ALDONA: A Hybrid Solution for Sentence-Level Aspect-Based Sentiment Analysis Using a Lexicalised Domain Ontology and a Neural Attention Model

Donatas Meškelė
Erasmus University Rotterdam
Rotterdam, the Netherlands
donatas.meskele@gmail.com

Flavius Frasincar
Erasmus University Rotterdam
Rotterdam, the Netherlands
frasincar@ese.eur.nl

ABSTRACT

Sentences containing several different polarity aspects cause one of the main problems in sentiment analysis. Depending on an aspect, the same context words can have different effects on its sentiment value. Additionally, the polarity can be influenced by the domain-specific knowledge, showing the necessity to incorporate it into the sentiment classification. In this paper we present a hybrid solution for sentence-level aspect-based sentiment analysis using A Lexicalised Domain Ontology and Neural Attention (ALDONA) model to handle the problems mentioned above. To measure the influence of each word in a given sentence on an aspect's polarity, we introduce the bidirectional context attention mechanism. Moreover, the classification module is designed to handle the sentence's complex structure. Finally, the manually created lexicalised domain ontology (represented in OWL) is integrated to exploit the field-specific knowledge. Computational results obtained on a benchmark data set based on Web reviews have shown ALDONA's ability to outperform several state-of-the-art models and stress its contribution to aspect-based sentiment classification.

CCS CONCEPTS

• **Information systems** → **Sentiment analysis**; *Information extraction*; Web mining;

KEYWORDS

Aspect-based sentiment classification, Hybrid model, Lexicalised domain ontology, Bidirectional gated neural network

ACM Reference Format:
Donatas Meškelė and Flavius Frasincar. 2019. ALDONA: A Hybrid Solution for Sentence-Level Aspect-Based Sentiment Analysis Using a Lexicalised Domain Ontology and a Neural Attention Model. In *The 34th ACM/SIGAPP Symposium on Applied Computing (SAC '19), April 8–12, 2019, Limassol, Cyprus.* ACM, New York, NY, USA, 8 pages. https://doi.org/10.1145/3297280.3297525

SAC '19, April 8–12, 2019, Limassol, Cyprus

ACM ISBN 978-1-4503-5933-7/19/04…$15.00
https://doi.org/10.1145/3297280.3297525

1 INTRODUCTION

As one of the core natural language processing (NLP) problems, sentiment analysis (SA) has been broadly assessed using different approaches, such as aspect extraction [6, 26], opinion identification [9], and aspect-based sentiment classification [24, 25, 32]. However, since each of these sub-tasks requires deep analysis, we solely focus on improving accuracy of the latter. Sentiment classification is usually performed using one of the three main methods: knowledge-based, machine learning/neural networks, or hybrid.

A hybrid sentiment classifier consolidates the knowledge-based and statistical (machine learning/neural networks) approaches, providing a powerful tool for sentiment analysis [4, 29, 30]. A hybrid two-step aspect-based sentiment classification method employing a lexicalised domain ontology and using the support vector machine (SVM) as a backup mechanism has been presented by [29]. Considering the obtained promising results, this paper exploits the proposed idea. However, due to the increased potential of neural networks [5, 15, 19], we replace SVM with a more powerful neural attention model. Inspired by [19, 33], we propose the bidirectional context attention mechanism and the classification module to boost the accuracy of aspect-based sentiment classification. As a result, we introduce A Lexicalised Domain Ontology and Neural Attention (ALDONA) model. Split into the two-step procedure, ALDONA initially utilises the field-specific knowledge to determine the polarity of a given aspect. If the sentiment value is not obtained (due to some missing concepts, properties, or polarity categories) or more than one polarity category is predicted, the neural attention model is employed. Field-specific knowledge is captured using a lexicalised domain ontology introduced by [29]. While there are several methods to create an ontology [1], the manually created ontology approach is chosen to ensure that all relations among entities and their properties are correctly defined. The ontology classification algorithm is constructed to be able to differentiate among three sentiment types, namely, generic, category-dependent, and context-dependent. Based on its type, a sentiment can be classified into *Positive* and *Negative* classes. The *Neutral* class is omitted, as it is not well represented in data sets.

The neural attention model (also referred as Deep Bidirectional Gated Recurrent Unit (DBGRU)) is split into word embeddings, the bidirectional context attention mechanism, sentence-level content attention mechanism, and the classification module. The former part converts sentences and their aspects to their embedded forms, the second block assigns weights for each word in a sentence based on the word ordering, correlations among aspects and their context words, as well as the past and future information. The third block summarises the obtained information in the weighted embedding sentence vector. The classification module captures the meaning of the whole sentence by explicitly defining relationships among an aspect and weighted sentence representations.

The computational results have shown that ALDONA has made a meaningful contribution to aspect-based sentiment classification by outperforming several state-of-the-art models, such as the ontology classification model (Ont) [29], content attention model (BaseA) [19], sentence-level content attention model (BaseB) [19], sentence-level position attention model (BaseC) [19], and Content Attention Based Aspect based Sentiment Classification model (CABASC) [19]. CABASC is a state-of-the-art aspect-based sentiment analysis model that proved to be superior to many of the previously designed neural models for the task [19]. The other baseline models employ the context attention module [19] combined with unidirectional long short-term memory module (CTX-LSTM), bidirectional long short-term memory module (CTX-BLSTM), or bidirectional gated recurrent unit (CTX-BGRU).

The paper is structured as follows. In Sec. 2 some related literature is discussed. Then in Sec. 3 the proposed model, ALDONA, is presented in detail. The data is described in Sec. 4 and performance of the proposed model is compared with other models in Sec. 5. Conclusion and future research are presented in Sec. 6. The source code of all implemented models is provided at https://github.com/donmesh/ALDONA.

