Journal of Big Data

RESEARCH Open Access

Toward multi-label sentiment analysis: a transfer learning based approach

Jie Tao[1*] and Xing Fang[2]

*Correspondence: jtao@fairfield.edu
[1] Dolan School of Business, Fairfield University, 1073 N Benson Rd, Fairfield, CT, USA
Full list of author information is available at the end of the article

Abstract

Sentiment analysis is recognized as one of the most important sub-areas in Natural Language Processing (NLP) research, where understanding implicit or explicit sentiments expressed in social media contents is valuable to customers, business owners, and other stakeholders. Researchers have recognized that the generic sentiments extracted from the textual contents are inadequate, thus, Aspect Based Sentiment Analysis (ABSA) was coined to capture aspect sentiments expressed toward specific *review aspects*. Existing ABSA methods not only treat the analytical problem as single-label classification that requires a fairly large amount of labelled data for model training purposes, but also underestimate the *entity aspects* that are independent of certain sentiments. In this study, we propose a transfer learning based approach tackling the aforementioned shortcomings of existing ABSA methods. Firstly, the proposed approach extends the ABSA methods with multi-label classification capabilities. Secondly, we propose an advanced sentiment analysis method, namely Aspect Enhanced Sentiment Analysis (AESA) to classify text into sentiment classes with consideration of the entity aspects. Thirdly, we extend two state-of-the-art transfer learning models as the analytical vehicles of multi-label ABSA and AESA tasks. We design an experiment that includes data from different domains to extensively evaluate the proposed approach. The empirical results undoubtedly exhibit that the proposed approach outperform all the baseline approaches.

Keywords: Transfer learning, Multi-label classification, Sentiment analysis, Natural language processing, Deep learning

Introduction

With a great amount of online communities generating social media contents constantly as a high speed, understanding sentiments in social media contents is valuable for customers, business owners, and other stakekholders. Sentiment analysis has become the most crucial application in the field of Natural Language Processing (NLP) [1–4]. Particularly, understanding explicit or implicit sentiments expressed in social media contents is one of the most popular sub-areas in sentiment analysis. Performing sentiment analysis toward social media contents is definitely a big data analytics task. Early studies on sentiment analysis classify texts in a certain linguistic unit as positive, negative, or neutral—assuming a sentence is a self-contained unit in terms of expressing sentiments. Recent studies extend traditional sentiment analysis methods by assuming that sentiments are expressed toward specific aspects (e.g., features/characteristics of products/services). This type of analysis is termed as the Aspect Based Sentiment Analysis (ABSA), which employs supervised machine learning techniques as the classification tools [5]. Taking the sentence, "`I enjoyed this restaurant because of the chic atmosphere`", as an example. The positive sentiment (i.e., *enjoyed, chic*) is expressed toward a certain aspect (e.g., *atmosphere*).

Although, due to the design of the ABSA methods, there are three major shortcomings. Firstly, they can only classify instances (e.g., sentences) to one of the pre-defined classes (e.g., aspect-sentiment pairs) [6]. However, consumers may express sentiments toward one or more aspects in the social media contents. For instance, a sentence "`the food quality is decent, but the price is very steep`" contains two aspect-sentiment pairs: "food, positive" and "price, negative". Thus, ABSA should sometimes be treated as a multi-label classification problem, whereas an instance is classified to a subset of the predefined classes. Secondly, people sometimes on social media express sentiments to the entity itself (e.g., restaurants, movies), rather than a specific aspect of the entity (e.g., food of a restaurant, or cast of a movie). For example, in a sentence "`I like [movie name] as a horror movie`", the reviewer expresses positive sentiment toward the movie, with the movie being referred to as a horror movie. The ABSA methods typically cannot handle this type of analysis, because unlike in the context of ABSA (where sentiments are expressed toward certain aspects), the aspects and sentiments are independent of one another in this context. Last but not the least, due to the supervised nature of the classification techniques, a relatively large set of labelled data is required to train the machine learning models to achieve satisfactory results. However, quality labelled data is scarce in a variety of domains.

In order to address the aforementioned limitations, this study makes several research contributions toward the domains of sentiment analysis and transfer learning. Firstly, we extend recent ABSA methods by introducing the multi-label ABSA method. Secondly, we propose a new sentiment analysis method, which classifies textual contents into predefined sentiment classes at a comprehensive linguistic level (e.g., documents), with considerations of the entity aspects. The entity aspects, unlike the aspects used in ABSA, are the aspects describing the entity as a whole (e.g., types of wines, genres of movies). We term this type of sentiment analysis as Aspect Enhanced Sentiment Analysis (AESA). Thirdly, we extend two recently released transfer learning models, namely BERT and XLNet, by making them as the analytical vehicles for multi-label ABSA and AESA classification problems. To the best of our knowledge, this is the first study using transfer learning based approaches for multi-label ABSA and AESA tasks. We extensively evaluate the proposed approach using data from several different domains. The empirical results from the experiment show the superiority of our approach against a variety of baseline approaches. Additionally, this study also makes following impacts toward practitioners. Firstly, we propose an end-to-end solution for the multi-label sentiment classification problem, which is able to yield quality results without additional preprocessing of the data or the transformation of the problem. Secondly, our approach takes social media contents at different linguistic levels as the raw input, and yields quality ABSA/AESA classification results, with only limited number of epochs in model training. Lastly, we minimize human biases introduced in the analysis and relax the heavy requirements of labelled data, by using the transfer learning models, which rely on the pre-trained deep learning models. Our proposed approach can be used to develop automated tagging schema that can be used in different domains, or other big data analytics applications (e.g., automated query machines).

Related work

Aspect based sentiment analysis

Sentiment analysis refers to the process of extracting explicit or implicit polarity of opinions expressed in textual data (e.g., social media including online consumer reviews [1, 7]). Sentiment analysis has been used for information seeking and demand addressing needs on the consumer side, whereas for business owners and other stakeholders for operational decision making (e.g., *branding, preventive/reversal actions*) [5]. Traditional sentiment analysis focus on extracting opinion polarities at a coarse level, which cannot fully satisfy aforementioned purposes. Sentiments are normally domain dependent (e.g. delicious indicates positive sentiment in the food domain, where it does not indicate any sentiment in the laptop domain). Additionally, consumers tend to express their sentiment regarding/associated to specific features or *aspects* of different goods or services. Extracting sentiments with consideration of the associated aspects is termed as *Aspect Based Sentiment Analysis* (ABSA) [8]. ABSA brings additional values to different audiences: users can better align their preferences based on sentiments and the associated aspects (e.g., a customer prefers the *superior service* over the *chic ambience* when selecting a restaurant); for business owners, sentiments toward different aspects can assist them making finer grained decisions of business operations (e.g. a restaurant owner focuses on the *food quality* rather than the *location* based on the customers' collective preference). Compared to traditional sentiment analysis methods, it is evidential that ABSA provides additional values to the stakeholders.

There are two major groups of ABSA methods, namely *lexicon based* and *machine learning based* [5]. For lexicon based methods, both sentiment and domain-specific aspect lexicons are required in the analyses. For example, a recent analysis relied on sentiment lexicons (i.e., *dictionaries*) containing Chinese words to extract aspect based sentiments from micro-blogs [9]. Machine learning based methods utilize unsupervised and/or supervised learning techniques to extract aspects and sentiments from textual contents. For example, Siering et al. [6] utilized text statistics and linguistic information to extract a variety of aspects from airline company reviews, then they trained the supervised classifiers to assign sentiments to different aspects, for the purpose of explaining and providing recommendations to (potential) customers. Additionally, Akhtar et al. [7] designed an optimizer-based feature selection method to extract aspects terms from texts, then an ensemble machine learning model was trained to classify sentiments toward extracted aspects.

Given the state-of-the-art ABSA methods, several limitations can be identified within the machine learning based methods:

- Although lexicon based methods do not require any training, they suffer from inferior performances with respect to accuracy due to the limited coverage of the lexicons. In addition, lexicon based methods are not applicable to domains that the definitive lexicons are non-existent.

- In order to use supervised machine learning techniques in ABSA, a fairly large labelled training dataset is required. Obtaining such dataset may be time consuming and labor intensive. In addition, unsupervised machine learning techniques do not require labelled training data, yet it is difficult to guarantee satisfactory performances from them.
- A majority of machine learning based ABSA methods follow a two-step fashion: treating classification of aspects and sentiments as separate steps. It is more efficient to treat both steps in a holistic fashion, to avoid separating the explicit or implicit logical/semantic connects between them.
- Several prior related studies extract aspects and sentiments at word level [6, 9]; while other studies rely on linguistic patterns among words [5, 10]. However, it is beneficial to treat sentences as sequences (of words) to maintain the semantic meanings in them.
- In some domains, opinion polarities are not directly expressed toward aspects of products/services, rather than the entities being reviewed (e.g., restaurants, movies). However, these sentiments may be expressed toward the entities, along with certain aspects in consideration (e.g. a certain type of restaurants, or a certain genre of movies). Existing traditional sentiment analysis and ABSA methods have overlooked this type of analyses largely.

