Understanding Complex Emotions in Sentences

Stanford CS224N Custom Project

Hannah Levin* **Samir Agarwala*** **Juan Triana***
Department of Computer Science
Stanford University
{levinh, samirag, juan3112}@stanford.edu

Abstract

Humans express complex emotions and often even in a single sentence, a person might express several different emotions. Designing machine learning systems that can deconstruct sentences with complex emotions into segments is a challenging task where there is relatively limited work. Work in this area can enable progress in various areas ranging from the training of psychotherapists to personalized education. In our work, we first generate a large scale dataset with 3,907 sentences in total across our training, validation and test sets by in-context prompting of the LLAMA-2 70B model and several post-processing steps. To the best of our knowledge, this is the first large-scale dataset for emotion segmentation of multiple emotions in sentences. We then propose using a pretrained sentence encoder BERT model along with a bidirectional LSTM model for the challenging task of sentence segmentation. Our proposed approach outperforms our baselines with a mean segmentation IoU of 0.9230 and an emotion classification accuracy of 0.5021 on classifying 5 distinct emotion classes. The large scope for improvement in the emotion classification task indicates the need to further study this area to understand how humans express emotions and develop models that can understand complex emotions in language.

1 Key Information to include

- Contributions: Samir worked on setting up the data loading and training code, n-gram baselines, regression architectures and balancing the data distribution. Hannah worked on prompt engineering, LLM models testing, generating the data, and post processing it consistently for use by our model. Juan worked on supporting the creation of the evaluation code, LSTMs bidirectional architectures, data analysis, and aided in the extraction of evaluation metrics. ALL contributed to writing and helpful discussions.

2 Introduction

Humans have complex emotions and a single thought is often not just composed of one emotion. Paul Ekman defines the six basic emotions: anger, disgust, fear, happiness, sadness, and surprise (Ekman, 1992). A study of human narratives reveals how dynamic emotions are and how they should be studied as changing time-series data (Ong et al., 2021). Humans naturally express themselves with mixed emotions and our work offers a more realistic context to NLP models. Furthermore, emotion is not often a set length and can be expressed from as little as one word to many words. The challenge lies in models to adapt in identifying these segment changes in emotion.

While current research has accomplished assigning an emotion token to an entire sentence, our model is able to break down a given sentence or thought into segments expressing one to three unique emotions. This paper provides novel work in emotion detection and recognition among complex

*Equal contribution

thoughts with multiple emotions which better reflect natural human dialogue. Current research demonstrates promise in detecting a single emotion from a given sentence spoken by the patient using a bot assistant (Poria et al., 2019). Our work expands on this to allow for better and more complex content for therapist bots to better understand emotions expressed by patients.

While there are many other emotions, the six established by Ekman are the basis for other emotions and each have distinct facial expressions associated with them. Building upon the relationship between detected emotions and facial expressions, future work could include creating a psychotherapist training simulator with a generated avatar of a patient which can more easily extract emotion from text and smoothly transition facial expression that is consistent with the expressed text dialogue. There is much potential to integrate this work in such settings.

3 Related Work

Much of current work in the sentiment analysis space focuses on classifying text as positive or negative (Medhat et al., 2014), rather than using fine grained categories of emotions that more fully encapsulate the complexities of human expression. Recent models have accurately identified the emotion token for given sentences; one example is a CRF-based machine learning approach that assigns one of the six Ekman emotion tokens, along with <Neutral>, to a given sentence but can only achieve a satisfied word level emotion tagging accuracy of sentences with at most 8 words (Das and Bandyopadhyay, 2009). The study further implements word level emotion tagging, however its corpus is English SemEval 2007 Affect Sensing which is exclusively news headlines and not applicable to natural dialogue with conversational agents. Furthermore, news headlines are snippets and not as complex as human expression.

The research done on complex emotions is limited, especially in time series data. A study with a datset of complex emotions was annotated on a scale of very positive to very negative sentiment rather than using the specific emotions (Ong et al., 2021). Oftentimes, a change in sentiment can be detected through use of conjugations such as "but" or "and", but as thoughts become more complex these hints for a switch or continuation of an emotion in time series data may go unnoticed by models without proper training.

Other works have attempted to identify even more fine grained emotions such as admiration and confusion (Alon and Ko, 2021). They do so by mapping 27 emotions to the six basic Ekman emotions. However, the model assigns a single emotion to an entire example and does not explore segmentation of text into multiple emotions as we do in our work.

4 Dataset

We could not find an appropriate dataset and annotations for our task of complex emotion detection. In short, the task is to take the input of a given sentence and output the sentence divided into segments of unique emotions with the accurate emotion labeling. The input is a sentence and the output is that same sentence divided into segments where each segment is labeled with an emotion. All valid emotions among the annotations include Ekman's basic emotions {`ANGER, FEAR, HAPPY, SAD, SURPRISE`}, excluding {`DISGUST`} as explained in future sections, and an added {`NEUTRAL`} token for any segment that does not fall under any of those emotions.