2 RELATED LITERATURE

Sentiment analysis (SA) has been broadly presented in [18]. Due to its complexity, the task is usually considered as a combination of sub-problems such as aspect extraction [6, 26], opinion identification [9], and aspect-based sentiment classification [24, 25, 32]. The latter sub-task is elaborately presented by the authors of [28] and is the main focus in our research.

Sentiment classification is usually approached by one of the main classification methods: knowledge-based, machine learning/neural network, or hybrid combining the former two. Knowledge-driven sentiment classification incorporates the domain specific knowledge inferred from a given ontology. By respecting the set of design criteria proposed by [11], ontologies can be created manually [29, 30], semi-automatically [8], or fully automatically [3]. The defined ontology concepts, their properties, and relations among them can be used to infer new relations by means of the ontology reasoner (e.g., if *pizza* is *savoury food* and *savoury food* is *food*, then *pizza* is *food*). Carefully designed and broadly-spanning ontologies have been shown to produce promising results in aspect-based sentiment classification [8, 29].

On the other hand, machine learning classification depends on statistical relations derived from given feature vectors. While the bag-of-words (BoW) with the support vector machine (SVM) is one of the simplest, yet surprisingly effective methods [12, 21, 22], it has some drawbacks. The main weaknesses are the disregarded word ordering and the ignored semantics of words [17]. Moreover, the performance of classical machine learning models is highly dependent on the manual feature engineering. Because of this expensive process, neural attention mechanisms have been employed in the recent research [16].

The relative aspect-context position mechanism has been presented by [32]. However, the assumption that context words which are closer to a given aspect are more important than context words which are further away is not the general truth and thus can lead to lower generalisation ability. Due to the promising results in the language sequence optimisation, the recurrent neural network (RNN) has been incorporated in the context attention mechanisms [27, 31, 33]. Extracting information from the left and right context words with respect to a given aspect [31], as well as exploiting syntax and semantics of a given sentence by means of the bidirectional gated neural network [33], have proved to be highly useful.

Hybrid models, combining field-specific knowledge with statistical relations, have been effectively used for sentiment analysis [4, 30]. Authors of [29] propose a lexicalised domain ontology classification algorithm backed up with the support vector machine (SVM). In the first phase the knowledge-based classifier assigns a given sentiment to either a *Positive* or *Negative* class depending on its inferred type (generic sentiments, category-dependent sentiments, or context-dependent sentiments). In case both or none of the classes are selected, the SVM bag-of-words classification is performed in the second stage. Due to the auspicious results, the proposed idea is exploited in this paper. Nonetheless, the main emphasis is put on the second phase, as a result of the high potential of neural networks [5, 15] and their gated representations [19, 33]. In this research we propose the bidirectional context attention mechanism and the classification module to cope with sentence complexity. By combining them with knowledge derived from the lexicalised domain ontology we introduce the Lexicalised Domain Ontology and Neural Attention model (ALDONA).

3 METHODOLOGY

By making use of the lexicalised domain ontology (represented in the Web Ontology Language (OWL)), the bidirectional context attention mechanism, sentence-level content attention mechanism, and the classification module, ALDONA incorporates the field-specific knowledge, word ordering, correlations among aspects and their context words, as well as the past and future information to determine the sentiment value of a given aspect.

3.1 Lexicalised Domain Ontology

The purpose of the lexicalised domain ontology is to define relationships among various entities and their properties. By means of a reasoner, not directly defined relationships can still be obtained and used to infer the sentiment value of a given aspect.

The manually created ontology contains three main classes: *SentimentMention*, *SentimentValue*, and *AspectMention*. The latter class is responsible for modelling the mentions of aspects, the *SentimentValue* defines the polarity of an aspect which can be either positive (then the sentiment is a subclass of *Positive)* or negative (then the sentiment is a subclass of *Negative*). As the edge values between the *Positive* and *Neutral*, or *Neutral* and *Negative* classes are not intuitively defined, the *Neutral* class has not been implemented in this ontology. The neutral sentiment is also not so well represented in data sets, as usually people take a stand of liking or disliking. The *SentimentMention* class models the expressions of sentiments. It can differentiate three sentiment types and apply specific rules on each of them.

Type-1: generic sentiments which always have the same sentiment value independently of an aspect class they describe (e.g., "awesome" is *Positive*). Type-2: category-dependent sentiments which can be used when a specific category of aspects is present (e.g., "delicious" is *Positive* and can be used to describe *SustenanceMention* (food and drinks), but does not apply for *ServiceMention*). Hence, type-2 sentiment will be inferred only if the aspect class matches the sentiment class, and will be ignored otherwise. Type-3: context-dependent sentiments which directly depend on the aspect category (e.g., *cold beer* is *Positive*, while *cold pizza* is *Negative*). If a certain aspect-sentiment combination does not exist, a new subclass is created.

Since the ontology is lexicalised each concept has an annotation of type *lex* attached to it. As the result, each concept in the ontology might have multiple lexicalisations (e.g., the United States of America can be referred as "the United States of America", "USA", or "US"). Moreover, concepts which are subclasses of the *SentimentValue* class have an *antonym* property which is used when the concept is negated. Additionally to the negation relations in the word dependency graph, for each word in a sentence the three preceding words are inspected to check whether or not they are present in a set of negating words, namely {*not, no, never, isn't, aren't, won't, wasn't, weren't, haven't, hasn't, don't, doesn't, can't, couldn't*} [14]. The presence of these words changes the sentiment dependency from the *Positive* subclass to the *Negative* subclass and vice versa.

The sentiment value prediction procedure consists of two main steps: analysing each word in a given sentence by verifying its type, and determining whether or not it is negated. The types are ordered and exclusive, meaning that the word dependency to the next type will be examined only if the current type is not applicable (e.g., a specific word can only have type-2 if does not have type-1). If the ontology reasoner is not able to infer the sentiment value, the polarity is determined by the neural attention model introduced below.