Multi-label classification

As a variety sources of data (e.g., text, images, videos) being used in the field of machine learning, new applications have been designed to learn from these data sources. Such applications include text classification, and semantic annotation of images and videos. Traditional single-label based machine learning techniques have been proven inadequate in these applications. For example, an image may contain multiple objects to be detected, or a text excerpt may discuss multiple topics in a document. Thus, multi-label classification techniques, which classify instances to a subset of pre-defined classes, have attracted increasing attention recently [9]. Existing multi-label classification methods can be grouped into three categories, namely problem transformation based methods, algorithm adaptation based methods, and ensemble model based methods.

As suggested by the name, the problem transformation methods transform a multi-label classification problem into one or multiple single-label classification problem(s). In the training phase, multi-label training data are transformed into single-label data, on which a single-label classifier, where a plethora of traditional machine learning techniques can be applied to, is trained. In the testing phase, multiple single-label predictions are made on each instance in the testing set. The simplest strategy in this category is the one-versus-rest (OR) strategy, which transforms a multi-label classification problem into multiple single-label classification problems, in which each instance in the dataset is classified into one label, or the rest of the labels in the set. A closely related strategy is Binary Relevance (BR) [11], in which a multi-label classification problem is transformed into an example belongs to one label or not. Building on the BR strategy, a new strategy namely Classifier Chains (CC) [12], which also classify whether an example belongs to one label or not, in a chain-like structure (where each link is a binary classifier, and the prediction of a subsequent classifier is dependent on the predictions of all predecessors). Compared to BR, CC can capture the interdependencies between each pair of labels. Another type of the problem transformation methods transform the label space (the space containing all labels), rather than the data. This type of problem transformation strategy is called Label Powerset (LP) [13], in which each example is classified to a power set (subset) in the label space. A powerset is a combination of multiple (inter-related) labels, and the combined powersets are set as new labels so that examples in the dataset can be classified to.

The second group of multi-label classification methods are algorithm adaptation methods. Unlike problem transformation methods that work as wrappers over traditional machine learning techniques, algorithm adaptation methods transform traditional machine learning techniques so that they are capable of handling multi-label classification problems. For example, k Nearest Neighbors (kNN), as a popular traditional machine learning technique, has been combined with the BR strategy, and results in an algorithm called BRkNN [14]. Similarly, Zhang and Zhou [15] proposed MLkNN by relying on the maximum a posteriori (MAP) principle on trained kNN models to determine the proper label set that a testing example belonged to. Neural Network is another type of popular machine learning techniques. Benites and Sapozhnikova [16] proposed a fuzzy Adaptive Resonance Associative Map (ARAM) adaptation to the neural networks so that the proposed algorithm is useful for multi-label classification problems.

The last group of multi-label classification methods are ensemble methods. In the field of machine learning, ensemble models typically refer to stack and/or combine different models together for better performances/results. In the context of multi-label classification, the ensemble methods are developed based on the aforementioned problem transformation and algorithm adaptation methods. The most well-known methods in this group include Random k-Labelsets (Rakel) [13] and Ensemble Classifier Chain (ECC) [17]. Rakel trains each base classifier based on a small random set of labels, and then train a single-label classifier to predict the powerset of each random subset. ECC treats CC as base classifiers, the final prediction is obtained by summing up the prediction by labels and then comparing the results to the threshold in order to select relevant labels.

In the context of multi-label sentiment analysis, there are a few limitations that can be identified from them:

- Traditional multi-label classification approaches require either additional processing of the data (i.e. calculating posteriori rules for ML-kNN) or transformation of the problem (e.g. OR/BR/CC transforms the classification problem), which may increase computational complexity and/or possibility of introducing human biases (e.g. sequence of classifiers in CC).
- In order to let users trust the (multi-label) classification results, the results need to be at a satisfactory level, with regards to different evaluation metrics. It is deemed necessary to search for a better performing approach that is able to outperform traditional multi-label classification approaches (e.g., [9, 17]).

Transfer learning

Transfer learning uses domain-specific data to fine tune the pre-trained deep learning models. The benefit of conducting transfer learning is twofold: The time spent in training is much less than the time used in training from scratch. In computer vision, it is a common practice to use transfer learning: Parameters of the fully connected layers of a pre-trained CNN are replaced with randomly initialized values. A fine-tuning process is then performed by updating the new values only, using backpropagation, while the parameters in the convolutional layers stay untouched [18, 19]. Transfer learning in NLP, however, has been shown as a somewhat difficult task. One early successful case involves fine-tuning the pre-trained word embeddings [20], has had a large impact in practice. Howard and Ruder [21] proposed a transfer learning method that fine-tunes a three-layered LSTM language model [22] for text classification. In the first step of their approach, sentences of a training set are used to finetune the parameters in the LSTM layers. The labels of the training set are then used in step two to update the parameters of the fully connected layers. This approach reuses the embeddings of the original language model. Radford et al. [23] used the Generative Pre-trained Transformer (OpenAI GPT) to achieve state-of-the-art results on many sentence-level tasks from the GLUE benchmark [24].

However, the aforementioned transfer learning models do not support multi-label classification natively. It is because the softmax function used in the output layers of the models only support single-label classification tasks. Essentially, the softmax function produces a probability distribution over all the classes, where only one class with the highest probability will be selected as the output. This limitation must be addressed since multi-label classification tasks require the predictions to have more than one classes.

Methodologies

Multi-label aspect based sentiment analysis

To address the first shortcoming of the ABSA methods, we propose a multi-label classification extension to the existing ABSA methods. Essentially, those existing ABSA methods classify any one example (i.e. a review, a sentence) in a dataset into one of the pre-defined aspect/sentiment pairs (i.e., class); while the multi-label ABSA method proposed in this study classifies one example into a set of aspect/sentiment pairs (i.e., a set of classes). The details of multi-label classification mechanism used in our multi-label ABSA approach are discussed in "Our approach" subsection below.

We report a few examples to illustrate the labelling mechanism of the proposed multi-label ABSA method, in Tables 1, 2. Specifically, the second review example describes the food positively, the service negatively without mentioning the rest of other aspects. The review is then labelled as "10000000001000000000" (see Table 2). As for the experiment, we use the reviews that at least cover one of the aspects (i.e., a review must have at least one class labelled as 1). In other words, they are the reviews where the classes cannot be all zeros.

Table 1 Examples of reviews and labels in multi-label ABSA

Reviews	Labels
The food was delicious service always came quickly with a joke or a smile and the portions are unbelievably HUGE	10000000100000000000
The food selection was fantastic but waiting over hour to be seated	10000000001000000000
Everything that I have eaten here has put me in a coma of ecstasy so please bring a designated driver to take you home	10000000000000000000
They also have a cheap lunch buffet with Pad Thai and other dishes one of my favorite dishes there are the Garlic Wings	01001000000000000000
I really like the atmosphere here	00000000000010000000
Let me preface the following review by saying that if I didn’t absolutely have a terrible experience I wouldn’t have said anything	00000000000000000010

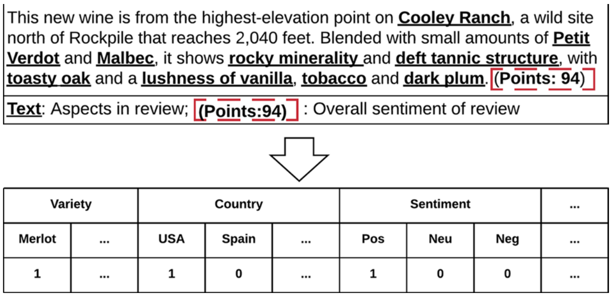
Multi-label aspect enhanced sentiment analysis

According to Do et al. [25], the study of sentiment analysis can be done at three different levels—document, sentence, and entity/aspect. Traditional sentiment analysis studies focusing on the document or sentence level assume that there is only one topic existing in the document/sentence, where the sentiment is expressed on. Thus, techniques like ABSA have been developed to bring sentiment analysis to a finer granularity: where sentiments are extracted along with different entities/aspects.