Many common datasets are composed of short snippets of text such as news headlines and do not reflect natural human dialogue. Other emotion annotated datasets only provided a single emotion per example and the sample sentences themselves were not as complex as we would have wanted to use. One of the largest fully annotated English language fine-grained emotion datasets of 58,000 examples is composed of English Reddit comments and labeled with 27 emotion categories (Alon and Ko, 2021). However, each sentence is assigned only a single emotion. Furthermore, the nature of the data being a comment on a post tends to be not as complex or as long as what we were looking to test in this paper.

Instead, we generated our own dataset using the LLAMA-2 model with 70 billion parameters (Touvron et al., 2023) The dataset includes sentences with 1-3 unique emotions and their annotations of the divided segments. Current literature demonstrates the importance of including input-related context into LLM prompts (Tang et al., 2022) and show potential in accuracy in annotating text for

emotion recognition (Latif et al., 2023). Targeted in-context examples of sentences with 1-3 segments and ranging emotions were used in prompting for data generation. We provide the prompt given to LLAMA-2 in Appendix D. We prompted LLAMA-2 for 10 examples per call, and removed samples if they contained invalid emotions or other errors including a inconsistencies between the annotations or duplicate sentences.

The combined generated dataset has 3,907 samples with 75% data towards training, 5% for validation and 20% for testing. In the test set, there are a total of 783 full sentences, divided into 1,428 segments. Among these segments, 97 originated from data consisting of three segments each, 451 from data containing two segments, and 235 from data consisting of just one segment. See Figure 1 for the distribution of emotions and number of segments across our data. The average number of words per sentence is about 13 words.

4.1 Dataset Generation

Many other popular LLM models available on Together API were tested but the LLAMA-2 70B model was ultimately selected as the best option for our task. Other models such as Alpaca (Touvron et al., 2023) did not follow directions given in the prompt and consistently annotated using emotions outside of the allowed emotions instructed in the prompt. Adjusting temperature scores for the model to decrease for "creativity" in emotion types did not resolve this issue and instead led to more differences between wording in the original sentence and the annotated one, requiring it to be discarded from the dataset. Furthermore, many models other than LLAMA-2 did not output consistent formatting of the emotion annotations of sentences. This made it impossible to extract the emotions and segments due to different formatting each time across batches.

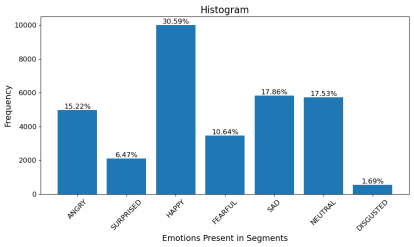
Another issue was the large number of duplicates (and sometimes relatively similar sentences) produced in our dataset. To address this, we replaced the in-context examples in the prompt during each batch with the generated examples from the previous batch. However, this proved ineffective because the model would learn from bad examples generated and recursively generate more bad examples. We then tried to increase the repetition penalty in hopes of having more diversity in the examples generated. However, this again resulted the sentence and concatenated segments from annotations to not be the same. Thus, we decided to keep the in-context examples in our prompt the same across all batches to maintain consistency in quality, generated more data, and removed duplicate examples to mitigate the issue.

Despite prompting for over 50,000 examples, after post processing we had a total of 16,944 examples and a large imbalance of emotions and number of segments in the distribution as seen in Figures 1a and 1c. In particular, examples with three segments and the emotion {DISGUST} was significantly underrepresented. The emotion distribution was much more consistent across emotions after applying techniques described in the next section. However, the representation of the emotion {DISGUST}, particularly in examples with 3 segments, remained significantly lower than the others. In an attempt to generate more examples of "disgust" for our dataset to equalize the distribution, we used more targeting prompting with "disgust" in-context examples but LLAMA-2 reduced the quality of the generated text overall; the generated data incorrectly annotated emotions such as by confusing segments that should be classified as {SAD} instead as {DISGUST}.

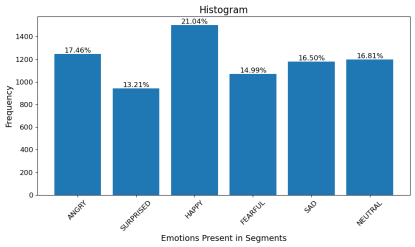
A more equitable distribution was chosen by removing the emotion {DISGUST} from our dataset to allow for a more fair comparison and an equalized distribution of emotions. We had to make a tradeoff between accuracy of the ground truth annotations and making up for underrepresented data. One model that performed well on generating more targeted requests such as the emotion {DISGUST} with 3 segments was GPT-4. Future work should utilize GPT-4 to generate data in underrepresented areas and better equalize the distribution. Due to cost constraints, we were unable to test paid models such as GPT-4 in depth for this paper.

