Emotion Carrier Recognition from Personal Narratives

Aniruddha Tammewar, Alessandra Cervone, Giuseppe Riccardi

Signals and Interactive Systems Lab, University of Trento, Italy
{aniruddha.tammewar, alessandra.cervone, giuseppe.riccardi}@unitn.it

Abstract

Personal Narratives (PN) - recollections of facts, events, and thoughts from one’s own experience - are often used in everyday conversations. So far, PNs have mainly been explored for tasks such as valence prediction or emotion classification (e.g. happy, sad). However, these tasks might overlook more fine-grained information that could prove to be relevant for understanding PNs. In this work, we propose a novel task for Narrative Understanding: Emotion Carrier Recognition (ECR). Emotion carriers, the text fragments that carry the emotions of the narrator (e.g. loss of a grandpa, high school reunion), provide a fine-grained description of the emotion state. We explore the task of ECR in a corpus of PNs manually annotated with emotion carriers and investigate different machine learning models for the task. We propose evaluation strategies for ECR including metrics that can be appropriate for different tasks.

Index Terms: emotion, computational paralinguistics

1. Introduction

A Personal Narrative (PN) is a recollection of events, facts, or thoughts felt or experienced by the narrator. People tell PNs in the form of stories to themselves and to others to place daily experiences in context and make meaning of them [1]. Rich information provided through PNs can help better understand the emotional state of the narrator, thus PNs are frequently used in psychotherapy [2]. Often, in psychotherapy sessions, clients are invited by therapists to tell their stories/PNs [3]. Through PNs, clients provide therapists with a rough idea of their orientation toward life and the events and pressures surrounding the problem at hand [3]. Nowadays, well-being applications are widely used, making the collection of PNs easier in the form of a journal from clients in digital form.

Automatic Narrative Understanding (ANU) is a growing field of research that aims at extracting different important and useful information from Narratives for target applications [4]. Examples of tasks include reading comprehension [5], summarization [6], and narrative chains extraction [7]. However, although PNs are widely used in psychotherapy, very few ANU tasks have been proposed to analyze PNs from the perspective of well-being.

A deep emotion analysis of the PNs is possible with ANU. For example, [8] work on predicting valence from the PN and found how different text fragments, including not only sentiment words but also words referring to events or people, proved to be useful to predict the valence score of a narrative for machine learning models. Following up on this analysis, [9] propose emotion carriers (EC) as the concepts from a PN that explain and carry the emotional state of the narrator. ECs thus include not only explicitly emotionally charged words, such as “happy”, but also mentions of people (friends), places (school, home), objects (guitar), and events (party) that carry an emotional weight within a given context. For example, in the narrative in Table 1 the word “family” (orig. “Familie”) is carrying a positive emotion for the narrator. [9] propose an annotation schema for ECs and perform an annotation experiment of German PNs from the Ulm State-of-Mind in Speech corpus [10]. Based on the analysis of the inter-annotator agreement, they find the task complex and subjective. Nevertheless, the task has potential benefits in many applications, including well-being.

Automatic recognition of ECs could support Conversational Agents (CA) in natural language understanding and generation. As illustrated in Figure 1, a CA could use ECR output to ask follow-up questions based on the recognized ECs. In the example, the user utterance can be considered as a short PN expressing negative sentiment. With current emotion analysis systems, it would be possible to understand that the user is upset about something. This information could be used for example to form a response showing empathy toward the user, e.g. the first part of the response ‘Sorry to hear that.’. In addition to emotion categorization, EC recognition (here, the identified ECs include ‘a fight’, ‘my boss’, and ‘work’) would allow to ask follow-up questions (e.g. ‘Can you tell me ...’) to better understand the relationship between an EC and the narrator’s emotional state.

The contributions of this paper are threefold. First, we propose Emotion Carriers Recognition (ECR), a novel task of narrative understanding. Second, to analyze the feasibility of automation of the task, we present different baseline models for ECR relying on bi-LSTM based architectures on an EC annotated corpus [9]. Third, we propose different evaluation strategies for ECR and compare the model’s performance to the inter-annotator agreement obtained from humans.

