

# Leveraging Recursive Processing for Neural-Symbolic Affect-Target Associations

Alexander Sutherland, Sven Magg, Stefan Wermter  
 Knowledge Technology, Department of Informatics,  
 University of Hamburg, Germany  
 Hamburg, Germany  
 {sutherland, magg, wermter}@informatik.uni-hamburg.de

**Abstract**—Explaining the outcome of deep learning decisions based on affect is challenging but necessary if we expect social companion robots to interact with users on an emotional level. In this paper, we present a commonsense approach that utilizes an interpretable hybrid neural-symbolic system to associate extracted targets, noun chunks determined to be associated with the expressed emotion, with affective labels from a natural language expression. We leverage a pre-trained neural network that is well adapted to tree and sub-tree processing, the Dependency Tree-LSTM, to learn the affect labels of dynamic targets, determined through symbolic rules, in natural language. We find that making use of the unique properties of the recursive network provides higher accuracy and interpretability when compared to other unstructured and sequential methods for determining target-affect associations in an aspect-based sentiment analysis task.

**Index Terms**—sentiment, recursive, hybrid, neural-symbolic, affective computing, natural language processing

## I. INTRODUCTION

Within the field of affective computing, the community seeks to learn how to recognise emotions in humans across multiple input modalities and then apply this knowledge to perform emotion-aware decision-making [1]. In spite of a large body of emotion recognition research with promising results [2], applications showing how recognised emotions can be applied to interaction scenarios with intelligent agents, such as companion robots, are rare. Emotion recognition is often performed without taking into account the interaction context, interpreting results from the abstract context of the chosen Human-Robot Interaction (HRI)-scenario. From a language perspective, this often entails utterances being associated with emotion labels and robots are to implicitly act without taking into account the granular semantic conversation context.

Understanding how a predicted emotion is reached and how it relates to the specific entities is pivotal for intelligent agents to make decisions surrounding identified emotions, e.g. a subject saying “*You are nice even though you make too much noise.*” should make the agent realise it’s too noisy, despite an overall positive sentiment. Given the prominence of neural networks, this also requires the ability to interpret and reason around neural processing and outcome, which is complicated

for models that perform high dimensional processing, such as the LSTM [3], that process features over a temporal span. Without this ability, intelligent agents lack the understanding to perform even simple emotional reasoning. This can lead to a diminished view of companion robot capability, especially when broaching a topic as intuitive to humans as emotion.

We seek to create interpretable associations that can inform decision-making by understanding how emotions are associated with other semantic elements in natural language. Our approach uses a mixture of commonsense principles, symbolic reasoning, and recursive neural learning to construct associations between affective labels and identified targets in natural language expressions. By adopting principles from similar works [4], [5] and leveraging features of a specific recursive neural architecture [6] we present a system tailored toward HRI-scenarios. The system dynamically identifies multiple *affective targets*, which are noun chunks the system believes the affective expression is aimed towards. It then determines affective language features for the target and provides an affective label for the target.

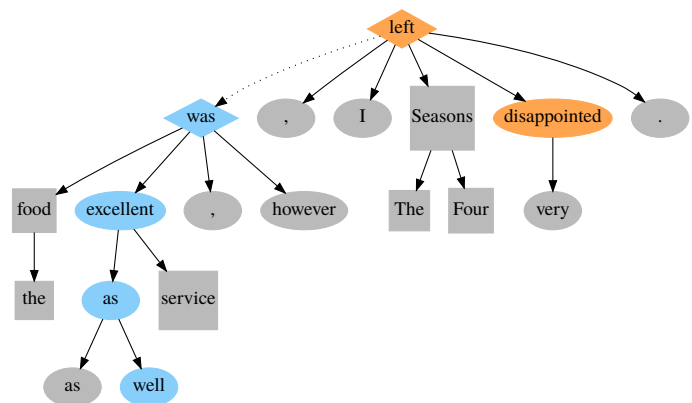


Fig. 1. The sentiment parsed dependency tree for the sentence “*the food was excellent as well as service, however, I left The Four Seasons very disappointed.*” as provided by a Dependency Tree-LSTM. Coloured nodes indicate that the node acting as the sub-tree of that particular parent node are evaluated as being negative (orange), positive (blue), or neutral (grey). Square nodes are potential targets for the root affective label and the dotted arrow separates two affective feature areas as dictated by root verbs.