Despite the success of ABSA, in some scenarios, people are more interested in the overall sentiment expressed at the document (i.e., review) level, while also capturing all the aspects in the document. For example, when a potential patron reads an online user review of a movie or a restaurant, she concentrates on whether the review recommends the movie/restaurant or not that is expressed as the overall sentiment. In the meantime, the reader cares about different entity aspects being reviewed. That is she may only be interested in American restaurants (type), or horror movies (genre). In other words, the reader focus on the overall sentiment of an entity being reviewed, along with certain entity aspects. The key difference between this type of analysis and ABSA is that: in ABSA, sentiments are expressed toward certain aspects, whereas the analysis here attempts to *enhance* document-level sentiments with aspects discussed in the document. Essentially, the document-level sentiments and the aspects are independent of one another. We define this type of sentiment analysis as *Aspect Enhanced Sentiment Analysis* (AESA). We use an example (a wine review) in Fig. 1 to illustrate the proposed AESA method. It is worth noting that in Fig. 1, different aspects (e.g., *winery location, variety of wine*, and *taste*) of the entity (wine) are discussed (labelled in bold, underlined font), but no specific sentiments are expressed toward them. However, an overall positive sentiment (expressed as **94 points**) are reported with the review. As a result, the labels are processed into a binary vector as shown in the lower part of Fig. 1. Since there are multiple 1s in the vector (a review must contain discussions of certain aspect(s) and the overall sentiment), AESA is by nature a *multi-label* classification problem.

Table 2 An example of a label in multi-label ABSA

Aspects																			
Food				Price				Service				Ambient				Misc.			
Pos	Neu	Neg	Con	Pos	Neu	Neg	Con	Pos	Neu	Neg	Con	Pos	Neu	Neg	Con	Pos	Neu	Neg	Con
1	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0



This new wine is from the highest-elevation point on **Cooley Ranch**, a wild site north of Rockpile that reaches 2,040 feet. Blended with small amounts of **Petit Verdot** and **Malbec**, it shows **rocky minerality** and **deft tannic structure**, with **toasty oak** and a **lushness of vanilla**, **tobacco** and **dark plum**. (Points: 94)

Text: Aspects in review; (Points:94) : Overall sentiment of review

Variety		Country			Sentiment			...
Merlot	...	USA	Spain	...	Pos	Neu	Neg	...
1	...	1	0	...	1	0	0	...

Fig. 1 An example of Aspect Enhanced Sentiment Analysis

Our approach

Feature selection is one of the major challenges in machine learning. It is even more challenging in multi-label classification than single-label classification, since it should extract features representing all the aspects and the sentiments. Transfer learning uses pre-trained deep learning models, where the feature selection process has been naturally embedded due to the use of raw data.

The transfer learning models used in this study are BERT [26] and XLNet [27]. Both BERT and XLNet have been reported as the state-of-the-art approaches in NLP-related learning tasks. Both models are essentially based on an encoder–decoder network, namely a transformer [28]. The original transformer's encoder network uses a six-layered neural network, where each layer has two sub-layers: a multi-headed attention layer and a single-layered feed-forward network. Since the transformer was proposed for learning long time dependencies without using recurrent layers, it uses positional encoding in addition to word embedding for its input:

$$\begin{aligned} PE_{(pos,2i)} &= \sin\left(\frac{pos}{10,000^{2i/d_{\text{model}}}}\right) \\ PE_{(pos,2i+1)} &= \cos\left(\frac{pos}{10,000^{2i/d_{\text{model}}}}\right) \end{aligned} \tag{1}$$

where pos is the position of a certain word token, i is the dimension, d_{model} is the embedding dimension. For example, let us consider a sentence that has four words, $\{w_0, w_1, w_2, w_3\}$. Let $d_{\text{model}} = 4$ and $pos = 2$, the positional encoding of the third word, w_2, is computed as follows:

$$\left[\sin\left(\frac{2}{10,000^0}\right), \cos\left(\frac{2}{10,000^0}\right), \sin\left(\frac{2}{10,000^{\frac{2}{4}}}\right), \cos\left(\frac{2}{10,000^{\frac{2}{4}}}\right)\right] \tag{2}$$

A position-dependent signal is added to each word-embedding to incorporate the order of words. This addition not only avoids destroying the embedding information but also adds the vital position information.

Following the input, a number of self-attentions are then calculated in the multi-headed attention layer. The following shows the computation of one self-attention:

$$\text{softmax}\left(\frac{(\mathbf{X}\cdot\mathbf{W}^{\mathbf{Q}})\cdot(\mathbf{X}\cdot\mathbf{W}^{\mathbf{K}})^{\top}}{\sqrt{d_k}}\right)\cdot(\mathbf{X}\cdot\mathbf{W}^{\mathbf{V}}) \tag{3}$$

where $\mathbf{X}$ is an $n \times m$ matrix representing the embedding of one sentence that has n words; m is the embedding dimension that is set to 512; $\mathbf{W}^{\mathbf{Q}}$, $\mathbf{W}^{\mathbf{K}}$, and $\mathbf{W}^{\mathbf{V}}$ are three $m \times m$ matrices. In the original encoder, a total number of 8 attentions are calculated and then summarized into one, before passing through one single-layered perceptron. This attention mechanism, when used in generating encoded vectors, allows a current word to pay attention to the other words that either to its left or to its right. This is the reason that the transformer's encoder is referred to as a bidirectional model. In terms of the configurations, both models uses a 24-layered transformer encoder, where each layer consists of a 16-headed self-attention layer in tandem with a single-layered feed-forward network. Both models share the same embedding dimension of 1024.

Despite the similarities shared by BERT and XLNet, parameters of BERT are obtained through the masked language modeling, where the model is trained to predict some masked-out words in given sentences. For example, in the original sentence, `a cat sat on a mat`, the word `cat` will be replaced with a mask token, `[MASK]`, at a probability of 80%; the word will be replaced with a random word (e.g., `a dictionary sat on a mat`), at a probability of 10%; the sentence will not be changed at all 10% of the time. Instead of predicting the masked-out words, XLNet adopts a different language modeling approach, namely the permutation language modeling.

Given a sentence that has n words, there exist $n!$ different word permutations. XLNet samples a permutation a time and tries to predict the target words based on the words that are permuted prior to the target words. As a concrete example, let us consider the sentence `Tom Sawyer rolls the boat`, where `Tom Sawyer` are the target words. In this case, BERT has the objective function of:

$$J_{BERT} = \log(p(\text{Tom}|\text{rolls the boat})) + \log(p(\text{Sawyer}|\text{rolls the boat})) \tag{4}$$

Suppose that XLNet samples the permutation: `[rolls, the, boat, Sawyer, Tom]`. The objective function of XLNet is:

$$J_{XLNet} = \log(p(\text{Sawyer}|\text{rolls the boat})) + \log(p(\text{Tomer}|\text{Swayer rolls the boat})) \tag{5}$$

In general, XLNet is able to cover more dependencies by not masking out the target words, and the permutation language modeling allows the model to gather more information from all positions.

Technical novelties

In order to extend the existing transfer learning models so that they fit the multi-label classification nature of this study, we make the following three enhancements to them. Specifically, the technical novelties proposed in this study include:

- The enhancement to the output layers' activation function.
- The enhancement to the models' loss functions.
- The enhancement to the label encodings.

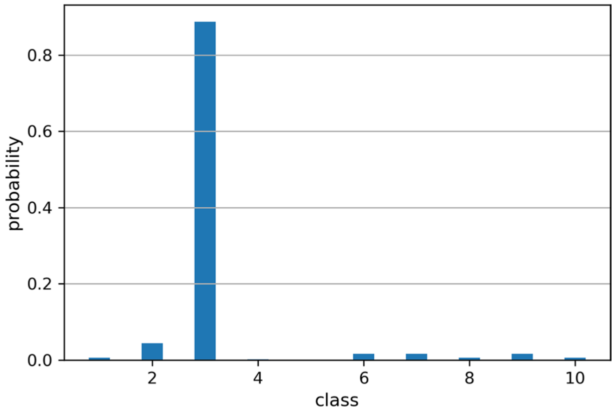


Fig. 2 A probability distribution based on the softmax activation

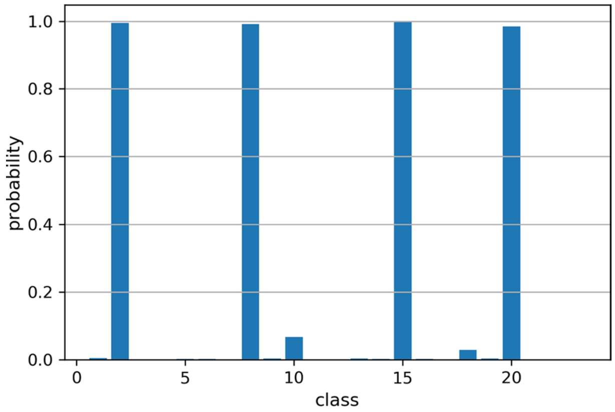


Fig. 3 A group of probabilities based on the sigmoid activation

The original setup of BERT and XLNet uses the softmax function as the output layers' activation to compute a conditional probability distribution (Fig. 2), where the sum of all the probabilities is equal to 1. The softmax function is the best choice as the activation function, if the neural network is trained to perform single-label classification: Only one class with the highest probability is selected as the classification result. In this case, the classes are mutually exclusive.

