. Speech and language processing: An introduction to speech recognition, computational linguistics and natural language processing. *Upper Saddle River, NJ: Prentice Hall*, 2008.

S. Kiritchenko, X. Zhu, C. Cherry, and S. Mohammad. Nrc-canada-2014: Detecting aspects and sentiment in customer reviews. In *Proceedings of the 8th international workshop on semantic evaluation (SemEval 2014)*, pages 437–442, 2014.

Y. LeCun, Y. Bengio, et al. Convolutional networks for images, speech, and time series. *The handbook of brain theory and neural networks*, 3361(10):1995, 1995.

Y. LeCun, Y. Bengio, and G. Hinton. Deep learning. *nature*, 521(7553):436–444, 2015.

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

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

Q. Liu, Z. Gao, B. Liu, and Y. Zhang. A logic programming approach to aspect extraction in opinion mining. In *2013 IEEE/WIC/ACM International Joint Conferences on Web Intelligence (WI) and Intelligent Agent Technologies (IAT)*, volume 1, pages 276–283. IEEE, 2013.

E. Marrese-Taylor and Y. Matsuo. Replication issues in syntax-based aspect extraction for opinion mining. *arXiv preprint arXiv:1701.01565*, 2017.

L. Meng, R. Huang, and J. Gu. A review of semantic similarity measures in wordnet. *International Journal of Hybrid Information Technology*, 6(1):1–12, 2013.

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

G. A. Miller, R. Beckwith, C. Fellbaum, D. Gross, and K. J. Miller. Introduction to wordnet: An on-line lexical database. *International journal of lexicography*, 3(4): 235–244.

K. Nigam, A. K. McCallum, S. Thrun, and T. Mitchell. Text classification from labeled and unlabeled documents using em. *Machine learning*, 39(2-3):103–134, 2000.

O. Owoputi, B. O'Connor, C. Dyer, K. Gimpel, N. Schneider, and N. A. Smith. Improved part-of-speech tagging for online conversational text with word clusters. In *Proceedings of the 2013 conference of the North American chapter of the association for computational linguistics: human language technologies*, pages 380–390, 2013.

G. Paltoglou and M. Thelwall. Sensing social media: A range of approaches for sentiment analysis. In *Cyberemotions*, pages 97–117. Springer, 2017.

B. Pang and L. Lee. Opinion mining and sentiment analysis. *Foundations and trends in information retrieval*, 2(1-2):1–135, 2008.

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

M. Pontiki, D. Galanis, J. Pavlopoulos, H. Papageorgiou, I. Androutsopoulos, and S. Manandhar. Semeval-2014 task 4: Aspect based sentiment analysis. proc. 8th int. workshop on semantic evaluation (semeval 2014). 2014.

M. Pontiki, D. Galanis, H. Papageorgiou, S. Manandhar, and I. Androutsopoulos. Semeval-2015 task 12: Aspect based sentiment analysis. In *Proceedings of the 9th international workshop on semantic evaluation (SemEval 2015)*, pages 486–495, 2015.

M. Pontiki, D. Galanis, H. Papageorgiou, I. Androutsopoulos, S. Manandhar, M. Al-Smadi, M. Al-Ayyoub, Y. Zhao, B. Qin, O. De Clercq, et al. Semeval-2016 task 5: Aspect based sentiment analysis. In *10th International Workshop on Semantic Evaluation (SemEval 2016)*, 2016.

A.-M. Popescu and O. Etzioni. Extracting product features and opinions from reviews. In *Natural language processing and text mining*, pages 9–28. Springer, 2007.

S. Poria, E. Cambria, and A. Gelbukh. Aspect extraction for opinion mining with a deep convolutional neural network. *Knowledge-Based Systems*, 108:42–49, 2016.

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

S. Raju, P. Pingali, and V. Varma. An unsupervised approach to product attribute extraction. In *European Conference on Information Retrieval*, pages 796–800. Springer, 2009.

P. Ray and A. Chakrabarti. A mixed approach of deep learning method and rule-based method to improve aspect level sentiment analysis. *Applied Computing and Informatics*, 2019.

D. E. Rumelhart, G. E. Hinton, and R. J. Williams. Learning representations by back-propagating errors. *nature*, 323(6088):533–536, 1986.

C. Sun, L. Huang, and X. Qiu. Utilizing bert for aspect-based sentiment analysis via constructing auxiliary sentence. *arXiv preprint arXiv:1903.09588*, 2019.

Z. Toh and J. Su. Nlangp at semeval-2016 task 5: Improving aspect based sentiment analysis using neural network features. In *Proceedings of the 10th international workshop on semantic evaluation (SemEval-2016)*, pages 282–288, 2016.

Z. Toh and W. Wang. Dlirec: Aspect term extraction and term polarity classification system. In *Association for Computational Linguistics and Dublin City University*. Citeseer, 2014.

M. M. Trusca, D. Wassenberg, F. Frasincar, and R. Dekker. A hybrid approach for aspect-based sentiment analysis using deep contextual word embeddings and hierarchical attention. *arXiv preprint arXiv:2004.08673*, 2020.

A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008, 2017.

J. Wagner, P. Arora, S. Cortes, U. Barman, D. Bogdanova, J. Foster, and L. Tounsi. Dcu: Aspect-based polarity classification for semeval task 4. 2014.

Y. Wang, M. Huang, X. Zhu, and L. Zhao. Attention-based lstm for aspect-level sentiment classification. In *Proceedings of the 2016 conference on empirical methods in natural language processing*, pages 606–615, 2016.

J. Wiebe, R. Bruce, M. Bell, M. Martin, and T. Wilson. A corpus study of evaluative and speculative language. In *Proceedings of the Second SIGdial Workshop on Discourse and Dialogue*, 2001.

J. Wiebe, T. Wilson, and C. Cardie. Annotating expressions of opinions and emotions in language. *Language resources and evaluation*, 39(2-3):165–210, 2005.

Y. Wu, Q. Zhang, X. Huang, and L. Wu. Phrase dependency parsing for opinion mining. In *Proceedings of the 2009 conference on empirical methods in natural language processing: Volume 3-volume 3*, pages 1533–1541. Association for Computational Linguistics, 2009.

H. Xu, B. Liu, L. Shu, and P. S. Yu. Double embeddings and cnn-based sequence labeling for aspect extraction. *arXiv preprint arXiv:1805.04601*, 2018.

H. Xu, B. Liu, L. Shu, and P. S. Yu. Bert post-training for review reading comprehension and aspect-based sentiment analysis. *arXiv preprint arXiv:1904.02232*, 2019.

W. Xue, W. Zhou, T. Li, and Q. Wang. Mtna: a neural multi-task model for aspect category classification and aspect term extraction on restaurant reviews. In *Proceedings of the Eighth International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 151–156, 2017.

Z. Zhai, B. Liu, H. Xu, and P. Jia. Grouping product features using semi-supervised learning with soft-constraints. In *Proceedings of the 23rd international conference on computational linguistics*, pages 1272–1280. Association for Computational Linguistics, 2010.

L. Zhang and B. Liu. Aspect and entity extraction for opinion mining. In *Data mining and knowledge discovery for big data*, pages 1–40. Springer, 2014.