This project has received funding from the European Unions Horizon 2020 research and innovation programme under the Marie Skłodowska-Curie grant agreement No 721619 (SOCRATES).

In this paper, we present a novel method combining symbolic parse-tree rules with recursive neural sub-tree emotion

recognition to identify and classify affective labels pertaining to specific targets located within utterances. In our approach, we utilize *sub-trees* of verbs to perform affective labelling of noun chunks in the sub-trees. An example of the type of sub-trees provided by verbs and associated targets can be seen in Figure 1, where the diamond shaped nodes are verbs and the dependency parse can be divided up between the verb sub-trees. In doing so, we are able to leverage the robustness of neural models with the structure and context provided by symbolic rules to construct affect-target pairings without being limited to constraints that are harmful to HRI, such as a predefined number of targets that limit robot speech comprehension or requiring extremely specific datasets.

## II. RELATED WORK

Contemporary uni-modal and multi-modal emotion recognition has focused primarily on labelling emotion expressions using either continuous or discrete values with late or early fusion [2]. However, none of these approaches attempt to create associations between sub-parts of language and identified features in a dynamic and proactive manner. Several works have performed Aspect-Based Sentiment Analysis (ABSA), i.e. the correlation of sentiment with some target entity, and often focus on trained approaches toward classifying both aspects and affect [7]–[10]. The limitation of these approaches lies with the non-dynamic determination of targets: if a target does not exist within the set of possible or generated target classes then it cannot be considered or identified. This is a problem in the varied and dynamic scenarios companion robots are expected to work within. Our work bypasses the issue of relying too heavily on a predefined set of targets or domain by instead focusing on syntactic rules to select targets.

Other work also leverages syntactic rules to determine affective targets. Cambria et al. [11] use a semantic-affective network, SenticNet, to aid in the classification of identified targets. In contrast to their work, our approach labels multiple targets and leverages the compositional processing of recursive neural networks to better classify sub-trees and to retain the parse structure of the network for interpretability. Our system also does not require Sentic patterns [5], sentiment specific symbolic rules for language interpretation, as we rely solely on the syntactic parse tree to determine targets and their affective areas of interest. As Tree-LSTMs only require root supervision [6] and we learn composition rules for different affective labels, our approach is suitable for target-affect extraction across a wide range of labels. This allows us to extend our approach to other affective measurements beyond sentiment, such as valence and arousal [12].

In our previous work [13], we had subjects react to audio stimuli and were able to map similar affective reactions to particular stimuli and capture subjective expression styles using an affective association space. We also disambiguated conflicting visual and language expressions by conducting extended dialogue with the subject. While this was an initial first step towards creating a more nuanced understanding of emotion and applying it to a scenario, the learning of user

music preferences and ambiguity resolution, we still lack a more nuanced understanding of what it is about stimuli that the users disliked or liked. We now address this by creating affective associations by giving a system the ability to create affective association hypotheses between targets and recognised affective expressions in natural language.

## III. SYMBOLIC TARGET IDENTIFICATION

To avoid working with an inflexible set of targets, we assume that affective targets can be derived from syntactic patterns in a dependency parse tree. This is similar to previous work which also defined dependency rules for extracting targets [5]. More specifically we aim to identify noun chunks, which function as targets, that exist as descendants of a verb and, based on the verb, we are able to determine the sub-tree containing relevant affective data for a target. Note, that only the closest parent verb of a target is considered to possess the correct sentiment for a target within its sub-tree. This allows for labelling of multiple targets without the need for target supervision, as we use the verbs as the root for identifying affective sub-trees. By labelling the sub-tree of the verb with an affective label, we are able to classify affective labels for multiple different targets, i.e. noun chunks in the verbs sub-tree, with minimal feature overlap between sub-trees.