3.2 Neural Attention Model

The neural attention model (DBGRU) consists of four main parts: word embeddings, the bidirectional context attention mechanism, sentence-level content attention mechanism, and the classification module. Each of them is built on top of each other and is introduced in the following subsections.

3.2.1 Word Embeddings. Let $S = \{s_1, s_2, ..., s_N\}$ be an input sentence of length N containing words s_n, and $S_a = \{s_i, s_{i+1}, ..., s_{i+L}\}$ be an aspect of length L in that sentence, where $L \geqslant 1$ represents the aspect being a phrase. The embedding of a word s_n is constructed as follows:

$$e_n = \mathbb{L} o_n \in \mathbb{R}^d, \tag{1}$$

where $o_n \in \mathbb{R}^{|V|}$ is a one-hot vector, $\mathbb{L}^{d\times|V|}$ is the embedding matrix, d is the length of a numeric vector representing each word, and V is a dictionary containing all known words. Hence, the embedded sentence S is represented by:

$$E = [e_1, e_2, ..., e_N] \in \mathbb{R}^{d\times N} \tag{2}$$

with the embedded aspect:

$$E_A = [e_i, e_{i+1}, ..., e_{i+L}] \in \mathbb{R}^{d\times L}. \tag{3}$$

To extract the word ordering information we split the sentence S into two parts such that the first part is from the beginning of the sentence to the end of the aspect and the second part is from the beginning of the aspect to the end of the sentence (here, the aspect is represented by $\{s_i, ..., s_{i+L}\}$). We call those parts S_{LS} and S_{RS}:

$$\begin{aligned} S_{LS} &= \{s_1, ..., s_{i-1}, s_i, ..., s_{i+L}\}, \\ S_{RS} &= \{s_i, ..., s_{i+L}, s_{i+L+1}, ..., s_N\}, \end{aligned} \tag{4}$$

and their embedded versions (obtained using Eq. 1):

$$\begin{aligned} E_{LS} &= [e_1, ..., e_{i-1}, e_i, ..., e_{i+L}], \\ E_{RS} &= [e_i, ..., e_{i+L}, e_{i+L+1}, ..., e_N]. \end{aligned} \tag{5}$$

3.2.2 Bidirectional Context Attention Mechanism. The main deficiency of the unidirectional gated recurrent unit (proposed in [19]) is the inefficient information processing. The method accounts for the past information (information from words which come before the current word) to determine the current word, but does not incorporate the future information (information from words which come after the current word). The following example illustrates this issue:

processing direction ⟶	current	
Very tasty food,	service	is awful.
Very tasty food,	service	is amazing.

Figure 1: Example of the information inefficiency when the unidirectional recurrent network (RNN) is used. After sequentially processing the first two words it is difficult to determine what is the sentiment value of the current word *service*. It has two different meanings, but in the *current* cell they are identical.

Due to this inefficiency, the information obtained from these two sentences is identical when the current word is "service", as the model incorporates only the phrase "Very tasty food". The inefficient information problem can be handled by the bidirectional recurrent neural networks (BRNN) which exploit both past and future information to determine the attention weight of the current word. As the result, each word in a sentence is assessed from both directions.

In order to present the bidirectional context attention mechanism, we briefly introduce the unidirectional gated recurrent unit (GRU) [7]. The pipeline consists of the reset gate, update gate, new memory generation, and generation of the new hidden state. The influence of the previous hidden state h_{n-1} on the newly generated memory $\tilde{h}_n$ is determined by the reset gate r_n. The new hidden state h_n is then constructed from the output of the update gate u_n (which

processing direction ⟶	current	⟵ processing direction
Very tasty food,	service	is awful.
Very tasty food,	service	is amazing.

Figure 2: Example of the efficiently used information by applying the bidirectional recurrent neural network (BRNN). Words before the *current* cell and after the *current* cell are known when the *current* cell is evaluated. This allows to differentiate between these two sentences and determine the polarity of *service.*

adjusts the effect of the previous hidden state h_{n-1} and the new memory $\tilde{h}_n$). The mathematical GRU representation is given below:

$$\begin{aligned} r_n &= \sigma(W_r e_n + U_r h_{n-1} + b_r), \\ u_n &= \sigma(W_u e_n + U_u h_{n-1} + b_u), \\ \tilde{h}_n &= \tanh(W_h e_n + U_h(r_n \odot h_{n-1}) + b_{\tilde{h}}), \\ h_n &= u_n \odot h_{n-1} + (1 - u_n) \odot \tilde{h}_n, \end{aligned} \tag{6}$$

where $\odot$ represents the element-wise multiplication, σ and tanh are logistic sigmoid and hyperbolic tangent functions, $e_n \in \mathbb{R}^d$ is a word embedding vector, $r_n \in \mathbb{R}^d$ and $u_n \in \mathbb{R}^d$ are the reset and update gates, and $\tilde{h}_n \in \mathbb{R}^d$ and $h_n \in \mathbb{R}^d$ are the new memory and the new hidden state, respectively. $W_r \in \mathbb{R}^{d \times d}$, $U_r \in \mathbb{R}^{d \times d}$, $W_u \in \mathbb{R}^{d \times d}$, $U_u \in \mathbb{R}^{d \times d}$, $W_h \in \mathbb{R}^{d \times d}$, $U_h \in \mathbb{R}^{d \times d}$ are weight matrices and $b_r \in \mathbb{R}^d$, $b_u \in \mathbb{R}^d$, $b_{\tilde{h}} \in \mathbb{R}^d$ are bias vectors. An alternative to GRU is the long short-term memory model (LSTM) [13], which introduces additional flexibility.