However, since both ABSA and AESA are essentially multi-label classification problems, where the classes are not mutually exclusive and the classification result should include multiple classes with the probabilities higher than a threshold (Fig. 3). The nature of the analysis requires us to find a different activation function that computes the stand alone probability for each neuron in the output layer, rather than computing a probability distribution over all the neurons. Another requirement is that the function should be continuously differentiable, hence, we implement the logistic sigmoid function, $f(x)$, as the output layers' activation function for the transfer learning models:

$$f(x) = \frac{1}{1 + \mathrm{e}^{-x}} \tag{6}$$

Originally, both transfer learning models use the categorical cross entropy as the loss function. The equation of computing the loss of the ith training instance, using the categorical cross entropy, is given as the following:

$$L_i = -y^i \cdot \log(prob(\hat{y} = y^i)) \tag{7}$$

where $prob(\hat{y} = y^i)$ is the conditional probability of the correct class, y^i, given by the softmax activation. Such loss function only measures the loss made by a single class that is when $\hat{y} = y^i$. When there are multiple classes need to be considered, we will need to implement a different loss function that measures the loss for all classes:

$$L_i = -\sum_{j=1}^{k}(y_j^i \cdot \log(f(d_j^i)) + (1 - y_j^i) \cdot \log(1 - f(d_j^i))) \tag{8}$$

where $d = \mathbf{W}^{\mathbf{C}} \cdot C^{\top}$ and $d \in \mathbb{R}^k$. Notice that the sigmoid function is element-wise, meaning that every element in d is used as a single input for the computation. $y_j^i = 1$, when j is the correct class; $y_j^i = 0$, when j is the incorrect class. Therefore, the loss is computed for all classes. As the training process minimizing the loss, it is equivalent to maximize $f(d_j^i)$, when j is the correct class.

The general loss function is defined as:

$$J = \frac{1}{m} \cdot \sum_{i=1}^{m} \left(L_i + \frac{\lambda}{2} \cdot ||\mathbf{W}||^2 \right) \tag{9}$$

where m is the batch size; $\frac{\lambda}{2} \cdot ||\mathbf{W}||^2$ is the L_2 regularization term to control overfitting, in which λ is a constant parameter and $||\mathbf{W}||^2$ represents the squared L_2 norm of the trainable parameters in both models.

Unlike in the single-label classification, where the labels are integers representing the correct classes, the labels in multi-label classification are in the multi-hot encoding format (Table 2). Since BERT and XLNet are not built with the considerations of multi-label classification, we are required to make another enhancement so that the multi-hot encoding can be used as the labels. The enhancement is termed as the reversible binary encoding.

Specifically, each multi-hot encoding is firstly transformed to an integer according to the positions of the 1s. For instance, the label in Table 2 would be transformed into $524,800 = 2^{19} + 2^9$. The integer can then be processed by the models' data pipelines. Right before the computation of the losses, the integer is then transformed back into its multi-hot encoding format. Additionally, we discover that this type of encoding also alleviates the bottleneck of GPU computation. Since the encoded labels use less system memory than their original format, when the labels are being transmitted to the GPU, the process runs faster than directly using the multi-hot encoding.

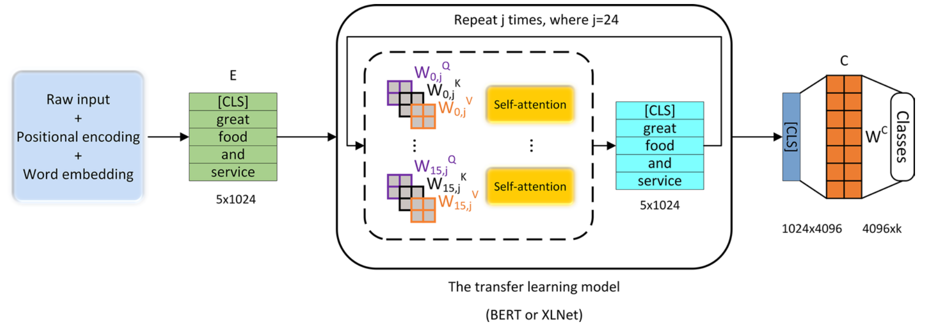


Fig. 4 The analytical pipeline of the proposed approach

The analytical pipeline

Figure 4 shows a concrete example of our analytical pipeline. Given a piece of raw input text, such as `great food and service`, the raw input will first be transformed into a stack of embeddings (E) that have a size of 5 × 1024. [CLS] is a special word embedding token indicating the start of a text sequence. The embeddings are then used as the input of a 24-layered transfer learning model, each of which computes 16 self-attentions in order to encode E. The final encoded output has the same exact size as the input (5 × 1024). The model then uses the encoded output of the token, [CLS], to map the encoded output onto the classes, where the probability distribution for all the classes is computed as:

$$P(k|E) = f(C^{\top} \cdot \mathbf{W}^{\mathbf{C}}) \tag{10}$$

where $C^{\top}$ is the activation of the final single feed forward layer that has a size of 1×4096; $\mathbf{W}^{\mathbf{C}}$ is a $4096 \times k$ matrix that is used to map C onto k classes.

Experiment setup and result analysis

Datasets

We collected three datasets from social media in different domains in order to evaluate our proposed approach. The first dataset [29], containing restaurant reviews, was collected from Yelp.com. The Yelp (YP) dataset was collected to demonstrate how the approach can be applied to the multi-label ABSA tasks. Due to the vast amount of restaurant review data (over 5 million reviews) available in the dataset, we selected the experiment data from several major metropolitan areas in U.S., including: Pittsburg, PA, Las Vegas, NV, and Charlotte, NC to ensure data volume and cultural/geographical coverage. We chose sentence as the unit of analysis in this experiment. Due to the complexity of manually labelling sentences with aspects and sentiments, we randomly sampled 10,000 sentences. Those sentences were labelled by three human coders, using a well-adopted schema in other related studies [30]. Five different aspects are defined for each review. Those aspects are: food (including drinks), price (including value), service, ambience (including location), and miscellaneous. There are four types of sentiments within each of the aspect: positive (pos), neutral (neu), negative (neg), and conflict (con). We define the conflict sentiment as both positive

Table 3 Aspects and sentiment of the WE dataset

WE		
Aspect		**Sentiment**
U.S.A., Australia France, Austria Chile, Argentina Portugal, Italy New Zealand,Spain (Country)	Chardonnay, Pinot Noir Cabernet Sauvignon Bordeaux-style Red Blend Red Blend,Zinfandel Sauvignon Blanc,Syrah Riesling,Merlot (Variety)	Positive Neutral Negative

Table 4 Aspects and sentiment of the RT dataset

RT	
Aspect	**Sentiment**
Action and Adventure, Animation Art House and International Anime and Manga, Classics Comedy, Documentary, Drama Cult Movies, Gay and Lesbian Faith and Spirituality, Horror Kids and Family Musical and Performing Arts Mystery and Suspense (Genre)	Positive Negative

and negative sentiments are expressed toward the same aspect in the same sentence. Each review has 20 classes associated to itself. As a result, we are able to obtain 8172 labelled sentences from the random sample. The labelling details of the YP dataset are illustrated in Tables 1 and 2.

The second dataset (Wine Reviews, WE) [31], containing descriptions and meta data of various types of wines, was scraped from winemag.com and made available on Kaggle. The third dataset (Rotten Tomatoes Movie Reviews, RT), was scraped from Rotten Tomatoes, a movie review website, and made available on Kaggle [32]. We prepared the WE and RT datasets to evaluate the performance of our proposed approach for the multi-label AESA tasks. In WE, the classes include the top 10 most popular countries, where the vineyards are located, the top 10 most popular wine varieties, and 3 sentiment polarities. Each example in the WE dataset is labelled with at least three 1s (one for country, variety, and sentiment, respectively). The WE dataset contains 80,638 wine reviews, after filtering samples without valid labels. In RT, the classes include the genre(s) and the sentiment polarities. It is worth noting that Rotten Tomatoes labels a movie review as “fresh” (positive) or “rotten” (negative), and we adopted these sentiment categories in our experiments. In addition, we ruled out 2 genres since there are too few reviews that support them. In total, 48,755 movie reviews are included the RT dataset under 21 genres and 2 sentiment categories. Each document in RT has a label that has at least two classes labelled as 1. In summary, the aspects and sentiment for both WE and RT are listed in Tables 3 and 4, respectively. Both the WE and RT datasets are multi-labelled. The labelling details of the WE dataset are illustrated in Fig. 1, whereas the labelling of the RT dataset is similar. For all three datasets (YP, WE, and RT), we randomly select 80% of the data as the training sets, and the remainder 20% as the testing sets. We report the classes of the WE and RT datasets in Tables 3 and 4, respectively.

