SentiHood: Targeted Aspect Based Sentiment Analysis Dataset for Urban Neighbourhoods

Marzieh Saeidi
University College London
msaeidi@cs.ucl.ac.uk

Guillaume Bouchard
Bloomsbury AI
guillaume@bloomsbury.ai

Maria Liakata
University of Warwick
m.liakata@warwick.ac.uk

Sebastian Riedel
University College London
sriedel@cs.ucl.ac.uk

Abstract

In this paper, we introduce the task of targeted aspect-based sentiment analysis. The goal is to extract *fine-grained* information with respect to entities mentioned in user comments. This work extends both *aspect-based* sentiment analysis that assumes a single entity per document and *targeted* sentiment analysis that assumes a single sentiment towards a target entity. In particular, we identify the sentiment towards each aspect of one or more entities. As a testbed for this task, we introduce the SentiHood dataset, extracted from a question answering (QA) platform where urban neighbourhoods are discussed by users. In this context units of text often mention several aspects of one or more neighbourhoods. This is the first time that a generic social media platform in this case a QA platform, is used for fine-grained opinion mining. Text coming from QA platforms is far less constrained compared to text from review specific platforms which current datasets are based on. We develop several strong baselines, relying on logistic regression and state-of-the-art recurrent neural networks.

1 Introduction

Sentiment analysis is an important task in natural language processing. It has received not only a lot of interest in academia but also in industry, in particular for identifying customer satisfaction on products and services. Early research in the field (Das and Chen, 2001; Morinaga et al., 2002) of sentiment analysis only focused on identifying the overall sentiment or polarity of a given text. The underlying assumption of this work was that there is one overall polarity in the whole text.

Aspect-based sentiment analysis (ABSA) (Jo and Oh, 2011; Pontiki et al., 2015; Pontiki et al., 2016) relates to the task of extracting fine-grained information by identifying the polarity towards different aspects of an entity in the same unit of text, and recognizing the polarity associated with each aspect separately. The datasets for this task were mostly based on specialized review platforms such as Yelp where it is assumed that only one entity is discussed in one review snippet, but the opinion on multiple aspects can be expressed. This task is particularly useful because a user can assess the aggregated sentiment for each individual aspect of a given product or service and get a more fine-grained understanding of its quality.

Another line of research in this field is *targeted* (a.k.a. target-dependent) sentiment analysis (Jiang et al., 2011; Vo and Zhang, 2015). Targeted sentiment analysis investigates the classification of opinion polarities towards certain target entity mentions in given sentences (often a tweet). For instance in the sentence "People everywhere love Windows & vista. Bill Gates", polarity towards Bill Gates is "Neutral" but the positive sentiment towards Windows & vista will interfere with identifying it if the usual methods for sentiment analysis task are employed. However this task assumes only the *overall* sentiment for each entity. Moreover, the existing corpora for this task so far has contained only a single target entity per unit of text.

Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers, pages 1546–1556, Osaka, Japan, December 11-17 2016.

Both settings are obviously limited, and there exists many scenarios in which sentiments towards different aspects of several entities are discussed in the same unit of text. As a running example, we use urban areas: choosing which area to live or to visit is an important task when moving or visiting a new city. Currently there are no dedicated platforms for reviewing and rating aspects of neighbourhoods of a city. However we can find many discussions and threads on several blogs and question answering platforms that discuss aspects of areas in many cities around the world. In general, these conversations are very comprehensible: they often contain specific information about several aspects of several neighbourhoods. One example is the following (area names are highlighted in bold and aspect related terms are underlined):

"*Other places to look at in South London are* ***Streatham*** *(good range of shops and restaurants, maybe a bit far out of central London but you get more for your money)* ***Brixton*** *(good transport links, trendy, can be a bit edgy)* ***Clapham*** *(good transport, good restaurants/pubs, can feel a bit dull, expensive) ...*"

The example above does not perfectly fit into the existing tasks in sentiment analysis mentioned earlier. In this work, we introduce a new task that not only subsumes the existing sub-fields of *targeted* and *aspect-based* sentiment analysis but it also makes less assumptions on the number of entities that can be discussed in the unit of text.