The final hidden state obtained by the bidirectional gated recurrent unit (BGRU) is a combination of hidden states from both forward and backward directions. The general mathematical representation of the bidirectional recurrent neural network (BRNN) is given by:

$$\begin{aligned} \overrightarrow{h_n} &= f(\overrightarrow{\Theta} |\ e_n, \overrightarrow{h_{n-1}}), \\ \overleftarrow{h_n} &= f(\overleftarrow{\Theta} |\ e_n, \overleftarrow{h_{n+1}}), \\ h_n &= g(\overrightarrow{h_n},\ \overleftarrow{h_n}), \end{aligned} \tag{7}$$

where $\overrightarrow{h_n}$ and $\overleftarrow{h_n}$ are hidden states obtained from the forward and backward directions, e_n is the new input, $\overrightarrow{\Theta}$ and $\overleftarrow{\Theta}$ are parameters to be optimised (here, two sets of weight matrices and bias vectors described in Eq. 6), $f(\cdot)$ is the unidirectional recurrent neural network (here, GRU) and $g(\cdot)$ is the activation function combining $\overrightarrow{h_n}$ and $\overleftarrow{h_n}$. Here, h_n is defined as follows:

$$h_n = tanh(W_{fw} \overrightarrow{h_n} + W_{bw} \overleftarrow{h_n} + b_{bi}), \tag{8}$$

where $W_{fw} \in \mathbb{R}^{d \times d}$ and $W_{bw} \in \mathbb{R}^{d \times d}$ are weight matrices and $b_{bi} \in \mathbb{R}^d$ is a bias vector. We define the relationship between the forward and backward hidden states by using hyperbolic tangent as the activation function due to its high performance compared to traditional logistic sigmoid [20].

The left and right part embeddings, E_{LS} and E_{RS} respectively, are separately fed to the bidirectional gated recurrent unit (BGRU) introduced above. The produced outputs are:

$$\begin{aligned} H_{LS} &= [h_1, ..., h_{i_l - 1}, h_{i_l}, ..., h_{i_l + L_l}], \\ H_{RS} &= [h_{i_r}, ..., h_{i_r + L_r}, h_{i_r + L + 1}, ..., h_N]. \end{aligned} \tag{9}$$

By exploiting this information and employing the multilayer perceptron (MLP) we get bidirectional context attention weights for each word in the sentence S:

$$\begin{aligned} \beta_{LS} &= [\beta_1, ..., b_{i_l}, ..., b_{i_l + L_l}], \\ \beta_{RS} &= [\beta_{i_r}, ..., \beta_{i_r + L_r}, ..., \beta_N], \\ \beta_A &= [\tfrac{\beta_{i_l} + \beta_{i_r}}{2}, ..., \tfrac{\beta_{i_l + L_l} + \beta_{i_r + L_r}}{2}], \\ \beta_{LC} &= [\beta_1, ..., \beta_{i-1}], \\ \beta_{RC} &= [\beta_{i+L+1}, ..., \beta_N], \\ \beta &= [\beta_{LC}, \beta_A, \beta_{RC}], \end{aligned} \tag{10}$$

where in the first two steps attention weights β_{LS} and β_{RS} are calculated for H_{LS} and H_{RS}, both containing their sentence parts, as well as the aspect information. Their elements are computed as follows:

$$\begin{aligned} \beta_l &= \sigma(W_1 h_l + b_1) + b_l, \\ \beta_r &= \sigma(W_2 h_r + b_2) + b_r, \end{aligned} \tag{11}$$

where $\beta_l \in \beta_{LS}, \beta_r \in \beta_{RS}, h_l \in H_{LS}$ and $h_r \in H_{RS}$, $W_1 \in \mathbb{R}^{1 \times d}$ and $W_2 \in \mathbb{R}^{1 \times d}$ are weight matrices, $b_1 \in \mathbb{R}$ and $b_2 \in \mathbb{R}$ are biases, and $b_l \in \mathbb{R}$ and $b_r \in \mathbb{R}$ are hyperparameters. The final aspect weight β_A is expressed as an average of the aspect information captured by β_{LS} and β_{RS}. By eliminating the aspect information from these two vectors we obtain β_{LC} and β_{RC}, weights for the left and right context information, respectively. Concatenating all three parts (left context, aspect, right context) together gives us the bidirectional context attention weights $\beta \in \mathbb{R}^{1 \times N}$ for the whole sentence S.

The obtained bidirectional context attention weights are used to scale the importance of each word in the given sentence by taking into account the word order information, aspect information, correlation between each word and the aspect, and the past and future information. Each memory slice $m_{w_n} \in \mathbb{R}^d$ of the weighted memory $M_w = [m_{w_1}, ..., m_{w_N}]$ is constructed as follows:

$$m_{w_n} = \beta_{tiled} \odot e_n, \tag{12}$$

where $\beta_{tiled} \in \mathbb{R}^d$ is the element $\beta_n \in \mathbb{R}$ replicated d times, and $\odot$ represents the element-wise multiplication. The weighted memory M_w is then fed into the sentence-level content attention module.

3.2.3 Sentence-Level Content Attention Mechanism. We make use of the sentence-level content attention mechanism to explicitly capture the aspect information and the meaning of the whole sentence to ensure that the model differentiates between the important and less relevant factors given a specific aspect.

In order to incorporate aspect information into the computation of score c_n for each word s_n in the sentence S, we transform the aspect into a vector representation v_a by taking an average of all word embedding vectors in E_A. A sentence representation v_s is created in the same manner by taking the average of all word embedding vectors e_n in the sentence S (it has been introduced as an effective method in [2]):

$$v_a = \frac{1}{L}\sum_{l=1}^{L} e_l, \qquad v_s = \frac{1}{N}\sum_{n=1}^{N} e_n. \tag{13}$$

The computation of scores c_n is given below:

$$c_n = W_3 \tanh(W_4 m_{w_n} + W_5 v_a + W_6 v_s + b_3), \tag{14}$$

where $m_{w_n} \in \mathbb{R}^d$ is the weighted memory slice of the word s_n, $v_a \in \mathbb{R}^d$ and $v_s \in \mathbb{R}^d$ are the aspect and sentence representations, $W_3 \in \mathbb{R}^{1\times m}$, $W_4 \in \mathbb{R}^{m\times d}$, $W_5 \in \mathbb{R}^{m\times d}$, $W_6 \in \mathbb{R}^{m\times d}$ are weight matrices and $b_3 \in \mathbb{R}^m$ is a bias vector.