Evaluation metrics

Based on the multi-label classification nature of this study, we select four evaluation metrics that are widely adopted in previous studies [9, 17]. The four evaluation metrics can be categorized into two groups, namely example based metrics and label based metrics. We do not select ranking based metrics, which is another group of multi-label classification models, since ranking of labels are not relevant in the scope of this study. We use two example based metrics, including the subset accuracy (*accuracy*) and the Hamming loss (*Hamming loss*), and two label based metrics, including the macro-average F1 score (*macro F1*) and micro-average F1 score (*micro F1*).

The subset accuracy is defined as the fraction of correct predictions, in which the predicted label set and the corresponding ground truth label set are exactly matched, in all the predictions (see Eq. (11)). The value of accuracy ranges within [0, 1]—the higher value indicates more superior performance.

$$\text{Accuracy} = \frac{\text{number of accurately classified examples}}{\text{total examples}} \tag{11}$$

The Hamming loss is defined as the fraction of incorrectly classified labels, normalized over the sample that it is reported from (see Eq. (12)). The loss value also ranges within [0, 1]; lower values indicate more accurate predictions.

$$\text{Hamming loss} = \frac{1}{N \cdot k} \sum_{i=1}^{N} \sum_{j=1}^{k} \mathbf{xor}(y_j^i, \hat{y}_j^i) \tag{12}$$

where N and k are the total number of testing instances and the number of classes, respectively; y_j^i is the label of the jth class in the ith testing instance; $\hat{y}_j^i$ is the predict of y_j^i.

The Micro average is computed as the average of micro-averaging precision and recall, which are defined as follows:

$$\text{precision}_{micro} = \frac{\sum_{c_i \in k} \text{TPs}(c_i)}{\sum_{c_i \in k} \text{TPs}(c_i) + \text{FPs}(c_i)} \tag{13}$$

$$\text{recall}_{micro} = \frac{\sum_{c_i \in k} \text{TPs}(c_i)}{\sum_{c_i \in k} \text{TPs}(c_i) + \text{FNs}(c_i)} \tag{14}$$

where c_i stands for a particular predicted class (e.g. an aspect-sentiment pair). Similarly, the Macro average is computed as the average of macro-average precision and recall:

$$\text{precision}_{macro} = \frac{1}{k} \cdot \sum_{c_i \in k} \frac{\text{TPs}(c_i)}{\text{TPs}(c_i) + \text{FPs}(c_i)} \tag{15}$$

$$\text{recall}_{macro} = \frac{1}{k} \cdot \sum_{c_i \in k} \frac{\text{TPs}(c_i)}{\text{TPs}(c_i) + \text{FNs}(c_i)} \tag{16}$$

With the precision and recall (macro- or micro-average) calculated, we can use following equation (see Eq. (17)) to calculate the Macro and Micro F1 scores, respectively.

$$\mathrm{F1}_{macro/micro} = 2 \cdot \frac{\mathrm{precision}_{macro/micro} \cdot \mathrm{recall}_{macro/micro}}{\mathrm{precision}_{macro/micro} + \mathrm{recall}_{macro/micro}} \tag{17}$$

The Macro F1 is the average of the harmonic mean of $\mathrm{precision}_{macro}$ and recall_{macro}, which are measured over each label in the overall label set. The Micro F1 is the harmonic mean of $\mathrm{precision}_{micro}$ and recall_{micro}, which are averaged over all the instances in the dataset and the label sets. The value of Macro/Micro F1 ranges in between 0 and 1—higher value of Macro/Micro F1 indicates better multi-label classification results. The Macro F1 gives equal weight to each aspect, whereas Micro F1 gives equal weight to each testing instance.

Experiment setup

Baseline deep learning models

According to [25, 33], we select three most widely applied deep learning models in sentiment analysis as the baseline deep learning models, to compare with the transfer learning models in performance. Those baseline models include: LSTM, Bi-LSTM (Bidirectional LSTM), CNN + LSTM. LSTM [34], as an enhanced recurrent neural network unit, is good at processing sequential information that has long-term dependencies, such as text sequences. Rather than processing the information based on only one direction, a Bi-LSTM [35–37] is able to process the sequences from both directions. A convolutional neural network (CNN) is commonly used in tandem with a LSTM in order to reduce the length of sequences, which can significantly facilitate the speed of training [38–40].

Baseline machine learning models

As discussed in "Related work" section, different methods can be used along with traditional machine learning models for multi-label classification problems. In this study, we select OR, LP, BR, and CC (*problem transformation*), RakelD (*ensemble*), MLARAM, MLkNN, BRkNNa, and BRkNNb [14] (*algorithm adaptation*) methods to enable the multi-label classification capability of base classifiers. Other multi-label classification methods are excluded in this article based on their inferior performances in a preliminary round. An additional point worth noting is that, we include Ensemble Classifier Chains (ECC) in our implementation of CC. Similar selection of multi-label classification methods can be found in prior related studies [9, 17].

Both linear and non-linear based classifiers were selected in the experiment, which are used with problem transformation and ensemble methods. For linear based models, we selected Linear Regression (LR) models and linear Support Vector Classifiers (SVC) models, while for non-linear based classifiers we select Support Vector Machines (SVM) with Stochastic Gradient Descent training (SGD) models and random forest (RF) models. It is worth noting we also tested that other base classifiers, including (multi-nominal, Gaussian) Naïve Bayes models, decision tree models, artificial neural networks, and SVM models with other kernel functions (e.g., radial basis) in a pilot study. We finally selected the four models (LR, SVC, SGD, RF) based on their overall superior performances, comparing to other base classifiers. Comparing to prior related studies [9, 17], we examined more base classifiers in the experiment.

As a result, we selected a total of 24 baseline machine learning models in this experiment, including: 16 problem transformation models (4 method × 4 base classifiers), 4 ensemble models (1 method × 4 base classifiers), and 4 algorithm adaptation models. We select these 24 baseline machine learning models to compare our proposed transfer learning based approach to the state-of-the-art multi-label classification models.

Model hyper-parameters

To select the best hyper-parameters for the baseline deep learning models, we conducted a manual grid search over a series of combinations. We tested a number of different layer sizes, $\{32, 64, 128, 256, 512\}$ together with different number of layers, $\{1, 2, 3, 4, 5\}$ for the LSTM and BiLSTM layers. Based on the results of the testing, we selected the best combinations that can lead our models to the highest accuracy.

Hyper-parameters of the baseline models are shown in Table 5. Both the LSTM and Bi-LSTM models are set to be two layers, with the first layer has a size of 128 and the second layer has a size of 256. The CNN + LSTM model has two convolutional layers, where each of those has 32 filters with five strides, followed by a single-layered LSTM that has a size of 64.

Table 6 shows the hyper-parameters' setup of both transfer learning models. The number of epochs controls the total number of times the models learning on an entire training set; the batch size is the number of documents used in each training batch; the maximum length allows the maximum number of words in one document to be 128; the learning rate is used in the optimization algorithm for updating the parameters in both models.

We follow the recommendations from prior related studies [9, 17] with respect to hyper-parameter settings, where applicable. Table 7 reports the hyper-parameters of three base classifiers used in the baseline machine learning models. No specific hyper-parameter is set for the SVC model. Additionally, for the RakelD based models, we use $min(2 \times C, 100)$ models, where C is the number of classes is the respective dataset.