To compare with the existing aspect-based sentiment analysis task, take the following example from the restaurant dataset used by SemEval shared ABSA (Pontiki et al., 2016) task. "*The design of the space is good but the service is horrid!*". The ABSA task aims to identify that a positive sentiment towards the *ambiance* aspect is expressed (opinion target expression is "space"). Moreover, a negative sentiment is expressed towards the *service* aspect (opinion target expression is "service"). In this example, it is assumed that both of these opinions are expressed about a single restaurant which is not mentioned explicitly. However, take the following *synthetic* example that ABSA is not addressing:

"*The design of the space is good in* ***Boqueria*** *but the service is horrid, on the other hand, the staff in* ***Gremio*** *are very friendly and the food is always delicious.*"

In this example, more than one restaurant are discussed and restaurants for which opinions are expressed, are explicitly mentioned. We call these target entities. Current ABSA task can only recognise that positive and negative opinions towards aspect "service" are expressed. But it can not identify the target entity for each of these opinions (i.e. Germio and Boqueria respectively). Targeted aspect-based sentiment analysis handles extracting the target entities as well as different aspects and their relevant sentiments.

In the following, we argue that this task is both very relevant in practice, and raises interesting modelling questions. To facilitate research on this task we introduce the SentiHood dataset. SentiHood is based on the text from a QA platform in the domain of neighbourhoods of a city. Table 2 shows examples of input sentences and annotations provided.

Sentence	**Labels**
The cheap parts of London are **Edmonton** and **Tottenham** and they are all poor, crime ridden and crowded with immigrants	(Edmonton,price,Positive) (Tottenham,price,Positive) (Edmonton,safety,Negative) (Tottenham,safety,Negative)
Hampstead area, more expensive but a better quality of living than in **Tufnell Park**	(Hampstead,price,Negative) (Hampstead,live,Positive)

Table 1: Examples of input sentences and output labels in the system.

Our contributions in this paper can be summarised as follows:

- We introduce the task of *targeted aspect-based* sentiment analysis as a further step towards extracting more fine-grained information from more complex text in the field of sentiment analysis.
- We use the text from social media platforms, in particular QA, for fine-grained opinion mining. So far, all datasets in this field have utilised text from review specific platforms where certain assumptions can be made and data is more constrained and less noisy.
- We propose SentiHood, a benchmark dataset that is annotated for the task of targeted aspect-based sentiment analysis in the domain of urban neighbourhoods.
- We show that despite the fact that the texts in QA were not written with the goal of writing a review in mind, question answering platforms and online forums are in general rich in information.
- We provide strong baselines for the task using both logistic regression and Long Short Term Memory (LSTM) networks and analysis of the results.

2 SentiHood

SentiHood is a dataset for the task of *targeted aspect-based* sentiment analysis. It is based on the text taken from question answering platform of Yahoo! Answers that is filtered for questions relating to neighbourhoods of the city of London. In this section we explain the data collection and annotation process and summarise properties of the dataset.

2.1 Data Collection Process

Entities in the dataset are locations or neighbourhoods. Yahoo! Answers was queried using the name of each neighbourhood of the city of London. Location (entity) names were taken from the gazetteer GeoNames[1] and restricted to those within the boundaries of London. This list includes names of areas and boroughs and therefore entities are not always geographically exclusive (a borough contains several areas or neighbourhoods). The content of each question-answer pairs was aggregated and split into sentences. We keep only sentences that have a mention of a location entity name and discard other sentences.

2.2 Categories

The Number of location mentions in a single sentence in our dataset varies from one to over 50. To simplify the task, we only annotate sentences that contain one or two location mentions. These sentences were divided into two groups: sentences containing one location mention — Single, and sentences containing two location mentions — Multi. This is to observe the difficulty of annotating two groups by human annotators and by the models.

2.3 Aspects

Like existing work in the aspect-based sentiment analysis task (Brychcın et al., 2014), a pre-defined list of aspects is provided for annotators to choose from. These aspects are: *live*, *safety*, *price*, *quiet*, *dining*, *nightlife*, *transit-location*, *touristy*, *shopping*, *green-culture* and *multicultural*. Adding an additional aspect of *misc* was considered. However in the initial round of annotations, we realised that it had a negative effect on the decisiveness of annotators and it led to a lower overall agreement. Aspect *general* refers to a generic opinion about a location, e.g. "I love **Camden Town**".

2.4 Sentiment

For each selected aspect, annotators were required to select a polarity or sentiment. Most work in this area considers three sentiment categories of "Positive". "Negative" and "Neutral". In our annotation however, we only provided "Positive" and "Negative" sentiment labels. This is because in our data we rarely come across cases where aspects are discussed without a polarity.