Attention weights $\alpha = [\alpha_1, ..., \alpha_N]^T \in \mathbb{R}^N$ are constructed by applying *softmax* function on the each score in the word score vector $c = [c_1, ..., c_N]^T \in \mathbb{R}^N$. Each attention weight α_n is obtained as follows:

$$\alpha_n = exp(c_n) / \sum_{j=1}^{N} exp(c_j). \tag{15}$$

Then a weighted embedding sentence vector $v_{we} \in \mathbb{R}^d$ is calculated by:

$$v_{we} = M_w \alpha, \tag{16}$$

where $\alpha = [\alpha_1, ..., \alpha_N]^T \in \mathbb{R}^N$, $M_w \in \mathbb{R}^{d\times N}$.

3.2.4 Classification Module. In order to increase the classifier's ability to generalise and predict the correct sentiment value, we introduce the classification module. We explicitly define the relationship between the sentence representation v_s and the weighted embedding sentence vector v_{we}, as well as the explicit relationship between the aspect representation v_a and the weighted embedding sentence vector v_{we}. All relations are modelled using hyperbolic tangent activation functions. The obtained results are then combined into an output vector v_o.

$$\begin{aligned} v_{sw} &= \tanh(W_7 v_s + W_8 v_{we} + b_4), \\ v_{aw} &= \tanh(W_9 v_a + W_{10} v_{we} + b_5), \\ v_o &= \tanh(W_{11} v_{sw} + W_{12} v_{aw} + b_6), \end{aligned} \tag{17}$$

where $W_7 \in \mathbb{R}^{d\times d}$, $W_8 \in \mathbb{R}^{d\times d}$, $W_9 \in \mathbb{R}^{d\times d}$, $W_{10} \in \mathbb{R}^{d\times d}$, $W_{11} \in \mathbb{R}^{k\times d}$, and $W_{12} \in \mathbb{R}^{k\times d}$ are weight matrices, $b_4 \in \mathbb{R}^d$, $b_5 \in \mathbb{R}^d$, and $b_6 \in \mathbb{R}^k$ are bias vectors, $v_a \in \mathbb{R}^d$ and $v_s \in \mathbb{R}^d$ are the aspect and sentence representations, and $v_{we} \in \mathbb{R}^d$ is the weighted embedding sentence vector.

A linear layer is used to convert the output vector $v_o \in \mathbb{R}^k$ into a vector $v_L \in \mathbb{R}^{|C|}$:

$$v_L = W_{13} v_o + b_7, \tag{18}$$

where $|C|$ is the number of possible aspect polarity categories, $W_{13} \in \mathbb{R}^{|C|\times k}$ is a weight matrix and $b_7 \in \mathbb{R}^{|C|}$ is a bias vector.

Finally, the linear layer output v_L is fed into a *softmax* function to generate aspect's polarity probabilities $p \in \mathbb{R}^{|C|}$:

$$p = softmax(v_L). \tag{19}$$

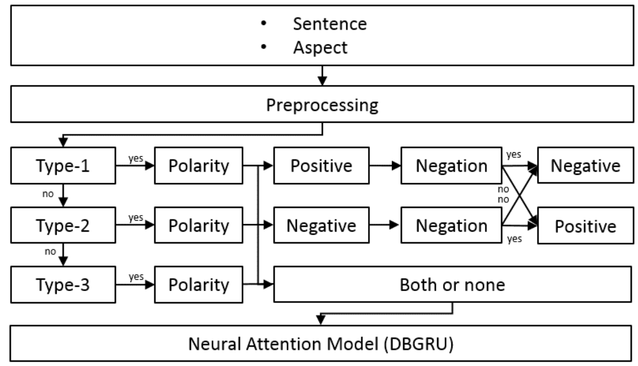

Figure 3: Stage 1. Graphical representation of the Lexicalised Domain Ontology classification algorithm.

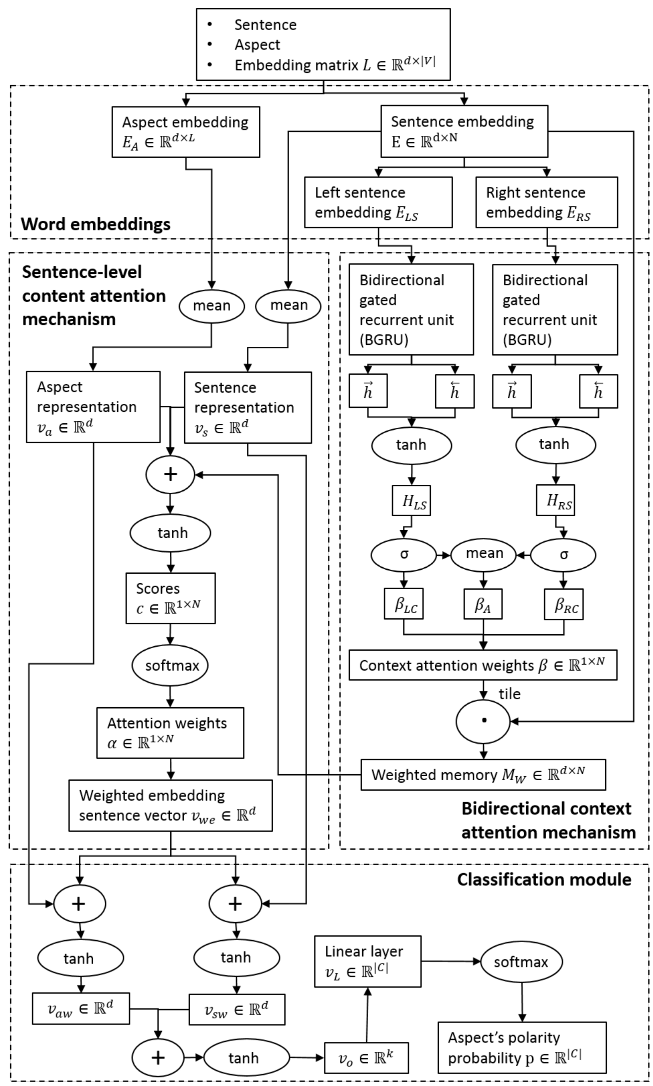

Figure 4: Stage 2. Graphical representation of the neural attention model (DBGRU).