To obtain dependency parse trees and noun chunks we use the natural language processing (NLP) library SpaCy<sup>1</sup>. Using the dependency parse trees, we identify every verb and obtain the targets associated with that verb’s sub-tree which we use as targets. We define the *sub-sentence* of a sub-tree as the in-order sequence of words that can be constructed from that tree. The algorithm for target selection is defined as follows:

---

### Algorithm 1 Affect Target Identification

---

```

1: doc ← parse(sent) Parse sentence to SpaCy document
2: for word in doc do
3:   if word is verb then
4:     verbs ← verb
5:   end if
6: end for
7: for verb in verbs do
8:   substring ← getSubSentence(verb)
9:   for nounChunk in doc do
10:    if nounChunk in substring then
11:      targets ← nounChunk
12:    end if
13:  end for
14: end for
15: return targets

```

---

### A. SemEval 2016 Dataset

There are few evaluation datasets with the desired affect-target relationship we need that also possess an adequate scope of targets. This is due to many datasets focusing on

<sup>1</sup><https://spacy.io/>

Named Entity Recognition, Topic Recognition, or Aspect Identification to simplify learning and evaluation. We found that the SemEval 2016 Aspect-Based Sentiment Analysis Data [14] is the most appropriate evaluation medium, as it possesses a broad set of aspects with affective sentiment labels.

We extract targets using Algorithm 1 and consider the target correct if the labelled aspect exists as a substring in the extracted noun phrase. As an example, the targets “ball”, “the ball”, and “the red ball” would all be valid extractions for the aspect “ball”. Our symbolic approach is able to identify 1310 aspects out of the possible 1880 (69.7%) located in the SemEval restaurant data. If we are able to correctly identify an aspect in a sentence, then we attempt to correctly label this aspect using the approach described in Section IV.

#### IV. TARGET SENTIMENT LABELLING

##### A. Dependency Tree-LSTMs

To efficiently determine the affect of a verb sub-tree for a particular target, we utilize a Dependency Tree-LSTM [6] trained on the Stanford Sentiment Treebank (SST) [15]. A Tree-LSTM functions, in principle, as the standard LSTM but can take an arbitrary number of inputs at every time-step and processes inputs according to a particular tree-structured format as opposed to in sequence. It also retains the trainable weight matrices  $W$  and  $U$ , as well as a bias vector  $b$ .

A specific node  $j$  in the dependency tree possesses a set of dependent child nodes, denoted  $C(j)$ , which contains the indexes of children  $k$ , with each child possessing a hidden state  $h_k$ . As we are processing dependency trees, the input word vector  $x_j$  is the vector corresponding to the word associated with  $j$  in the parse tree. The summed hidden states  $\tilde{h}_j$  of the children act as the input to the input gate  $i_j$ . The Tree-LSTM has multiple forget vectors  $f_{jk}$  for each of the  $k$  children of a node  $j$ . Aside from the forget vectors, the Tree-LSTM retains traits of a standard LSTM, such as an output gate activation vector  $o_j$  and a memory cell candidate vector  $u_j$ . The updated memory cell state vector  $c_j$  is calculated by summing the elementwise multiplication of  $i_j$  and  $u_j$  with the sum of the elementwise multiplications of each child cell state  $c_k$  with  $f_{jk}$ . The hidden state  $h_j$  is then determined through elementwise multiplication of  $o_j$  and  $\tanh(c_j)$ .

The formalised transition equations for the Dependency Tree-LSTM, as defined in [6], are as follows:

$$\tilde{h}_j = \sum_{k \in C(j)} h_k, \quad (1)$$

$$i_j = \sigma(W^{(i)}x_j + U^{(i)}\tilde{h}_j + b^{(i)}), \quad (2)$$

$$f_{jk} = \sigma(W^{(f)}x_j + U^{(f)}h_k + b^{(f)}), \quad (3)$$

$$o_j = \sigma(W^{(o)}x_j + U^{(o)}\tilde{h}_j + b^{(o)}), \quad (4)$$

$$u_j = \sigma(W^{(u)}x_j + U^{(u)}\tilde{h}_j + b^{(u)}), \quad (5)$$

$$c_j = i_j \odot u_j + \sum_{k \in C(j)} f_{jk} \odot c_k, \quad (6)$$

$$h_j = o_j \odot \tanh(c_j) \quad (7)$$

This approach is able to both learn affective language features from a phrase composition perspective and retain an interpretable parse-tree over how a prediction is reached from constituent elements. This compositional and hierarchical processing of input aids in the understanding of how systems reach particular outcomes, as depicted in Figure 1, wherein we see that the two verbs act as the root for two different sentiment interpretations. Note that in practice we would be able to identify additional targets by adding rules that can handle expressions without verbs, i.e. “Good food.”, however, in this paper, we are only interested in investigating the ability of the system to determine the affective label of a noun chunk existing as a sub-string within the sub-trees we define.

##### B. Evaluating affective features in sub-trees

We adapt a PyTorch implementation<sup>2</sup> based on the original implementation [6] to perform fine-grained sentiment analysis and we retained the hyper-parameters from the original implementation. We retain the hyper-parameters from the original work for the sake of comparability. As such, the Tree-LSTM had an input of size 300, as dictated by the Common Crawl 840B pretrained embedding, a hidden layer of size 168 and an output layer of size 3 with softmax applied for classification.

As benchmarks against the recursive neural approach we chose to use a Bidirectional LSTM model (BLSTM) approach which has been shown to work well on emotion recognition from text [16] and a Logistic Regression Linear Classifier, as linear classifiers have been shown to be competitive with neural approaches [17]. A bag-of-words approach is also less likely to be influenced by sub-sequence data than models normally requiring full sequences. The Logistic Regression approach uses an LMBFGS-solver [18] with a penalty of 12. The BLSTM has a hidden layer size of 64 with a dropout of 0.5 before the final dense layer that applies softmax.

All baseline systems are given the same data as the Tree-LSTM, although they process the data in different non-recursive manners. Our first baseline is a simple Bag-of-Words (BoW) Logistic Regression Classifier, trained on the combination of all sub-trees and labels from the SST, as provided by SST Utils<sup>3</sup>. We also provide a benchmark on the BLSTM that processes the SST sub-trees as sequences of word embeddings [19]. For both the LSTM model and the Tree-LSTM we utilize GloVe word embeddings<sup>4</sup> pretrained on Common Crawl 840B data.

Mapping entire utterances to affective labels has its uses, especially for integration into multi-modal approaches [2], this disregards the semantic context of the language. To determine how affective expressions relate to specific entities in language we need a dynamic method of distributing features, i.e. sub-sequences of text, between identified entities. If we do not distribute features we assume that all entities possess the same affective features, which is not necessarily correct.

<sup>2</sup><https://github.com/tpro1995/TreeLSTMSentiment>

<sup>3</sup><https://github.com/JonathanRaiman/pytreebank>

<sup>4</sup><https://nlp.stanford.edu/projects/glove/>

Commonsense principles dictate that the context in which an entity exists will be local to the position of the entity in a sentence. Both LSTMs and 1D-CNNs with a temporal sliding window use this local context to determine what features to extract. These networks have to learn sentence composition and how to interpret sequences of words. While this gives the networks some flexibility, intermediate interpretations of sequences of words can be unclear and difficult to interpret as the system is working in a high dimensional space.

Tree-LSTM’s recursive and structured processing allows them to learn features from the syntactic structure of language. This allows for the understanding of how sub-tree compositions result in affective labels and it is possible to track the affective reasoning throughout the hierarchical levels of the parse tree. This can be visualised by the coloured nodes in Figure 1 showing how the words “well” and “disappointed” are the beginning of the affective interpretations for the verb roots “was” and “left”. We can also derive that the positive interpretation of the sub-tree “as well as” is primarily based on the word “well”. The final positive interpretation for the entire phrase as depicted by the sentiment value for “was” we can also assume is likely reinforced by the word “excellent”.

The selection of dependency sub-trees is based on the principle that affective expressions in English are often described by linking verbs or copula [20], and we, therefore, divide dependency trees based on their verbs. By selecting the dependency sub-tree associated with a verb we are able to isolate the salient features for the targets within that sub-tree. In Figure 2 we see that, by dividing the tree into two sub-trees with the verbs as roots, we are able to capture relevant affective language features to correctly associate emotions with their respective targets “the food” and “The Four Seasons”.