[1] http://www.geonames.org/

2.5 Target Entity

Target entity is a location entity in which an opinion (aspect and sentiment) is expressed for. We also refer to target entity as target location.

2.6 Out of scope

For the sentences that do not comply with our schema, we define the two following special labels. Sentences marked with one of the these labels are removed from the dataset.

1. **Irrelevant**: When the identified name does not refer to a location entity: for example in the sentence "**Notting Hill** (1999) stars Julia Roberts and Hugh Grant use the characteristic features of the area as a backdrop to the action", "Notting Hill" refers to the movie and not the area.

2. **Uncertain**: When two contradicting sentiments are expressed for the same location and aspect, e.g. "Like any other area, **Camden Town** has good and bad parts". Moreover, when the opinion is expressed for an area without a direct mention in the sentences, e.g. "**It**'s a very trendy area and not too far from **King's Cross**".

2.7 Procedure:

We use the BRAT annotation tool (Stenetorp et al., 2012) to simplify the annotation task. Three annotators were initially selected for the task. None of the annotators are experts in linguistics. Annotators began by reading the guidelines and examples. Each annotator was then required to annotate a small subset of the data. After each round of annotation, agreements between annotators were calculated and discussed and this procedure continued until they reached a reasonable agreement. 10% of the whole dataset was randomly selected and annotated by all the three annotators. The annotator with the highest inter-annotator agreement was selected to annotate all the dataset.

Agreements: Cohen's Kappa coefficient(K) (COHEN, 1960) is often used for measuring the pairwise agreement between each two annotators for the task of aspect-based sentiment analysis (Gamon et al., 2005; Ganu et al., 2009) and other tasks (Liakata et al., 2010). The Kappa Coefficient is calculated over aspect-sentiment pairs per each location. Pairwise inter-annotator agreement for aspect categories measured using Cohen's Kappa is 0.73, 0.78 and 0.70, which is deemed of sufficient quality. It is worth mentioning that agreements on different aspect categories varied, with some aspects having a higher agreement rate. Agreements for aspect expressions are 0.93, 0.94, 0.93. These agreements indicate reasonably high inter-annotator agreements (Pavlopoulos, 2014).

Disagreements: Main disagreements between annotators occurred in detecting the aspect rather than detecting the sentiment, aspect expression or the target location. For instance, some annotators associated the expression "residential area" with a "Positive" sentiment for aspect "quiet" or "live" and others did not agree that "residential" implies quietness or desirable for living. In the case of disagreements, the vote of the majority was considered as the correct annotation.

Some ambiguity was also observed with respect to detecting the target location. This occurred mainly when a location is confined in another location. For instance the sentence "**Angel** in **Inslington** has many great restaurants for eating out" expresses a "Positive" sentiment for the aspect "dining" of area **Angel** which is within the borough of **Islington**. Some annotators suggested that the sentence also implies the same opinion for **Islington**. However at the end all annotators agreed that in such cases no implicit assumptions should be made and only confined area should be labeled.

2.8 Dataset

SentiHood currently contains annotated sentences containing one or two location entity mentions.[2] SentiHood contains 5215 sentences with 3862 sentences containing a single location and 1353 sentences containing multiple (two) locations. Figure 1 shows the number of sentences that are labeled with each aspect, breaking down on the sentiment "Positive" or "Negative". "Positive" sentiment is dominant for

[2]SentiHood data can be obtained at `http://annotate-neighborhood.com/download/download.html`

aspects such as dining and shopping. This shows that for some aspects, people usually talk about areas that are good for it as oppose to areas that are not. The *general* aspect is the most frequent aspect with over 2000 sentences while aspect *touristy* has occurred in less than 100 sentences. Notice that since each sentence can contain one or more opinions, the total number of opinions (5920) in the dataset is higher than the number of sentences.

Location entity names are masked by **location1** and **location2** in the whole dataset, so the task does not involve identification and segmentation of the named entities. We also provide the dataset with the original location entity names.