3.2.5 Regularization and Loss Function. In order to handle the model's complexity and prevent over-fitting, we employ the dropout technique. We minimise the cross-entropy loss function given below:

$$loss = -\sum_{C}\sum_{S} y_{c,s}\ln(p_{c,s}), \tag{20}$$

where C is the set of polarity categories, S are the training examples, $p_{c,s} \in [0,1]$ is the estimated probability that a given aspect in a sentence s belongs to a category c, and $y_{c,s} \in \mathbb{B}$ is the true probability that the aspect in the sentence s is in the category c.

4 DATA

The data used in this research was made available for SemEval 2016 Task 5 [24]. SemEval (Semantic Evaluation) is a yearly computational semantic analysis competition organised by SIGLEX, the Special Interest Group on the Lexicon of the Association for Computational Linguistics. We make use of "Subtask 1 Restaurant domain English Training data" and "Subtask 1 Restaurant domain English Gold Annotations data" (which consist of Web restaurant reviews) as the training and test data sets, respectively. Every review is split into sentences which contains information about its attributes: *target* (the aspect itself), *category* (category of the aspect), *polarity* (sentiment value of the aspect), *from* and *to* (character-wise aspect position indication in a given sentence). The data snippet is given in Figure 5.

```
<sentence id="en_BlueRibbonSushi_478218345:2">
 <text>It has great sushi and excellent service.</text>
 <Opinions>
   <Opinion target="sushi" category="FOOD#QUALITY"
          polarity="positive" from="13" to="18"/>
   <Opinion target="service" category="SERVICE#GENERAL"
          polarity="positive" from="35" to="42"/>
 </Opinions>
</sentence>
```

Figure 5: Example data snippet.

The data is very skewed containing 70.19% and 74.34% positive labels in the train and test sets, respectively (see Table 1). Moreover, the data can be grouped in 12 aspect categories. Proportions of these classes are shown in the pie charts in Figure 6 for the train and test data sets, respectively.

Table 1: Polarity distribution in train and test data sets

	Positive		Neutral		Negative		Total	
	Freq.	%	Freq.	%	Freq.	%	Freq.	%
Train data	1314	70.19	71	3.79	487	26.01	1872	100
Test data	478	74.34	32	4.98	133	20.68	643	100

All words are converted to lowercase, """, "'" and "&" are replaced with a double quote symbol ("), an apostrophe ('), and the word *and*, respectively. All punctuation signs, numbers and tabulation are removed as well. Furthermore, reviews with several targets are treated as different entities. Sentences containing implicit aspects (*target="NULL"*) are not considered in this paper and are left for further investigation without significantly changing the remaining data set. Sentences without *Opinions* are also excluded, as they do not provide any

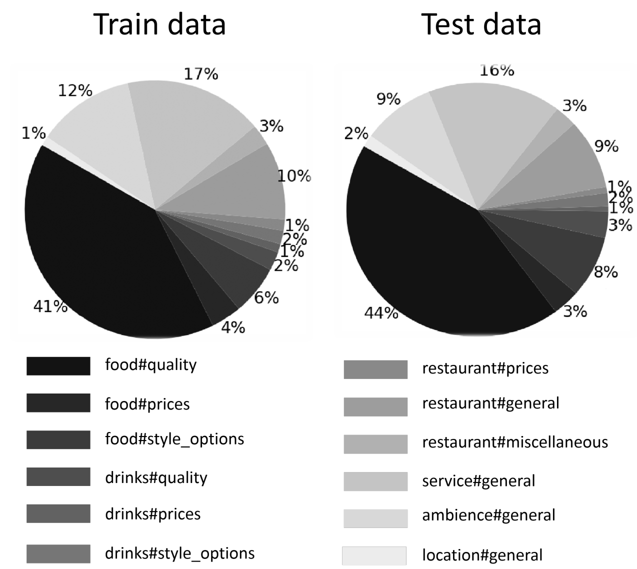

Figure 6: Aspect categories in train and test data sets.

aspect information. We make use of GloVe word embedding vectors [23]: 1.9 million vocabulary size with 300-dimensional vectors gives a solid base for our research. However, we eliminate words without embedding vectors from train and test data sets. Pure train and validation data sets are created by applying the stratified random sampling, and by splitting the train data set into 75/25 proportions. Finally, the tokenization and lemmatization are applied in order to infer the aspect types introduced in Subsec. 3.1.

5 PERFORMANCE EVALUATION

Initially, we evaluate the lexicalised domain ontology classification algorithm (Ont) proposed by [29]. As originally presented, the model is able to differentiate only *Positive* and *Negative* sentiment values. In case both or none of the labels are predicted for a given sentiment, the major polarity class is assigned (here, *Positive*).

Several advanced neural network models used as benchmarks in this evaluation have been introduced in [19]. Containing different attention mechanisms, such as the content attention (BaseA), sentence-level content attention (BaseB), sentence-level position attention (BaseC), or sentence-level context attention (CABASC), they provide the strong background for our evaluation. The last three baseline models employ the context attention mechanism [19] combined with unidirectional long short-term memory module (CTX-LSTM), bidirectional long short-term memory module (CTX-BLSTM), or bidirectional gated recurrent unit (CTX-BGRU). Train and test accuracy obtained by these models, as well as DBGRU and ALDONA are given in Table 2. Accuracy is chosen as a common performance measure for SemEval tasks.