The process in Figure 2 begins with the sentiment parse tree from the Tree-LSTM in the central box, with blue representing positive, grey neutral, and orange negative sentiment. The dotted edge between the two verbs that act as roots is removed to split the tree into two sub-trees with target noun phrases, represented as square nodes in the tree. This results in the top left and top right sub-trees in Figure 2, that have positive and negative roots respectively. The polarity of these roots are then “trickled down” into identifiable noun phrases, as seen in the bottom left and right sub-trees in the figure. Finally noun phrases inherit the sentiment from the root of the sub-tree, which is determined to be the sentiment for targets in that sub-tree, resulting in the correct polarities for the targets.

To evaluate how well the Dependency Tree-LSTM works against other models when integrated into our system, we feed the determined sub-trees as Bags-of-Words, sentence order sequences, and sub-trees, to a Logistic Regression model, a Bidirectional LSTM model (BLSTM), and the Dependency Tree-LSTM model respectively. Using these pretrained models we determine whether we can identify the correct sentiment label for the targets we have identified within the sub-tree that was processed. As all models are pretrained on the SST, any biases that would impact their performance on the SemEval will be equal for all models. This still allows for the relative

accuracy analysis of sub-tree structured data against other structures. Furthermore, our hybrid method does not focus on the individual statistical performances of the pretrained models on the SST. Instead, we focus on how we can best utilise symbolically determined sub-tree data to identify targets and their affective polarity, with the goal of identifying which model provides the best relative accuracy.

## V. RESULTS

In this section we show the results for the aforementioned models pre-trained on the SST data and evaluated on the SemEval 2016 ABSA restaurant training data [14].

TABLE I  
ACCURACY ON THE STANFORD SENTIMENT TREEBANK FOR POSITIVE, NEUTRAL, AND NEGATIVE SENTIMENT FOR EACH MODEL. NEURAL MODEL ACCURACY IS THE ACCURACY FOR A SINGLE FOLD ON VALIDATION DATA.

Model	Stanford Sentiment Treebank
BoW Logistic Regression	<b>0.855</b>
Bidirectional LSTM*	0.679
Dependency Tree-LSTM	0.849

TABLE II  
ASPECT-BASED SENTIMENT ACCURACY PERFORMANCE OF THE THREE MODELS OVER THE THREE LABELS, PRE-TRAINED WITH THE SST DATASET, ON THE SEMEVAL 2016 ABSA RESTAURANT TRAINING DATA [14] WHEN DISREGARDING SAMPLES WITHOUT ASPECTS, I.E. OUT OF THE CORRECTLY IDENTIFIED TARGETS HOW MANY ARE CORRECTLY LABELLED USING ONLY SUB-TREE DATA.

Model	SemEval 2016 Restaurant Training Data
BoW Logistic Regression	0.584
Bidirectional LSTM*	0.524
Dependency Tree-LSTM	<b>0.669</b>

In Table I, we present a performance comparison between our models on the Stanford Sentiment Treebank dataset [15]. As we can see The Logistic regression model is roughly on par with the Dependency Tree-LSTM, assuming the test set for the neural models is representative. The BLSTM is only trained on the full sentences and root labels of the SST, as it was almost impossible for the network to make use of sub-tree roots due to the high number of redundant samples and class imbalance of the sub-trees [15].

Results in Table II indicate that the Dependency Tree-LSTM outperforms the linear classifier when generalising between the SST and SemEval datasets for association determination. The sub-tree data is more descriptive of the target’s affective label as opposed to the bag-of-words. This is in spite of the logistic regression model slightly outperforming the Dependency Tree-LSTM on the SST, as seen in Table I. The BLSTM was less efficiently able to learn how to transfer features to the substrings when compared to other models on the SST.