Table 5 The setup of the baseline deep learning models

Models	Layers	Layer sizes
LSTM	2	128 + 256
Bi-LSTM	2	128 + 256
CNN + LSTM	2 (CNN), 1 (LSTM)	32, 5 (CNN) 64 (LSTM)

Table 6 The setup of the transfer learning models

	BERT			XLNet		
	Yelp	Wine	Movie	Yelp	Wine	Movie
Epochs	27	40	10	9	14	8
Batch size	32			16		
Maximum length	128			128		
Learning rate	2×10^{-5}			2×10^{-5}		

Table 7 The setup of the baseline machine learning models

Base classifier	Hyper-parameters
SGD	Loss = 'hinge', penalty = 'l2', $\alpha = \frac{1}{training\_size \times 5}$, maximal iteration = 20, tolerance = 1×10^{-3}
LR	Solver = 'lbfgs'
RF	Num of trees = 100, maximal depth = 3

Experiment results

Result analysis of multi-label ABSA

The result of multi-label ABSA is shown in Table 8, where our proposed transfer learning models indeed present the most superior performances: XLNet yields the best subset accuracy of 66.61%, followed by BERT with its subset accuracy being 61.65%; the Hamming losses of BERT and XLNet are also among the lowest, whereas

Table 8 Experiment results of Yelp reviews

Model	Accuracy (%)	Hamming loss	Macro F1	Micro F1
Proposed models				
BERT	61.65	0.032	0.48	0.70
XLNet	66.61	0.027	0.56	0.77
Baseline deep learning models				
LSTM	35.66	0.053	0.21	0.49
BiLSTM	36.88	0.051	0.25	0.49
CNN + LSTM	19.20	0.056	0.08	0.33
Baseline machine learning models				
SGD + OR	28.69	0.052	0.26	0.47
LR + OR	15.35	0.051	0.12	0.29
SVC + OR	27.72	0.049	0.24	0.45
RF + OR	16.21	0.051	0.14	0.31
SGD + BR	28.98	0.051	0.26	0.47
LR + BR	15.35	0.051	0.12	0.29
SVC + BR	27.72	0.049	0.24	0.45
RF + BR	16.21	0.051	0.14	0.31
SGD + CC	39.29	0.055	0.29	0.49
LR + CC	31.67	0.056	0.16	0.43
SVC + CC	41.12	0.053	0.26	0.51
RF + CC	25.09	0.051	0.16	0.40
SGD + LP	38.09	0.062	0.29	0.45
LR + LP	39.00	0.060	0.20	0.45
SVC + LP	40.44	0.059	0.29	0.47
RF + LP	37.92	0.062	0.29	0.45
SGD + RakelD	38.09	0.062	0.29	0.45
LR + RakelD	39.00	0.060	0.20	0.45
SVC + RakelD	40.44	0.059	0.29	0.47
RF + RakelD	37.92	0.062	0.24	0.44
BRkNNa	5.27	0.103	0.01	0.09
BRkNNb	24.16	0.060	0.19	0.36
MLARAM	20.50	0.080	0.02	0.26
MLkNN	24.86	0.054	0.08	0.31

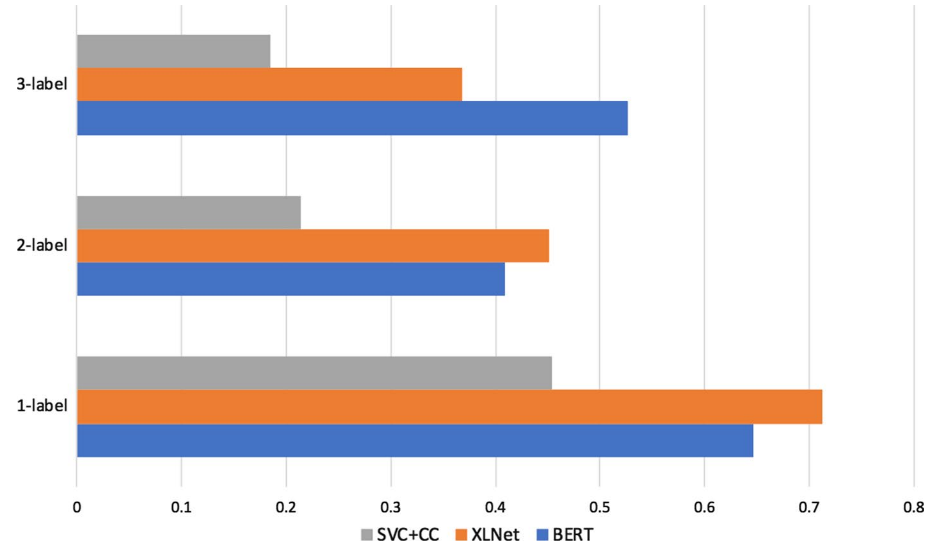


Fig. 5 The label group accuracy of YP

the Macro/Micro F1 of the models are among the highest. Both models significantly outperform the rest of other models. BERT, the second best model, leads the third best model, SVC + CC, by a significant margin of 20.53%. Such results prove that the proposed transfer learning models are capable of multi-label ABSA, and they outperform mainstream deep learning and machine learning models.

The subset accuracy only measures the performance by examples, a finer grained analysis should be considered since different examples may have different amount of 1s in their labels. We report *the label group accuracy* (LGA), which measures the accuracy in a certain label group. A label group contains examples that have the same amount of classes that are labelled as 1. The proposed label group accuracy is defined as follows:

$$\mathrm{LGA} = \frac{\text{number of accurately classified examples in a label group}}{\text{total examples in a label group}} \tag{18}$$

There are three label groups found in YP, which are 1-label group (only one class labelled as 1), 2-label group, and 3-label group. Figure 5 shows the LGA of the top three models. Both transfer learning models consistently outperform the SVC + CC model in all the label groups, with XLNet yielding the best accuracy in the 1-label and 2-label groups. BERT performs the best in the 3-label group, where it achieves an accuracy of 52.63%.

We also report the accuracy achieved by those three models, based on certain classes across the entire testing set. In particular, we select the top-three most frequent classes in YP, namely Food-Positive, Food-Negative, and Misc-Positive. Table 9 shows the accuracy values of respective models in each class, where the transfer learning models consistently lead in the classification performance. With the proposed transfer learning

Table 9 Class based accuracy of the top-three classes in the YP dataset

Class	Model		
	XLNet	BERT	SVC + CC
Food-Positive	92.11	89.60	78.35
Food-Negative	94.25	93.45	90.42
Misc-Positive	97.06	97.25	89.12

models excelling in the majority classes, it exhibits the proposed approaches are capable of capturing multi-label aspect based sentiments from online consumer reviews.

Result analysis of multi-label AESA

The results of multi-label AESA are shown in Tables 10 and 11, respectively. In WE, BERT yields the best subset accuracy of 79.13%, followed by XLNet's 78.41%. RF + LP

Table 10 Experiment results of wine reviews

Model	Accuracy (%)	Hamming loss	Macro F1	Micro F1
Proposed models				
BERT	79.13	0.021	0.86	0.92
XLNet	78.41	0.021	0.86	0.92
Baseline deep learning models				
LSTM	58.01	0.037	0.72	0.85
BiLSTM	56.75	0.039	0.73	0.85
CNN + LSTM	51.92	0.042	0.64	0.83
Baseline machine learning models				
SGD + OR	34.10	0.053	0.58	0.77
LR + OR	38.03	0.049	0.65	0.80
SVC + OR	47.92	0.041	0.77	0.84
RF + OR	64.97	0.029	0.83	0.88
SGD + BR	33.81	0.053	0.58	0.77
LR + BR	38.03	0.049	0.64	0.80
SVC + BR	47.92	0.041	0.77	0.84
RF + BR	63.35	0.030	0.82	0.88
SGD + CC	50.51	0.056	0.62	0.78
LR + CC	54.24	0.052	0.67	0.80
SVC + CC	64.12	0.039	0.79	0.85
RF + CC	68.06	0.030	0.83	0.88
SGD + LP	58.25	0.051	0.72	0.80
LR + LP	58.16	0.049	0.84	0.87
SVC + LP	70.94	0.034	0.84	0.87
RF + LP	72.54	0.035	0.82	0.87
SGD + RakelD	46.95	0.052	0.70	0.80
LR + RakelD	56.13	0.049	0.68	0.81
SVC + RakelD	57.47	0.038	0.83	0.85
RF + RakelD	70.82	0.034	0.83	0.87
BRkNNa	45.91	0.062	0.67	0.76
BRkNNb	46.13	0.060	0.66	0.77
MLARAM	50.53	0.044	0.71	0.79
MLkNN	48.25	0.055	0.68	0.77

Table 11 Experiment results of movie reviews

Model	Accuracy (%)	Hamming loss	Macro F1	Micro F1
Proposed models				
BERT	87.57	0.011	0.95	0.96
XLNet	89.86	0.009	0.94	0.97
Baseline deep learning models				
LSTM	76.99	0.021	0.87	0.92
BiLSTM	71.34	0.025	0.82	0.90
CNN + LSTM	76.73	0.021	0.87	0.92
Baseline machine learning models				
SGD + OR	74.88	0.022	0.94	0.92
LR + OR	76.11	0.021	0.92	0.92
SVC + OR	80.78	0.017	0.98	0.94
RF + OR	73.40	0.023	0.95	0.92
SGD + BR	75.06	0.022	0.94	0.92
LR + BR	76.11	0.021	0.92	0.92
SVC + BR	80.78	0.017	0.98	0.94
RF + BR	72.07	0.023	0.95	0.92
SGD + CC	75.16	0.022	0.95	0.92
LR + CC	76.25	0.021	0.92	0.92
SVC + CC	80.88	0.017	0.98	0.94
RF + CC	72.99	0.024	0.94	0.91
SGD + LP	74.58	0.023	0.94	0.92
LR + LP	74.36	0.024	0.95	0.91
SVC + LP	76.08	0.021	0.98	0.92
RF + LP	72.64	0.024	0.97	0.91
SGD + RakelD	72.73	0.024	0.94	0.91
LR + RakelD	73.58	0.025	0.93	0.91
SVC + RakelD	76.42	0.021	0.98	0.93
RF + RakelD	72.56	0.024	0.97	0.91
BRkNNa	71.93	0.025	0.96	0.91
BRkNNb	56.63	0.042	0.89	0.81
MLARAM	30.85	0.048	0.89	0.82
MLkNN	61.26	0.029	0.91	0.88

takes the third position across all the baseline models. The Hamming losses and the F1 scores achieved by the two transfer learning models are among the lowest and among the highest, respectively. In RT, XLNet returns to the first position with its accuracy of 89.86%, followed by BERT's 87.57%. The third best model, SVC + CC, falls behind of the best one by nearly 10%.