2. Related Work

An interesting ANU task, relevant for emotion analysis, is valence (emotional value associated with a stimulus) prediction from spoken PNs [11, 8]. Another task from emotion analysis that can be borrowed to ANU is Emotion Cause Extraction (ECE), aimed at identifying what might have triggered a particular emotion in a character from a narrative or the narrator himself.
were specifically trained to perform the task. The annotation written texts such as narratives and blogs [17, 18] (eg. ‘I broke the PNs are in German and taken from the Ulm State-of-Mind

Table 1: A small text fragment from a PN annotated with emotion carriers. The first row reports the original German words from the PN, the second row shows the corresponding English translation, while the third row shows the annotations. The annotation is performed by 4 annotators, thus for each token, there are 4 IO labels. For the token “Familie” the annotation is I/I/O, which means that the first three carriers classified as I, while the forth as O. The intensity of the red color in the background for the PN fragment also highlights the number of annotators who annotated the token (from lightest for I annotator to the darkest for all annotators).

<table>
<thead>
<tr>
<th>PN fragment:</th>
<th>Und</th>
<th>ahm</th>
<th>die</th>
<th>Gefühle</th>
<th>dabei</th>
<th>waren</th>
<th>dass</th>
<th>man</th>
<th>sich</th>
</tr>
</thead>
<tbody>
<tr>
<td>Annotation:</td>
<td>0/0</td>
<td>0/0</td>
<td>0/0</td>
<td>0/0</td>
<td>0/0</td>
<td>0/0</td>
<td>0/0</td>
<td>0/0</td>
<td>0/0</td>
</tr>
</tbody>
</table>
| Translation: And um, the feelings were that you are just pleased and happy that you have a family ...

In the preprocessing step, we first perform tokenization using the spaCy toolkit [19]. Next, we remove punctuation tokens from the data. Based on initial experiments, we found that removing punctuation helps improve the performance of the models. The number of annotations (ECs) identified by the annotators per narrative varies from 3 to 14 with an average of 4.6. On average, the number of tokens per EC consists of 1.1 tokens for three annotators, while the fourth annotator identified longer segments consisting of 2.3 tokens (avg.). On average, a narrative consists of 704 tokens, while a sentence consists of 22 tokens. The sentence splitting is performed using the punctuation provided in the original transcriptions.

We find that only 7.3% of the tokens are assigned the label I by at least one annotator. This shows that the classes I and O are highly imbalanced, which could result in inefficient training of the models. With further analysis, we notice that only 34% of the total sentences contain at least one EC, while the remaining 66% sentences do not contain any carrier marked by any annotator.

### 4. Task Formulation

We pose the recognition of emotion carriers from a given PN as a sequence labeling problem. The final goal is the binary classification of each token into classes I or O. As seen in Table 1, the task of selecting EC text spans is subjective. Each annotator has a different opinion toward the spans to be selected as ECs, so it is challenging to identify an annotation as valid or invalid. For each token, we have annotations from four annotators with IO labels. Some annotators may agree on the annotation, but it is infrequent that all four annotators annotate the token as I. In the example from Table 1, the token “glücklich” is annotated I by two annotators, three annotators agree that the token “Familie” is an EC, while a few tokens are marked as EC by only one annotator. In this scenario, it is difficult to provide a hard I or O label. To tackle this problem, we model the problem of Emotion Carrier Recognition (ECR) as providing scores to the label I, representing the likelihood that that token is a part of an EC. Label distribution learning (LDL) [20] can effectively capture the label ambiguity and inter-subjectivity within the annotators. We use LDL with a sequence labeling network and the KL-Divergence loss function. The advantage of LDL is that it allows to modeling the relative importance of each label. For evaluation, we use different strategies to select the final IO labels.

### 5. Model

We use sequence labeling architecture relying on biLSTM with attention, similar to [21]. As shown in Figure 2, the input text is first passed through the embedding layer to obtain the word embedding representation for each token. We use 100-dimensional pre-trained GloVe [22] embeddings. To encode the sequence information, we then use two stacked bidirectional LSTM layers.
with a hidden size of 512. We also use attention mechanism [23] along with the bi-LSTM, where attention weights $a_i$ represent the relative contribution of a specific token to the text representation. We compute $a_i$ at each output time $i$ as follows:

$$a_i = \text{softmax}(v^T \tanh(W_i h_i + b_i))$$

(1)

where $h_i$ is encoder hidden state and $v$ and $W_i$ are learnable parameters of the network. The output $z_i$ is the element-wise dot product of $a_i$ and $h_i$.