### A. Qualitative association behaviour

One of the core goals of our work is not to just be able to create affect-target associations, but also to have some structured method of understanding why a particular association

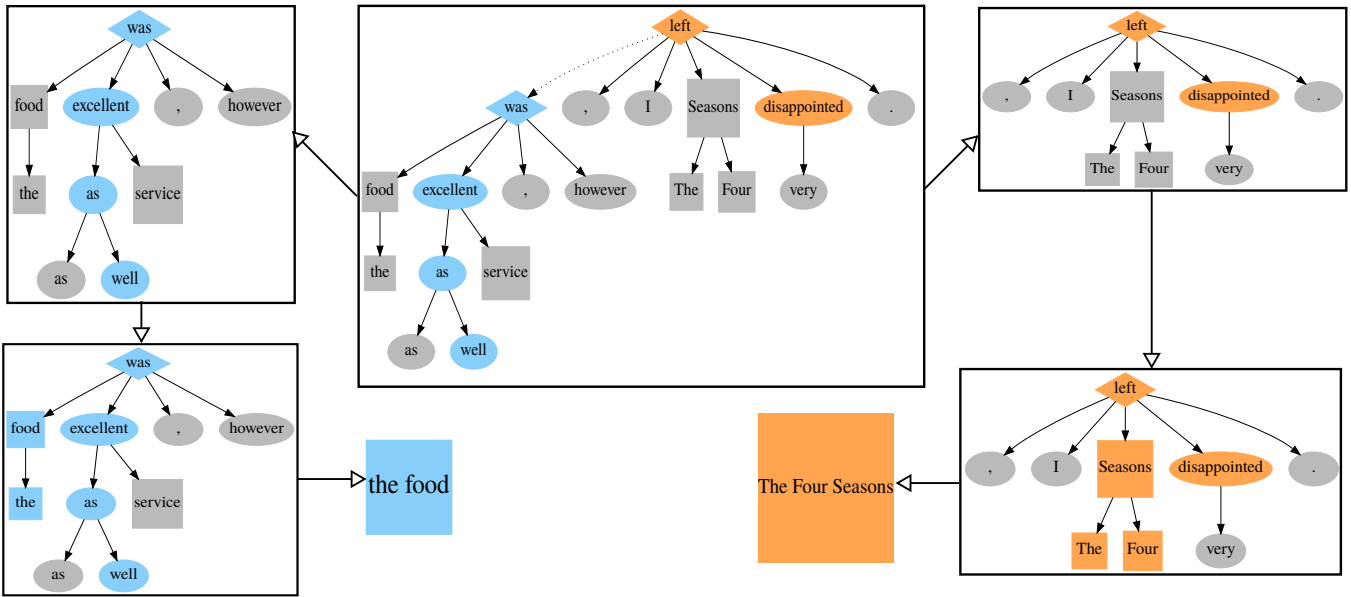


Fig. 2. An example of how the affective parse tree provided by the Dependency Tree-LSTM is deconstructed providing a distribution of language features over the set of local targets determined by splitting the tree based on verb phrases, the splitting point here being depicted by a dotted arrow. To the left, we see how positive sentiment trickles down from the root to charge the noun phrase “the food” and we see how negative sentiment trickles down from the root on the right. Note that although the correct sentiments for “the food” and “The Four Seasons” are extracted, the used SpaCy model does not recognize “service” as a noun chunk and it is therefore not extracted.

was created. The reason for this being that social companion agents should reasonably be able to motivate their ‘train of thought’ regarding a selected action or statement. Take for example Figure 3, where we see that correct sentiment parse for “the food is outstanding.” which correctly determines a positive sentiment for “the food”. Furthermore, if we were to observe the parse before charging “the food” with sentiment we would see that the entire tree is uncharged except for “outstanding” and “is”. This allows us to motivate that “the food” was viewed positively because “outstanding” caused the network to reach that final assertion about the statement.

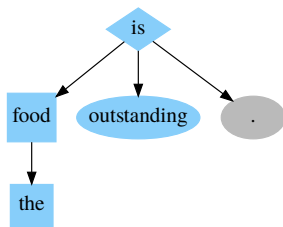


Fig. 3. The affective dependency parse tree for the sentence “the food is outstanding.”, which indicates a positive sentiment for the target “the food”.