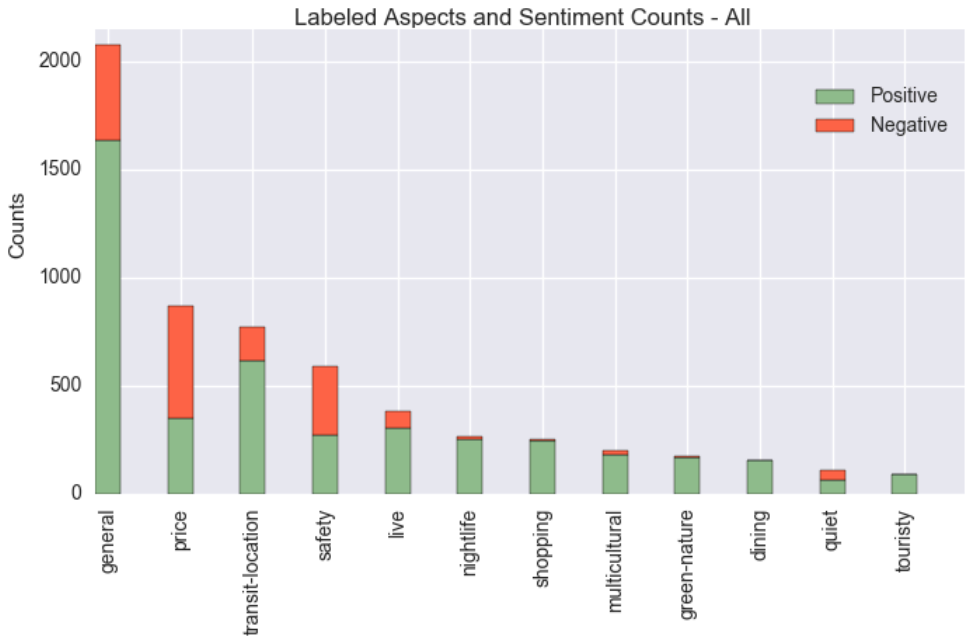

Figure 1: Number of annotated aspects and their sentiments.

3 Task

We define the task of targeted aspect-based sentiment analysis as follows: given a unit of text s (for example, a sentence), provide a list of tuples (labels) $\{(l, a, p)\}_{t=0}^{T}$, where p is the polarity expressed for the aspect a of entity l. Each sentence can have zero to T number of labels associated with it.

Within the current aspect-based sentiment analysis work, three tasks are defined (Brychcın et al., 2014): detecting the aspect, detecting the opinion target expression and detecting the sentiment, with detecting the opinion target expression being an intermediary task for identifying the sentiment of the aspect.

Here we focus on identifying only the aspect and sentiment for each entity. We identify each aspect, its relevant sentiment and the target location entity jointly by introducing a new polarity class called "None". "None" indicated that a sentence does not contain an opinion for the aspect a of location l. Therefore the overall task can be defined as a three-class classification task for each (l, a) pair with labels "Positive", "Negative", "None". Table 2 shows an example of the input sentence and output labels.

Sentence	**Labels**
location1 is very safe and **location2** is too far	(location1,safety,Positive) (location1,transit-location,None) (location2,safety,None) (location2,transit-location,Negative)

Table 2: Example of an input sentence and the output labels.

4 Evaluation

Most existing work in aspect-based sentiment analysis field, report F_1 measure for aspect detection task, and accuracy for sentiment classification. The scores can be calculated over 2-class or 3-class sentiments (Pontiki et al., 2015). In our results, F_1 score is calculated with a threshold that is optimized on validation set.

We also propose the AUC (area under the ROC curve) metric for both aspect and sentiment detection tasks. AUC captures the quality of the ranking of output scores and does not rely on a threshold.

5 Baseline

Here we propose baselines for the task. In all our methods, we treat the task as a three-class classification for each aspect and use a softmax function as follows:

$$p(y_{l,a} = c) = \text{softmax}(c) = \frac{exp(w_c.e_l + b_c)}{\sum_{c'=1}^{C} exp(w_{c'}.e_l + b_{c'})} \quad (1)$$

where $y_{l,a}$ is the sentiment label of aspect a for location l. w_c and b_C are the weights and the bias specific to each sentiment class c, respectively. e_l is a representation of location l. This representation can be a BoW or a distributional representation. Each method that we propose here define their own specific representation for e_l.

5.1 Logistic Regression

Many existing works in the aspect-based sentiment analysis task,[3] use a classifier, such as logistic regression or SVM, based on linguistic features such as n-grams, POS information or more hand-engineered features. We can think of these features as a sparse representation e_l that enter the softmax in equation 1. More concretely, we define the following sparse representations of locations:

Mask target entity n-grams: For each location, we define an n-gram representation over the sentence and mask the target location using a special token. This can help to differentiate between representations of two locations present in the same sentence.