Finally, the output is passed through the inference layer consisting of two fully connected layers with 50 units each, and a softmax layer to assign probabilities to the labels for each word. We also use layer normalization and two dropout layers with a hidden size of 512. We also use attention mechanism [23] along with the bi-LSTM, where attention weights $a_i$ represent the relative contribution of a specific token to the text representation. We compute $a_i$ at each output time $i$ as follows:

$$a_i = \text{softmax}(v^T \tanh(W_i h_i + b_i))$$

(1)

where $h_i$ is encoder hidden state and $v$ and $W_i$ are learnable parameters of the network. The output $z_i$ is the element-wise dot product of $a_i$ and $h_i$.

During training, we use the Kullback-Leibler Divergence (KL-DIV) as the loss function [25] and the Adam optimizer [26] with the learning rate of 0.001.

### 6. Experiments

As described in Section 3, there is class imbalance in the data. The class I tokens are very infrequent compared to the class O. This may result in a bias toward class O in the classifier. Another problem we have to deal with is the length of the narratives. The narratives are very long with an average length of 704 tokens. Standard machine learning and bi-LSTM based architectures are not efficient in dealing with very long contexts.

To address these challenges, we experiment with different levels of segmentation of the narratives and apply strategies to select proper train and test sets. We train and test the sequence-labeling models at narrative and sentence levels. In the narrative level, we consider the entire narrative as one sequence. In the sentence level, we consider one sentence as a sequence. In this way, we analyze how the length of a sequence affects the performance of the model. Also, note that at the sentence level, the model does not have access to other parts of the narrative. We study how limited access to context affects performance.

The sentence-level sequences are further considered in two ways: 1) **SentAll**: all the sentences are considered 2) **SentCarr**: only sentences containing at least one EC are considered. SentCarr reduces the class imbalance as we remove all sentences that do not contain any token tagged as class I. In a real-world scenario, we would have to extract carriers from the entire narrative or all the sentences, as we do not know beforehand which sentences contain the carriers. Thus, we use SentCarr only for training, but in the test set SentAll is used.

We also experiment with another sequence labeling model based on Conditional Random Fields (CRF) [27], a widely used machine learning algorithm for sequence-labeling problems in NLP, such as Part of Speech tagging. For the CRF model, we use the context window of ±3 with features such as the token, its suffixes, POS tag, prefix of POS tag, sentiment polarity.

### 7. Evaluation

In this section, we propose different evaluation strategies for the ECR task. Note that even though our model is trained to predict the probability distribution of the classes, our final goal is to assign one of the two classes from I and O. For all evaluations, for the ground truth, we consider that a token is annotated (i.e. I in the IO tags) if at least one of the annotators has annotated it. Similarly, for the output, we consider the output as I if the probability assigned crosses the minimum threshold of 0.25, which is equivalent to one of four annotators tagging the token as I. In all evaluations, we do five-fold cross-validation with the leave one group (of narrators) out (LOGO) strategy. For each training session, we split the data into train, dev, and test sets without any overlap of narrators in the three sets.

#### 7.1. Token Level

The token level evaluation measures the performance of predicting I or O class for each token in a sequence. We use this metric to evaluate our models with different data segmentations. We are concerned more about the prediction of the class I, as we are interested in applications of ECR such as Figure 1, where it is important to find one or more important carriers to start a conversation with the narrator. Thus, we show the F1 score of class I and weighted average (micro) of F1 score of I and O.

As discussed earlier, considering a real-world scenario, we need the model to perform well on the SentAll or Narrative data. In Table 2, we find that the model trained on SentCarr performs best on the SentAll. For further evaluation we use this model, thus recognition would be done on the sentences and not the entire narrative at once. Note that the performance of the Narrative strategy is only slightly worse, suggesting that the task is not affected much by the length of the context available. The CRF model is the worst performing one.

Using the SentCarr model, we extract the continuous sequences of tokens that are tagged as I. These text spans are considered as the ECs recognized by the model. In the metrics in the next section, we evaluate the model by comparing this set of carriers with the set of manually annotated reference carriers.