In Figure 4 we see a contrast to the previous sentence with “the food is very good, but not outstanding.”. We see the influence of “not” on the word “outstanding”, causing it to be negatively charged. We also see the parent “good” overriding the negative sentiment possessed by “not outstanding” leading to a final positive inference of “the food”. While this is labelled incorrect in the data, the data point describes “food” as neutral, we would argue that there is a positive sentiment towards “the

food”, as it is described as “very good”.

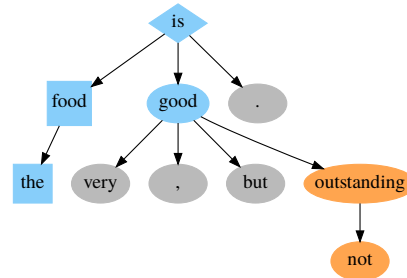


Fig. 4. The affective dependency parse tree for the sentence “the food is very good, but not outstanding.”, as we can see the negative sentiment from “not outstanding” is suppressed by the positive sentiment from the node “good”.

In Figure 5 we see an example of the remaining issue of when none of the children of a particular parent shares the parent’s sentiment. We know that the children together lead to a negative sentiment but none of them directly are responsible. This will need to be dealt with in the future if we wish to motivate how certain affective assumptions are made that are outside the scope of our current work.

## VI. DISCUSSION

An argument against this approach toward the creation of emotion-target associations is that the method could be too inclusive of targets. Our approach creates associations from all noun chunks if they are descended from a verb in the dependency parse tree, which also includes determiners. We argue that having a more comprehensive affective understanding of each target in an expression is beneficial, as it provides

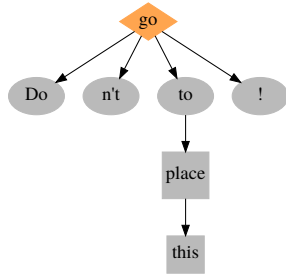


Fig. 5. The affective dependency parse tree as provided by our model for the sentence: “Don’t go to this place” before target association. Our association system will give the correct affective label for “this place” but we are unable to reason as to why this is, as all children are deemed to be neutral.

an intelligent agent with a more nuanced interpretation of the language. Differentiating between affect towards targets as dictated by determiners can provide valuable information to help understand what a user is expressing in context, i.e. talking about the current restaurant owner, “the restaurant owner is rude”, or describing a new desired restaurant owner “it would be good to have a more polite restaurant owner”.

This is not well suited for aspect-based sentiment analysis from the perspective of opinion mining, due to the high recall but low precision provided by the approach. However, it will provide significantly more data to work with for intelligent agents who are given a wide array of signals and are then able to select which of these is most appropriate to broach in dialogue, or otherwise. Unsupervised approaches [8] may still provide a dynamic neural solution, as they are able to move away from a fixed set of targets. However, there is still a lack of evidence these approaches toward aspect extraction can reliably extend to multiple domains at once and not suffer the same problems as a symbolic approach. As such, we believe our hybrid neural-symbolic approach is the most suitable for incorporation into current HRI systems, due to its ability to provide affective labels for a dynamic set of targets without compromising the affective labelling.

## VII. CONCLUSION

In summary, we presented a novel approach of combining the recursive and structured sentiment parse provided by the recursive Tree-LSTM with symbolic rules to perform targeted affect labeling. We identified aspects using our symbolic rule-set for aspect-labelling on the SemEval Restaurant ABSA dataset. Results showed the Dependency Tree-LSTM was best able to utilise the sub-tree features provided by the SST to identify sentiment in sub-trees for individual aspects.

In future work, we intend to extend upon our previous work [13] by incorporating the associations we have extracted into HRI-scenarios, allowing for dialogue decisions based on information garnered by our association system to be able to more directly reference emotions in dialogue.

## REFERENCES

[1] Angelica Lim and Hiroshi G Okuno. A recipe for empathy. *International Journal of Social Robotics*, 7(1):35–49, 2015.

[2] Soujanya Poria, Erik Cambria, Rajiv Bajpai, and Amir Hussain. A review of affective computing: From unimodal analysis to multimodal fusion. *Information Fusion*, 37:98–125, 2017.