All the examples in WE have exactly three classes labelled as 1. This is because a certain type of wine only belongs to one country, one variety, and can only be associated to one sentiment. Hence, the LGA is identical to the subset accuracy in WE. We select the top-five most frequent classes to show the models' performances on each class. Those five classes include the top-two most frequent aspects, U.S.A. (country), Pinot Noir (variety), and the three sentiment polarities, Positive, Neutral, and Negative. Table 12 summaries the class based accuracy for each model. Both BERT and XLNet are leading the baseline model in terms of all the classes.

Table 12 Class based accuracy of the top-five classes in the WE dataset

Class	Model		
	XLNet	BERT	RF + LP
U.S.A.	98.31	98.42	92.67
Pinot Noir	97.74	98.03	95.75
Positive	93.53	93.15	88.60
Neutral	89.96	89.49	84.49
Negative	96.39	96.34	94.69

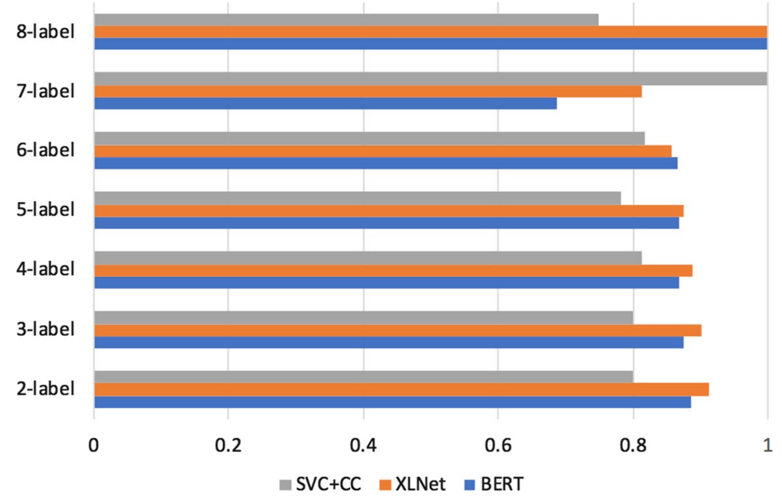


Fig. 6 The label group accuracy of RT

The LGA of the top three models in RT is shown in Fig. 6. BERT and XLNet take the lead in performance in all the label groups, except for the seven-label group, where the baseline model performs the best. XLNet shows a slight edge over BERT in the groups of 2, 3, 4, and 5. Both models perform equally well in the eight-label group.

Table 13 shows the class based accuracy of the three best models. The five classes include three aspects and two sentiment polarities. The baseline model slightly

Table 13 Class based accuracy of the top-five classes in the RT dataset

Class	Model		
	XLNet	BERT	SVC + CC
Action	99.95	99.95	99.87
Comedy	99.89	99.95	99.90
Drama	99.89	99.95	99.85
Positive	90.55	87.92	80.58
Negative	90.55	87.91	80.58

outperform XLNet in the comedy aspect prediction. However, the transfer models outperform the baseline model in predicting the sentiments as well as the rest of other aspects.

Discussions

Main findings

The experiment results undoubtedly exhibit the superior performances of our proposed transfer learning models in multi-label ABSA and AESA, comparing to deep learning models and state-of-the-art multi-label classification methods. In two out of the three datasets, YP and RT, XLNet outperforms all other model configurations, including BERT, in accuracy, hamming loss, macro and micro F1s. Given the limited size of the labelled data in the YP dataset, we believe such results are attributed to several reasons. According to the original introduction article of XLNet [27], XLNet is built based on several pretraining novelties (including factorial permutations, independent of data corruption, segment recurrence mechanism, and transformer refactorization), which performs better with limited labelled data to finetune the pretrained model. However, BERT runs much faster in terms of the training time, where it takes about 0.7 s/batch on a single GPU (Nvidia GTX 1080Ti), whereas XLNet uses approximately 1.4 s/batch. Based on this observation, we recommend using BERT for transfer learning based multi-label classification, for speedier training processes.

Additionally, previous studies [9, 17] reported that ECC and Random Forest of Predictive Clustering Trees (RF-PCT) performed best in their experiments. Although ECC (CC) performed best among all baseline machine learning models in two out of three datasets (YP and RT), RF-PCT did not pass our initial screening due to inferior performances. Such findings are consistent with previous studies'.

Research and practical impacts

Developing intelligent applications to extract opinions and polarities from social media contents is an very important and relevant topic in the field of big data analytics. The proposed transfer learning based multi-label classification models are particularly useful in extracting sentiments from social media contents [41]. Due to the huge volume of data made available by social media, and its fast changing nature, relying on transfer learning models with the additional multi-label classification models can relax the constraints of large amounts of labelled data, which often are unavailable in different domains.

In addition to the extension of multi-label classification capabilities to the ABSA methods, we also propose a new Aspect Enhance Sentiment Analysis (AESA) approach, which extracts document-level (in contrast to sentence-level sentiments) in association with entity-level aspects (in contrast to topics discussed in detail). Additionally, this approach can predict usefulness/informativeness, as well as entity aspect signals from social media contents. Thus, the proposed AESA approach can lead to a variety of (big) data analytics applications, including tagging systems of social media contents, and (automated) querying machines [42]. Additionally, some aspects are not explicitly expressed in the text contents. Thus, the traditional keyword-matching based tagging mechanism may not always work.

With the assistance of the proposed AESA method, implicit aspects (i.e., aspects needed to be inferred) can be predicted given the text contents.

Limitations and future research

We acknowledge a few limitations of our proposed approach in this study, which may point to the directions of future research. Firstly, in the experiment results, we observe that BERT performs better than XLNet in terms of higher-label groups (e.g., the 3-label group in the YP dataset, and the 7-/8-label group in the RT dataset). Future studies can focus on the reason(s) to this phenomenon. Secondly, the transfer learning models yielded restricted performances when the input information is limited to sentence level. It may be interesting to investigate how the models respond to other sentence level analyses. Last but not the least, the transfer learning models are memory-intensive, thus, the lengths of the input sequences (i.e., texts) are limited. Better network compression techniques can be investigated to relax the memory requirements of the transfer learning models.

Conclusion

We propose a transfer learning based approach to enhance the analytical capabilities of recent developments in the field of sentiment analysis. The existing ABSA methods focus on predicting a single aspect-sentiment label at the sentence level. We design a multi-label ABSA method that predicts one or multiple aspect-sentiment labels from the text. To further extend the analytical capabilities, we design a method to capture the associations between aspects and sentiments in social media contents for the purpose of using the detailed aspects to enhance document (e.g., review) level sentiments. We term this type of sentiment analysis as Aspect Enhanced Sentiment Analysis (AESA). AESA is naturally a multi-label classification method, and it is capable of predicting/inferring implicit, entity aspect(s) from text. We employ the state of the art transfer learning models as the analytical vehicle of the proposed multi-label ABSA and AESA methods. Specifically, we extend two transfer learning models, namely BERT and XLNet, by making them end-to-end differentiable based on the multi-labelled data, such that the models can be trained directly using the gradients backpropagated from the errors.

We design a comprehensive empirical evaluation for the proposed approach. Three datasets from different domains are selected in the experiment: Yelp restaurant reviews (YP) are used to evaluate the proposed approach on multi-label ABSA tasks, and wine/movie review (WE and RT, respectively) data are used to evaluate the proposed approach on multi-label AESA tasks. For comparison purposes, we select 27 mainstream multi-label machine learning and deep learning techniques, which are widely adopted in previous multi-label classification studies, and apply them on all the datasets. Additionally, we select four popular evaluation metrics in the context of multi-label classification, including the subset accuracy, the Hamming loss, the macro- and micro-average F1-scores, which cover example and label based metrics. We also complement these metrics with class based accuracy, as well as a newly-designed label group accuracy (LGA). The experiment results show that the proposed transfer learning models consistently outperform the baseline models across all three datasets. Such results confirm that our approach is more than capable of tackling both the multi-label ABSA and AESA tasks.