Left-right n-grams: we create an n-gram representation for both the right and the left context around each location mention. We then concatenate these two representations to obtain one single feature vector.

Left right pooling: Previously embedding representations over the left and right context have been used for automatic feature detection in the targeted sentiment analysis task (Vo and Zhang, 2015). Inspired by this approach, we obtain max, min, average and standard deviation pooling over all the word embeddings for left and right context separately. We then combine the pooled embeddings of the left and right context to obtain a single feature vector. Word embeddings are obtained by running word2vec tool on a combination of our Yahoo! Answers corpus and a substantially big corpus from the web.[4]

5.2 Long Short-Term Memory (LSTM)

Inspired by the recent success of applying deep neural networks on language tasks, we use a bidirectional LSTM (Hochreiter and Schmidhuber, 1997) to learn a classifier for each of the aspects. Representations for a location (e_l) are obtained using one of the following two approaches:

Final output state (LSTM - Final): e_l is the output embedding of the bidirectional LSTM.

Location output state (LSTM - Location): e_l is the output representation at the index corresponding to the location entity as illustrated in Figure 2.

[3] including participants of SemEval ABSA tasks

[4] http://ebiquity.umbc.edu/redirect/to/resource/id/351/UMBC-webbase-corpus

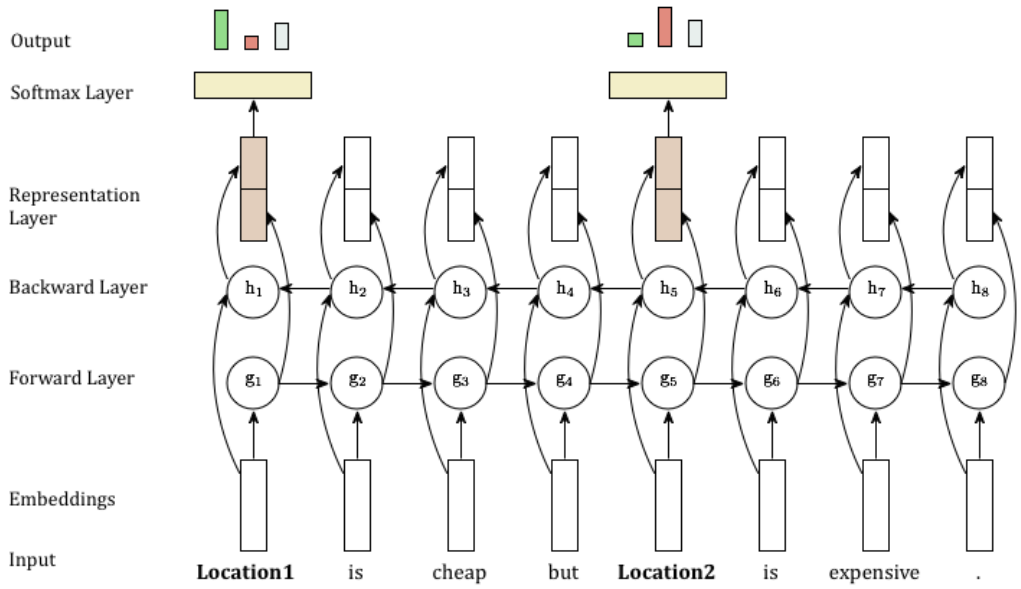

Figure 2: Bidirectional LSTM outputs a representation for each token in the sentence. The output state at the index of each location is then fed into a softmax layer to identify the sentiment class for the corresponding aspect. In this figure, LSTM is trained to identify the sentiment of aspect "price". Model should predict "Positive" for **location1** and "Negative" for **location2**

6 Experiments

In this paper, we select the four most frequent aspects from the dataset which are: "price", "safety", "transit-location" and "general" but the same approach can be applied to the remaining aspects. We divide each collection of single and multiple location mentions into train, dev and test set, with each having 70%, 10% and 20% of data respectively. We choose the best model with respect to the dev set.

In the case of the LSTM, we evaluate the loss on both training set and dev set after each iteration. We save the best model which has the lowest loss on the dev set over all the iterations. We then run this model on the test set and report the results. We report results separately on both categories of single location sentences and sentences with two locations and over all the data in the test set. Results on single location sentences mainly show the ability of the model to detect the correct sentiment for an aspect. On the other hand, results on two location sentences demonstrate the ability of the system not only on detecting the relevant sentiment of an aspect but also on recognising the target entity of the opinion.