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

[4] Erik Cambria, Daniel Olsher, and Dheeraj Rajagopal. Senticnet 3: a common and common-sense knowledge base for cognition-driven sentiment analysis. In *Twenty-eighth AAAI conference on artificial intelligence*, 2014.

[5] Soujanya Poria, Erik Cambria, Grégoire Winterstein, and Guang-Bin Huang. Sentic patterns: Dependency-based rules for concept-level sentiment analysis. *Knowledge-Based Systems*, 69:45–63, 2014.

[6] Kai Sheng Tai, Richard Socher, and Christopher D Manning. Improved semantic representations from tree-structured long short-term memory networks. *arXiv preprint arXiv:1503.00075*, 2015.

[7] Soujanya Poria, Iti Chaturvedi, Erik Cambria, and Federica Bisio. Sentic lda: Improving on lda with semantic similarity for aspect-based sentiment analysis. In *Neural Networks (IJCNN), 2016 International Joint Conference on*, pages 4465–4473. IEEE, 2016.

[8] Ruidan He, Wee Sun Lee, Hwee Tou Ng, and Daniel Dahlmeier. An unsupervised neural attention model for aspect extraction. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, volume 1, pages 388–397, 2017.

[9] Devamanyu Hazarika, Soujanya Poria, Prateek Vij, Gangeshwar Krishnamurthy, Erik Cambria, and Roger Zimmermann. Modeling inter-aspect dependencies for aspect-based sentiment analysis. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, volume 2, pages 266–270, 2018.

[10] Yukun Ma, Haiyun Peng, and Erik Cambria. Targeted aspect-based sentiment analysis via embedding commonsense knowledge into an attentive lstm. In *Proceedings of AAAI*, 2018.

[11] Erik Cambria and Amir Hussain. *Sentic computing: a common-sense-based framework for concept-level sentiment analysis*, volume 1. Springer, 2015.

[12] Sven Buechel and Udo Hahn. Emobank: Studying the impact of annotation perspective and representation format on dimensional emotion analysis. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers*, volume 2, pages 578–585, 2017.

[13] Henrique Siqueira, Alexander Sutherland, Pablo Barros, Matthias Kerzel, Sven Magg, and Stefan Wermer. Disambiguating affective stimulus associations for robot perception and dialogue. In *IEEE-RAS International Conference on Humanoid Robots (Humanoids)*, pages 433–440, Nov 2018.

[14] Maria Pontiki, Dimitris Galanis, Haris Papageorgiou, Ion Androutsopoulos, Suresh Manandhar, AL-Smadi Mohammad, Mahmoud Al-Ayyoub, Yanyan Zhao, Bing Qin, Orphée De Clercq, et al. Semeval-2016 task 5: Aspect based sentiment analysis. In *Proceedings of the 10th international workshop on semantic evaluation (SemEval-2016)*, pages 19–30, 2016.

[15] Richard Socher, Alex Perelygin, Jean Wu, Jason Chuang, Christopher D Manning, Andrew Ng, and Christopher Potts. Recursive deep models for semantic compositionality over a sentiment treebank. In *Proceedings of the 2013 conference on empirical methods in natural language processing*, pages 1631–1642, 2013.

[16] Yoon Kim. Convolutional neural networks for sentence classification. *arXiv preprint arXiv:1408.5882*, 2014.

[17] Hendrik Schuff, Jeremy Barnes, Julian Mohme, Sebastian Padó, and Roman Klinger. Annotation, modelling and analysis of fine-grained emotions on a stance and sentiment detection corpus. In *Proceedings of the 8th Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis*, pages 13–23, 2017.

[18] Richard H Byrd, Pei Huang Lu, Jorge Nocedal, and Ciyou Zhu. A limited memory algorithm for bound constrained optimization. *SIAM Journal on Scientific Computing*, 16(5):1190–1208, 1995.

[19] Jeffrey Pennington, Richard Socher, and Christopher Manning. Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 1532–1543, 2014.

[20] Aneta Pavlenko. Emotions and the body in russian and english. *Pragmatics & Cognition*, 10(1):207–241, 2002.