As expected, classification accuracy increases with complexity of a model and can be explained by the larger number of neurons involved in the optimisation process. Furthermore, the context attention modules incorporating neural sequences ((B)RNN) tend to show better performance compared to other attention modules.

Table 2: Train and test classification accuracy obtained by the models and expressed in percentages

	Train accuracy	Test accuracy
Ont	74.95	78.38
BaseA	77.40	83.05
BaseB	83.76	84.60
BaseC	84.29	85.23
CABASC	79.65	84.45
CTX-LSTM	82.43	84.60
CTX-BLSTM	84.99	84.60
CTX-BGRU	86.65	85.38
DBGRU	89.58	86.00
ALDONA	90.17	86.31

The hyperparameters, namely, $b_r = b_l = 0.5$, $d = 300$, $learning\_rate = 0.001$, and normally ($N(0, 0.0025)$) initialised weight matrices used for all models were inherited from [19]. Although mentioned in [19], $dropout\_probability$ was not explicitly defined. Based on ALDONA's results shown on the validation data set, we set it to 0.3. DBGRU and ALDONA specific hyperparameters are set based on the grid search: $m = 300$ and $k = 150$. Contrary to the used Stochastic Gradient Descent in [19], for efficiency reasons we optimise all models using the Minibatch Gradient Descent algorithm with $batch\_size = 128$.

The choice of $m = 300$ can be explained by the fact that m directly determines the complexity of word scores c_n which are then transformed to the attention weights α (Eq. 15). Thus, having not enough hidden neurons (small m) cannot capture all the relevant information. On the other hand, excessively large m produces too complex relations. The relatively small number of neurons $k = 150$ used to determine the output vector v_o in Eq. 17 indicates that the main information has been extracted by the previous layers and not many new relations can be derived in the last linear layer. One can note that this is different than [19], where $k = m = 300$, as the authors did not allow a different dimension in this last layer for no clear reasons.

6 CONCLUSION

In this article, we presented the 2-stage hybrid model for sentence-level aspect-based sentiment classification called ALDONA. ALDONA is constructed to extract the field-specific information by means of the lexicalised domain ontology, as well as to model statistical relations captured by the neural attention model. Evaluation on the benchmark data set and comparison with other advanced models have stressed the contribution made by our research as well as created a new threshold for further investigation.

Future studies could concentrate on replacing the manually created lexicalised domain ontology by the semi- or fully automatic algorithm. The potential positive effect on the classification accuracy would be obtained due to broader test coverage by the ontology. Another potential research topic involves classification of implicit aspects which have not been considered in this work. For this we plan to use the concept of most similar words to the current aspect as introduced in [10].

REFERENCES

[1] Rana Abaalkhail, Benjamin Guthier, Rajwa Alharthi, and Abdulmotaleb Saddik. 2009. Survey on Ontologies for Affective States and Their Influences. *Semantic Web* 0 (2009), 1–19.

[2] Yossi Adi, Einat Kermany, Yonatan Belinkov, Ofer Lavi, and Yoav Goldberg. 2017. Fine-Grained Analysis of Sentence Embeddings using Auxilary Prediction Tasks. In *Proceedings of the 5th International Conference on Learning Representations (ICLR)*.

[3] Harith Alani, Sanghee Kim, David E. Millard, Mark J. Weal, Wendy Hall, Paul H. Lewis, and Nigel R Shadbolt. 2003. Automatic Ontology-Based Knowledge Extraction from Web Documents. *IEEE Intelligent Systems* 18 (2003), 14–21.

[4] Erik Cambria. 2016. Affective Computing and Sentiment Analysis. *IEEE Intelligent Systems* 31 (2016), 102–107.

[5] Peng Chen, Zhongqian Sun, Lidong Bing, and Wei Yang. 2017. Recurrent Attention Network on Memory for Aspect Sentiment Analysis. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing (EMNLP 2017)*. ACL, 452–461.

[6] Zhiyuan Chen, Arjun Mukherjee, and Bing Liu. 2014. Aspect Extraction with Automated Prior Knowledge Learning. In *52nd Annual Conference of the Association for Computational Linguistics (ACL 2014)*, Vol. 1. ACL, 347–358.

[7] Kyunghyun Cho, Bart van Merriënboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. 2014. Learning Phrase Representations using RNN Encoder–Decoder for Statistical Machine Translation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Association for Computational Linguistics, 1724–1734.

[8] Anni Coden, Dan Gruhl, Neal Lewis, Pablo N. Mendes, Meena Nagarajan, Cartic Ramakrishnan, and Steve Welch. 2014. Semantic Lexicon Expansion for Concept-Based Aspect-Aware Sentiment Analysis. In *Semantic Web Evaluation Challenge (CCIS)*, Vol. 475. Springer International Publishing, 34–40.

[9] Sanjiv R. Das and Mike Y. Chen. 2007. Yahoo! for Amazon: Sentiment Extraction from Small Talk on the Web. *Management Science* 53, 9 (2007), 1375–1388.

[10] Nikoleta Dosoula, Roel Griep, Rick den Ridder, Rick Slangen, Ruud van Luijk, Kim Schouten, and Flavius Frăsincar. 2016. Sentiment Analysis of Multiple Implicit Features per Sentence in Consumer Review Data. In *Databases and Information Systems IX - Selected Papers from the Twelfth International Baltic Conference, DB&IS 2016 (Frontiers in Artificial Intelligence and Applications)*, Vol. 291. IOS Press, 241–254.

[11] Thomas Robert Gruber. 1995. Toward Principles for the Design of Ontologies Used for Knowledge Sharing. *International Journal of Human-Computer Studies* 43, 5–6 (1995), 907–928.