Training LSTMs We implement our LSTM models using tensorflow (ten, 2015). To tackle the problem of having an unbalanced dataset (i.e. too many "None" instances), we train the LSTM model in batches with every batch having the same number of sentences selected randomly from each sentiment class. We tune the hyper parameters of the model on the dev set. The best model uses hidden units of size 50 and batch sizes of size 150. The Adam optimizer is used for optimization with a starting learning rate of 0.01 which is tuned to be the best performing on the dev set. Dropout is used both on initial word embeddings and on LSTM cells with the probability of 0.001. Tensorflow (ten, 2015) is used for the implementation of LSTM.

Training Logistic Regression Logistic regression models were based on implementations from scikit-learn.[5] Since we have an unbalanced dataset, we use a weighted logistic regression. To obtain the best weights, we cross-validate them on the development set. Weights inversely proportional to the size of each class result in the best performance.

[5] `http://scikit-learn.org/`

7 Results

Table 3 shows the results (averaged over all selected aspects) in terms of both F_1/accuracy and AUCs. It also shows the results of logistic regression based models versus LSTM models.

As we can see, the n-gram representation with location masking achieves slightly better results over the left-right context. N-grams include unigrams and bigrams. Also, by adding POS information, we gain an increase in the performance. We also experimented with adding tri-grams but it did not have a positive effect on the overall scores. Separating the left and the right context (LR-Left-Right) for BoW representation, does not improve the performance. Left-right pooling of dense embeddings performed weakly in comparison with other representations and therefore their results were omitted.

Amongst the two variations of LSTM, the model with final state embeddings does slightly better than the model where we use the embeddings at the location index, however they are not significantly different (with a p value less than 0.01). It is interesting to note that the best LSTM model is not superior to logistic regression model, especially in terms of AUC. This can be due to the fact that the amount of training data is not sufficient for LSTM to perform well. Moreover, while we provide some grammar information to logistic regression model through POS tags, such information is not incorporated into LSTM models. Another interesting observation is that the F_1 measure for logistic regression model with n-grams and POS information is very low while this model's performance is superior to other models in terms of AUC. This is because in general, it is easier to rank prediction scores than to assign predicted labels to instances by choosing a hard threshold.

Model	Aspect (F_1)	Sentiment (Accuracy)	Aspect (AUC)	Sentiment (AUC)
LR-Left-Right	0.683	0.847	0.903	0.875
LR-Mask(ngram)	**0.697**	0.853	0.918	0.885
LR-Mask(ngram+POS)	*0.393*	**0.875**	**0.924**	**0.905**
LSTM-Final	0.689	0.820	0.898	0.854
LSTM-Location	**0.693**	0.819	0.897	0.839

Table 3: Results of best logistic regression (LR) models and LSTM models.

Table 4 shows the average AUC (over aspect and sentimentclassification tasks) for two categories of data: Single — sentences that contain one location entity and Multi — sentences that contain two location entities. While logistic regression can perform slightly better on son Single location sentences, LSTM performs slightly better on Multi location sentences.

Model	Single	Multi
LR - Mask (n-gram + POS)	**0.916**	**0.907**
LSTM - Final	0.872	0.890

Table 4: Results of best logistic regression (LR) and LSTM models on sentences with a single location (Single) and multiple locations (Multi). AUC scores are averaged over aspect and sentiment classification tasks.

Table 5 shows the break down of average AUC scores for each aspect. We can see that aspects such as "safety" can be predicted with a better AUC score than aspect "general".

Model	Price	Safety	Transit	General
LR - Mask (n-gram + POS)	**0.940**	**0.960**	**0.879**	0.864
LSTM - Final	0.875	0.932	0.836	**0.869**

Table 5: LR and LSTM performance breakdown on aspects. AUC scores are averaged over aspect and sentiment detection.

Table 6 shows examples of correct and incorrect predictions using the best logistic regression model. The top part of the table contains examples that each contain a single location entity. At the bottom of the table, a sentence with two location entities is provided. The system correctly identifies that a "Positive" sentiment is expressed for the *general* aspect about **location2**. However, no sentiment is expressed for this aspect for **location1**.