#### 7.2. Agreement Metrics

We evaluate the performance of the models using the metrics that were used to evaluate the inter-annotator agreement between the four annotators (pair-wise) in [9], based on the positive (specific) agreement [28]. This evaluation is important as it compares the performance of the system with the inter-annotator agreement,
While both strategies improved the scores, the improvement with which can loosely be considered as human performance. (strategy b) and another is to use the partial match (strategy c). Evaluation based on the agreement metrics (position considered(T), agnostic(F)); (lexical level: token, lemma) (w/ stopwords)

Table 3: Evaluation based on the agreement metrics (positive agreement) with different parameter configurations. For each configuration, the corresponding inter-annotator agreement (IAA) score is in the last column (in terms of F1 score). Parameters: (Matching strategy: Exact, Partial); (Position agnostic(F)); (lexical level: token, lemma)

<table>
<thead>
<tr>
<th>sr</th>
<th>Parameters</th>
<th>Prec(std)</th>
<th>Recall(std)</th>
<th>F1(std)</th>
<th>IAA (F1)</th>
</tr>
</thead>
<tbody>
<tr>
<td>a</td>
<td>Exact, F, token</td>
<td>32.6(3.3)</td>
<td>52.1(3.6)</td>
<td>40.0(2.9)</td>
<td>25.2</td>
</tr>
<tr>
<td>b</td>
<td>Exact, F, token (w/ stopwords)</td>
<td>42.3(4.6)</td>
<td>67.2(4.2)</td>
<td>51.7(4.0)</td>
<td>NA</td>
</tr>
<tr>
<td>c</td>
<td>Partial, F, token</td>
<td>37.4(4.0)</td>
<td>51.3(0.4)</td>
<td>43.1(2.7)</td>
<td>32.0</td>
</tr>
<tr>
<td>d</td>
<td>Partial, F, token</td>
<td>59.4(5.5)</td>
<td>83.6(4.7)</td>
<td>69.2(3.5)</td>
<td>39.9</td>
</tr>
<tr>
<td>e</td>
<td>Partial, F, lemma</td>
<td>61.8(6.4)</td>
<td>86.5(4.2)</td>
<td>71.8(4.3)</td>
<td>40.3</td>
</tr>
</tbody>
</table>

which can loosely be considered as human performance.

We also explore the different criteria to decide whether two spans match or not, as used in the original metrics. The evaluations are based on Exact Match, where the two carriers match if they are exactly the same and Partial/soft Match, where the token overlap of the two carriers w.r.t the reference carrier is considered. Other parameters in the matching criteria include position considered of the carrier in the narrative, which has two possibilities for matching spans have to be at the same position. Another criterion is based on the matching of tokens vs lemmas. We remove stopwords from the annotation as we are interested in the content words.

Results: Table 3 summarizes the evaluation using the Agreement metrics. As expected, with the loosening of the matching criteria, the results improve. A similar trend is observed in the model’s output showing the model’s predicted score and ground truth probabilities for each token. In most cases, the probability distribution in the model’s output seems to follow similar trends to that of the ground truth probabilities. We also observe frequent cases of false positives, where the model assigns a high probability to class I even when the ground truth label is O, as can be seen in the third example. This behavior could be a result of training the model at the sentence level, thus the position in the narrative is not taken into consideration, resulting in recognition of multiple occurrences of the same carrier. Additionally, the performance further improves when we match lemmas instead of tokens (from d to e).

Analysis: Figure 3 shows example sentences from a test set with a heatmap showing the model’s predicted score and ground truth probabilities for each token. In most cases, the probability distribution in the model’s output seems to follow similar trends to that of the ground truth probabilities. We also observe frequent cases of false positives, where the model assigns a high probability to class I even when the ground truth label is O, as can be seen in the third example. This behavior could be a result of training the model at the sentence level with the SentCarr strategy, where all the sentences in the training set contain at least one EC, biasing the model towards that distribution.

To study if the model is biased towards the recognition of ECs with sentiment words (angry, joy) versus content words (internship, parents) in ECs, we find the sentiment and content carriers using the polarity score assigned by the textblob-de library. We find the mean of fraction of content ECs per narrative to be 60% in the reference while 64% in the prediction, suggesting a minimal bias in the model.

8. Conclusions

We proposed Emotion Carriers Recognition (ECR), a novel task to recognize the text spans that best explain the emotional state of the narrator from personal narratives. The proposed task allows to have a fine-grained representation of the emotional state of the narrator by recognizing relevant text fragments, including mentions of events, people, or locations. We presented different baseline models to address the task, and evaluated them using both token-level and agreement metrics. We compared our best model performance with the inter-annotator agreement and found that our model agrees well with the annotators. We believe this task could be useful as a first step towards a richer understanding of emotional states articulated in personal narratives. As future work, we plan to investigate the integration of an ECR model into a conversational agent, supporting well-being.

9. Acknowledgements

The research leading to these results has received funding from the European Union – H2020 Programme under grant agreement 826266: COADAPT.

---

1https://tinyurl.com/textblob-de
10. References