[12] Zellig Sabbettai Harris. 1954. Distributional Structure. *WORD* 10, 2–3 (1954), 146–162.

[13] Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long Short-Term Memory. *Neural Computation* 9, 8 (1997), 1735–1780.

[14] Alexander Hogenboom, Paul van Iterson, Bas Heerschop, Flavius Frăsincar, and Uzay Kaymak. 2011. Determining negation scope and strength in sentiment analysis. In *IEEE International Conference on Systems, Man, and Cybernetics 2011 (SMC 2011)*. IEEE SMC Society, 2589–2594.

[15] Soufian Jebbara and Philipp Cimiano. 2016. Aspect-Based Sentiment Analysis Using a Two-Step Neural Network Architecture. In *Semantic Web Challenges. Third SemWebEval Challenge at ESWC 2016. Revised Selected Papers*, Vol. 641. Springer International Publishing, 153–170.

[16] Yoon Kim, Carl Denton, Luong Hoang, and Alexander M. Rush. 2017. Structured Attention Networks. In *Proceedings of the 5th International Conference on Learning Representations (ICLR 2017)*, Vol. arXiv/1702.00887.

[17] Quoc Le and Tomas Mikolov. 2014. Distributed Representations of Sentences and Documents. In *Proceedings of the 31st International Conference on Machine Learning (ICML 2014)*, Vol. 32. JMLR: W&CP, 1188–1196.

[18] Bing Liu. 2015. *Sentiment Analysis: Mining Opinions, Sentiments, and Emotions*. Cambridge University Press.

[19] Qiao Liu, Haibin Zhang, Yifu Zeng, Ziqi Huang, and Zufeng Wu. 2018. Content Attention Model for Aspect Based Sentiment Analysis. In *Proceedings of the 2018 World Wide Web Conference (WWW 2018)*. IW3C2, 1023–1032.

[20] Ahmet Vehbi Olgaç and Bekir Karlik. 2011. Performance Analysis of Various Activation Functions in Generalized MLP Architectures of Neural Networks. *International Journal of Artificial Intelligence And Expert Systems* 1 (2011), 111–122.

[21] Georgios Paltoglou and Mike Thelwall. 2013. More than Bag-of-Words: Sentence-based Document Representation for Sentiment Analysis. In *Proceedings of Recent Advances in Natural Language Processing (RANLP 2013)*. ACL, 546–552.

[22] Bo Pang, Lillian Lee, and Shivakumar Vaithyanathan. 2002. Thumbs up? Sentiment Classification using Machine Learning Techniques. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP 2002)*, Vol. 10. ACL, 79–86.

[23] Jeffrey Pennington, Richard Socher, and Christopher D. Manning. 2014. GloVe: Global Vectors for Word Representation. In *Empirical Methods in Natural Language Processing (EMNLP 2014)*. ACL, 1532–1543.

[24] Maria Pontiki, Dimitris Galanis, Haris Papageorgiou, Ion Androutsopoulos, Suresh Manandhar, Mohammad AL-Smadi, Mahmoud Al-Ayyoub, Yanyan Zhao, Bing Qin, Orphee De Clercq, Veronique Hoste, Marianna Apidianaki, Xavier Tannier, Natalia Loukachevitch, Evgeniy Kotelnikov, Núria Bel, Salud María Jiménez-Zafra, and Gülşen Eryiğit. 2016. SemEval-2016 Task 5: Aspect Based Sentiment Analysis. In *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval 2016)*. ACL, 19–30.

[25] Maria Pontiki, Dimitris Galanis, John Pavlopoulos, Harris Papageorgiou, Ion Androutsopoulos, and Suresh Manandhar. 2014. SemEval-2014 Task 4: Aspect Based Sentiment Analysis. In *Proceedings of the 8th International Workshop on Semantic Evaluation (SemEval 2014)*. ACL, 27–35.

[26] Soujanya Poria, Erik Cambria, and Alexander Gelbukh. 2016. Aspect Extraction for Opinion Mining with a Deep Convolutional Neural Network. *Knowledge-Based Systems* 108, 15 (2016), 42–49.

[27] Sebastian Ruder, Parsa Ghaffari, and John G. Breslin. 2016. A Hierarchical Model of Reviews for Aspect-based Sentiment Analysis. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing (EMNLP 2016)*. ACL, 999–1005.

[28] Kim Schouten and Flavius Frăsincar. 2016. Survey on Aspect-Level Sentiment Analysis. *IEEE Transactions on Knowledge and Data Engineering* 28, 3 (2016), 813–830.

[29] Kim Schouten and Flavius Frăsincar. 2018. Ontology-Driven Sentiment Analysis of Product and Service Aspects. In *15th Extended Semantic Web Conference (ESWC 2018) (LNCS)*, Vol. 10360. Springer International Publishing, 608–623.

[30] Kim Schouten, Flavius Frăsincar, and Franciska de Jong. 2017. Ontology-Enhanced Aspect-Based Sentiment Analysis. In *17th International Conference on Web Engineering (ICWE 2017) (LNCS)*, Vol. 10360. Springer International Publishing, 302–320.

[31] Duyu Tang, Bing Qin, Xiaocheng Feng, and Ting Liu. 2016. Effective LSTMs for Target-Dependent Sentiment Classification. In *Proceedings of the 26th International Conference on Computational Linguistics (COLING 2016)*. ACL, 3298–3307.

[32] Duyu Tang, Bing Qin, and Ting Liu. 2016. Aspect Level Sentiment Classification with Deep Memory Network. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing (EMNLP 2016)*. ACL, 214–224.

[33] Meishan Zhang, Yue Zhang, and Duy-Tin Vo. 2016. Gated Neural Networks for Targeted Sentiment Analysis. In *Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence (AAAI 2016)*. AAAI Press, 3087–3093.