Sentence	Aspect	Predicted	Label
location1 is not a nice cheap residential area to live trust me i was born and raised there	Price	Positive	Negative
I think you'd find it tough to find something affordable in **location1**	Price	Positive	Negative
I can't recommend **location1** for affordability	Price	Negative	Negative
I only know about **location1**, most people prefer location2	General	None	None
I only know about location1, most people prefer **location2**	General	Positive	Positive

Table 6: Examples of input sentences and predicted labels using the best system (LR - Mask (n-gram + POS). Target entity locations are highlighted in bold.

8 Related Work

The term sentiment analysis was first used in (et al, 2003). Since then, the field has received much attention from both research and industry. Sentiment analysis has applications in almost in every domain and it raised many interesting research questions. Furthermore, the availability of a huge volume of opinionated data on social media platforms has accelerated the development in this area.

In the beginning work on sentiment analysis mainly focused on identifying the overall sentiment of a unit of text. The unit of text varied from document (Pang et al., 2002; Turney, 2002), paragraph or sentences (Hu and Liu, 2004). However, only considering the overall sentiment fails to capture the sentiments over the aspects on which an entity can be reviewed or sentiment expressed toward different entities. Two remedy this, two new tasks have been introduced: *aspect-based* sentiment analysis and *targeted* sentiment analysis.

Aspect based sentiment analysis assumes a *single entity* per a unit of analysis and tries to identify sentiments towards different aspects of the entity (Lu et al., 2011; Lakkaraju et al., 2014; Alghunaim, 2015; Bagheri et al., 2013; Somprasertsri and Lalitrojwong, 2008; Alghunaim, 2015; Lu et al., 2011; Titov and McDonald, 2008; Brody and Elhadad, 2010). This task however does not consider more than one entity in the given text.

Targeted (target dependent) sentiment analysis is another task that identifies polarity towards a target entity (as opposed to over entire unit of text) (Mitchell et al., 2013; Jiang et al., 2011; Dong et al., 2014; Vo and Zhang, 2015; Zhang et al., 2016). (Jiang et al., 2011) was the first to propose targeted sentiment analysis on Twitter and demonstrates the importance of targets by showing that 40% of sentiment errors are due to not considering them in classification. However this task only identifies the *overall sentiment* and the existing corpora for the task consist only of text with one single entity per unit of analysis.

The task of targeted aspect-based sentiment analysis caters for more generic text by making fewer assumptions while extracting fine-grained information.

9 Conclusion

In this paper, we introduced the task of *targeted aspect-based* sentiment analysis and a new dataset. We also provide two strong baselines using logistic regression and LSTM. Ways to improve the baselines can involve using parse trees for identifying the context of each location. Data augmentation can be used

to make the models and especially LSTM more robust to variations in the data. We also like to provide more detailed analysis of what each system can achieve.

References

Abdulaziz Alghunaim. 2015. *A Vector Space Approach for Aspect-Based Sentiment Analysis*. Ph.D. thesis, Massachusetts Institute of Technology.

Ayoub Bagheri, Mohamad Saraee, and Franciska De Jong. 2013. Care more about customers: unsupervised domain-independent aspect detection for sentiment analysis of customer reviews. *Knowledge-Based Systems*, 52:201–213.

Samuel Brody and Noemie Elhadad. 2010. An unsupervised aspect-sentiment model for online reviews. In *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 804–812. Association for Computational Linguistics.

Tomáš Brychcın, Michal Konkol, and Josef Steinberger. 2014. Uwb: Machine learning approach to aspect-based sentiment analysis. *SemEval 2014*, page 817.

JACOB COHEN. 1960. A coefficient of agreement for nominal scales. *Educational and Psychological Measurement*, 20(1):37–46.

Sanjiv Das and Mike Chen. 2001. Yahoo! for amazon: Extracting market sentiment from stock message boards. In *Proceedings of the Asia Pacific finance association annual conference (APFA)*, volume 35, page 43. Bangkok, Thailand.

Li Dong, Furu Wei, Chuanqi Tan, Duyu Tang, Ming Zhou, and Ke Xu. 2014. Adaptive recursive neural network for target-dependent twitter sentiment classification. In *ACL (2)*, pages 49–54.

Jeonghee Yi et al. 2003. Extracting sentiments about a given topic using natural language processing techniques; jeonghee yi et al; ibm. In *Proceedings of the Third IEEE International Conference on Data Mining (ICDM'03) 0-7695-1978-4/03*, volume 17.