Acknowledgements
Not applicable.

Authors' contributions
JT performed the primary literature review, methodology design, and data collection, and also conducted the experiment on the baseline machine learning models. XF conducted the experiment on the transfer learning models and baseline deep learning models and performed the result analysis. JT and XF worked together to draft the manuscript. All authors read and approved the final manuscript.

Funding
Not applicable.

Availability of data and materials
The Yelp data is available at: https://www.yelp.com/dataset/.
The Wine Review data is available at: https://www.kaggle.com/zynicide/wine-reviews.
The Movie Review data is available at: https://www.kaggle.com/rpnuser8182/rotten-tomatoes.
The BERT project is available at: https://github.com/google-research/bert.
The XLNet project is available at: https://github.com/zihangdai/xlnet.

Competing interests
The authors declare that they have no competing interests.

Author details
[1] Dolan School of Business, Fairfield University, 1073 N Benson Rd, Fairfield, CT, USA. [2] School of Information Technology, Illinois State University, Normal, IL, USA.

Received: 16 October 2019 Accepted: 23 December 2019
Published online: 06 January 2020

References

1. Fang X, Zhan J. Sentiment analysis using product review data. J Big Data. 2015;2(1):5.
2. Choi Y, Lee H. Data properties and the performance of sentiment classification for electronic commerce applications. Inf Syst Front. 2017;19(5):993–1012.
3. Deng S, Sinha AP, Zhao H. Adapting sentiment lexicons to domain-specific social media texts. Decis Support Syst. 2017;94:65–76.
4. Lee G, Jeong J, Seo S, Kim C, Kang P. Sentiment classification with word localization based on weakly supervised learning with a convolutional neural network. Knowl Based Syst. 2018;152:70–82.
5. Tao J, Zhou L, Feeney C. I understand what you are saying: leveraging deep learning techniques for aspect based sentiment analysis. In: Proceedings of the 52nd Hawaii international conference on system sciences. IEEE, Maui, Hawaii, USA. University of Hawaii-Manoa. 2019.
6. Siering M, Deokar AV, Janze C. Disentangling consumer recommendations: explaining and predicting airline recommendations based on online reviews. Decis Support Syst. 2018;107:52–63.
7. Akhtar MS, Gupta D, Ekbal A, Bhattacharyya P. Feature selection and ensemble construction: a two-step method for aspect based sentiment analysis. Knowl Based Syst. 2017;125:116–35.
8. Pham DH, Le AC. Learning multiple layers of knowledge representation for aspect based sentiment analysis. Data Knowl Eng. 2018;114(January 2017):26–39.
9. Liu SM, Chen JH. A multi-label classification based approach for sentiment classification. Expert Syst Appl. 2015;42(3):1083–93.
10. Kang Y, Zhou L. RubE: Rule-based methods for extracting product features from online consumer reviews. Inf Manag. 2017;54(2):166–76.
11. Boutell MR, Luo J, Shen X, Brown CM. Learning multi-label scene classification. Pattern Recogn. 2004;37(9):1757–71.
12. Read J, Pfahringer B, Holmes G, Frank E. In: Proceedings of the 20th European conference on machine learning.
13. Tsoumakas G, Katakis I, Vlahavas I. Random k-labelsets for multilabel classification. IEEE Trans Knowl Data Eng. 2011;23(7):1079–89.
14. Spyromitros E, Tsoumakas G, Vlahavas I. An empirical study on several classification algorithms. In: Proceedings of the 5th Hellenic conference on artificial intelligence: theories, models, and applications. 2008. pp. 401–6.
15. Zhang M-L, Zhou Z-H. ML-KNN: a lazy learning approach to multi-label learning. Pattern Recogn. 2007;40:2038–48.
16. Benites F, Sapozhnikova E. HARAM: a hierarchical ARAM neural network for large-scale text classification. In: Proceedings-15th IEEE international conference on data mining workshop, ICDMW 2015, No. 7. 2016. pp. 847–54.
17. Madjarov G, Kocev D, Gjorgjevikj D, Džeroski S. An extensive experimental comparison of methods for multi-label learning. Pattern Recogn. 2012;45(9):3084–104.
18. Huang G, Liu Z, van der Maaten L, Weinberger KQ. Densely connected convolutional networks. In: 2017 IEEE conference on computer vision and pattern recognition (CVPR), IEEE. 2017. pp. 2261–9.
19. Long M, Cao Y, Wang J, Jordan MI. Learning transferable features with deep adaptation networks. In: Proceedings of the 32nd international conference on international conference on machine learning-Vol. 37. JMLR.org. 2015. pp. 97–105.

20. Mikolov T, Sutskever I, Chen K, Corrado GS, Dean J. Distributed representations of words and phrases and their compositionality. In: Advances in neural information processing systems. 2013. pp. 3111–9.
21. Howard J, Ruder S. Universal language model fine-tuning for text classification. In: Proceedings of the 56th annual meeting of the association for computational linguistics (Vol. 1: Long Papers). 2018. pp. 328–39.
22. Merity S, Xiong C, Bradbury J, Socher R. Pointer sentinel mixture models. In: Proceedings of the international conference on learning representations. 2017.
23. Radford A, Narasimhan K, Salimans T, Sutskever I. Improving language understanding by generative pre-training. Technical report, OpenAI.
24. Wang A, Singh A, Michael J, Hill F, Levy O, Bowman SR. Glue: a multi-task benchmark and analysis platform for natural language understanding. arXiv preprint arXiv:1804.07461. 2018.
25. Do HH, Prasad PWC, Maag A, Alsadoon A. Deep learning for aspect-based sentiment analysis: a comparative review. Expert Syst Appl. 2019;118:272–99.
26. Devlin J, Chang M-W, Lee K, Toutanova K. Bert: pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805. 2018.
27. Yang Z, Dai Z, Yang Y, Carbonell J, Salakhutdinov R, Le QV. Xlnet: generalized autoregressive pretraining for language understanding. arXiv preprint arXiv:1906.08237. 2019.
28. Vaswani A, Shazeer N, Parmar N, Uszkoreit J, Jones L, Gomez AN, Kaiser Ł, Polosukhin I. Attention is all you need. In: Advances in neural information processing systems. 2017. pp. 5998–6008.
29. The Yelp restaurant reviews. https://www.yelp.com/dataset/.
30. ABSA Labelling Schema. http://alt.qcri.org/semeval2014/task4/data/uploads/semeval14_absa_annotationguidelines.pdf.
31. Wine Reviews. https://www.kaggle.com/zynicide/wine-reviews.
32. Movie Reviews. https://www.kaggle.com/rpnuser8182/rotten-tomatoes.
33. Zhang L, Wang S, Liu B. Deep learning for sentiment analysis: a survey. Wiley Interdiscip Rev Data Mining Knowl Discov. 2018;8(4):1253.
34. Hochreiter S, Schmidhuber J. Long short-term memory. Neural Comput. 1997;9(8):1735–80.
35. Fang X, Xu M, Xu S, Zhao P. A deep learning framework for predicting cyber attacks rates. EURASIP J Inf Secur. 2019;2019(1):5.
36. Althelaya KA, El-Alfy EM, Mohammed S. Evaluation of bidirectional lstm for short-and long-term stock market prediction. In: 2018 9th international conference on information and communication systems (ICICS). 2018. pp. 151–6.
37. Cui Z, Ke R, Wang Y. Deep bidirectional and unidirectional lstm recurrent neural network for network-wide traffic speed prediction. arXiv preprint arXiv:1801.02143. 2018.
38. Fang X, Yuan Z. Performance enhancing techniques for deep learning models in time series forecasting. Eng Appl Artif Intell. 2019;85:533–42.
39. Lin T, Guo T, Aberer K. Hybrid neural networks for learning the trend in time series. In: Proceedings of the 26th international joint conference on artificial intelligence. AAAI Press. 2017. pp. 2273–9.
40. Liu J, Zhao K, Kusy B, Wen J-R, Jurdak R. Temporal embedding in convolutional neural networks for robust learning of abstract snippets. arXiv preprint arXiv:1502.05113. 2015.
41. Zhou S, Qiao Z, Du Q, Wang GA, Fan W, Yan X. Measuring customer agility from online reviews using big data text analytics. J Manag Inf Syst. 2018;35(2):510–39.
42. Zhou L, Pan S, Wang J, Vasilakos AV. Machine learning on big data: opportunities and challenges. Neurocomputing. 2017;237(December 2016):350–61.

Publisher's Note

Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Terms and Conditions