Michael Gamon, Anthony Aue, Simon Corston-Oliver, and Eric Ringger. 2005. Pulse: Mining customer opinions from free text. In *Advances in Intelligent Data Analysis VI*, pages 121–132. Springer.

Gayatree Ganu, Noemie Elhadad, and Amélie Marian. 2009. Beyond the stars: Improving rating predictions using review text content. In *WebDB*, volume 9, pages 1–6. Citeseer.

Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural computation*, 9(8):1735–1780.

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

Long Jiang, Mo Yu, Ming Zhou, Xiaohua Liu, and Tiejun Zhao. 2011. Target-dependent twitter sentiment classification. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies-Volume 1*, pages 151–160. Association for Computational Linguistics.

Yohan Jo and Alice H Oh. 2011. Aspect and sentiment unification model for online review analysis. In *Proceedings of the fourth ACM international conference on Web search and data mining*, pages 815–824. ACM.

Himabindu Lakkaraju, Richard Socher, and Chris Manning. 2014. Aspect specific sentiment analysis using hierarchical deep learning.

Maria Liakata, Simone Teufel, Advaith Siddharthan, Colin R Batchelor, et al. 2010. Corpora for the conceptualisation and zoning of scientific papers. In *LREC*. Citeseer.

Bin Lu, Myle Ott, Claire Cardie, and Benjamin K Tsou. 2011. Multi-aspect sentiment analysis with topic models. In *Data Mining Workshops (ICDMW), 2011 IEEE 11th International Conference on*, pages 81–88. IEEE.

Margaret Mitchell, Jacqueline Aguilar, Theresa Wilson, and Benjamin Van Durme. 2013. Open domain targeted sentiment.

Satoshi Morinaga, Kenji Yamanishi, Kenji Tateishi, and Toshikazu Fukushima. 2002. Mining product reputations on the web. In *Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 341–349. ACM.

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

Ioannis Pavlopoulos. 2014. Aspect based sentiment analysis. *Athens University of Economics and Business*.

Maria Pontiki, Dimitrios Galanis, Haris Papageorgiou, Suresh Manandhar, and Ion Androutsopoulos. 2015. Semeval-2015 task 12: Aspect based sentiment analysis. In *Proceedings of the 9th International Workshop on Semantic Evaluation (SemEval 2015), Association for Computational Linguistics, Denver, Colorado*, pages 486–495.

Maria Pontiki, Dimitrios Galanis, Haris Papageorgiou, Ion Androutsopoulos, Suresh Manandhar, Mohammad AL-Smadi, Mahmoud Al-Ayyoub, Yanyan Zhao, Bing Qin, Orphée De Clercq, Véronique Hoste, Marianna Apidianaki, Xavier Tannier, Natalia Loukachevitch, Evgeny Kotelnikov, Nuria 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 '16, San Diego, California, June. Association for Computational Linguistics.

Gamgarn Somprasertsri and Pattarachai Lalitrojwong. 2008. Automatic product feature extraction from online product reviews using maximum entropy with lexical and syntactic features. In *Information Reuse and Integration, 2008. IRI 2008. IEEE International Conference on*, pages 250–255. IEEE.

Pontus Stenetorp, Sampo Pyysalo, Goran Topić, Tomoko Ohta, Sophia Ananiadou, and Jun'ichi Tsujii. 2012. Brat: a web-based tool for nlp-assisted text annotation. In *Proceedings of the Demonstrations at the 13th Conference of the European Chapter of the Association for Computational Linguistics*, pages 102–107. Association for Computational Linguistics.

2015. TensorFlow: Large-scale machine learning on heterogeneous systems. Software available from tensorflow.org.

Ivan Titov and Ryan McDonald. 2008. Modeling online reviews with multi-grain topic models. In *Proceedings of the 17th international conference on World Wide Web*, pages 111–120. ACM.

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

Duy-Tin Vo and Yue Zhang. 2015. Target-dependent twitter sentiment classification with rich automatic features. In *Proceedings of the Twenty-Fourth International Joint Conference on Artificial Intelligence (IJCAI 2015)*, pages 1347–1353.

Meishan Zhang, Yue Zhang, and Duy-Tin Vo. 2016. Gated neural networks for targeted sentiment analysis. In *Thirtieth AAAI Conference on Artificial Intelligence*.