4.2 Balancing Data Distribution

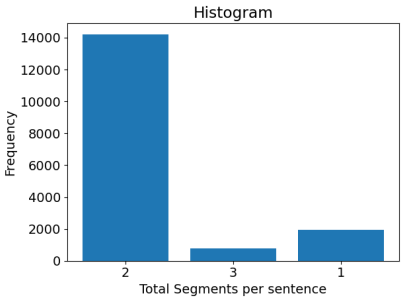
After generating our raw dataset by prompting LLama-2, we noticed that there was an imbalance in the distribution of the different emotions in the dataset. Thus, as a post-processing step we explored ways to balance the data to get a more equitable distribution of each emotion class for segments of each possible size. For this, we processed segments of different lengths independent of one another. For segments of each length, we first computed the frequency of each emotion $\forall_{\text{emt}} \quad N_{\text{emt}}$ across



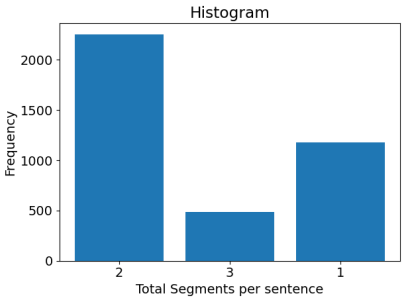
(a) Emotion distribution in initial data



(b) Emotion distribution in final data



(c) Number of segments in initial data



(d) Number of segments in final data

Figure 1: This figure compares the distribution of the original post-processed data (left) with the final balanced dataset that we use in our project (right). The histograms show the combined distribution of the train, validation and test splits.

segments of a given length. We also compute a certain quantile of data as N_{keep} or the target number of sentences with a given emotion using the computed counts of the distinct emotions in our data. The quantile we choose for each segment length is a hyperparameter based on the emotion distribution in the retained data. The probability of keeping a sentence with a single emotion is computed as $P_{\text{keep-emt}} = \frac{N_{\text{keep}}}{N_{\text{emt}}}$. If a sentence has multiple emotions, we consider the occurrence of emotions as independent events for simplicity without considering conditional probabilities, and keep the sentence in our final dataset with a probability of $P_{\text{keep}}(\text{sentence}) = \Pi_{\text{emt} \in \text{ sentence}} P_{\text{keep-emt}}$. Thus, our method of balancing the data distribution takes into account the occurrence of emotions and gives a more balanced data distribution. Figure 1 compares the distribution of the original generated data with the final balanced data and we provide the emotion distribution by number of segments in appendix C.

5 Approach

In this section, we discuss our proposed approach for deconstructing emotion in complex sentences. We first discuss the model architecture and then inference behavior.

5.1 Post-processing

Post processing steps included standardizing the text using lowercase, removing punctuation, and expanding contractions. Examples with duplicate emotions (such as two segments of happy) were removed, however, this is an area we do encourage future work to examine. Examples generated with emotions other than the predefined ones (such as "hopeful" or "excited") were removed. Examples with a mismatch in wording between the sentence and concatenated segments from the annotations of that sentence were also removed.

5.2 Model Architecture

Our model consists of three main components: a pretrained and frozen sentence encoder, a bidirectional LSTM that aggregates the token embeddings and learns task-specific information and lastly a

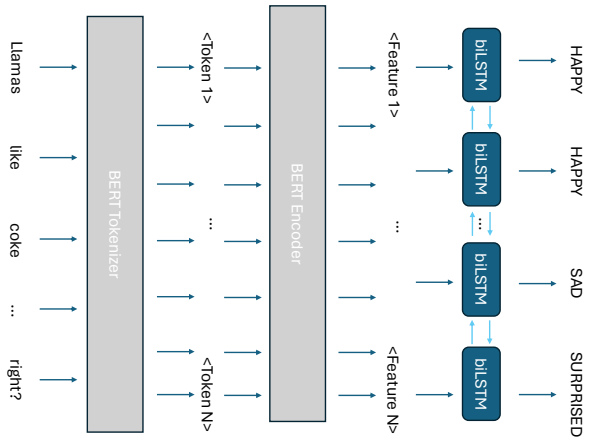


Figure 2: We approach the challenging problem of emotion segmentation in a sentence by encoding sentences using a pretrained uncased BERT encoder (Devlin et al., 2018) and then using a 4-layer bidirectional LSTM to predict a per-token emotion during training. These per-token emotions are used to segment sentences into parts with different emotions.

linear layer that takes in the refined embeddings from the LSTM network and maps it to a per-token emotion class. Figure 2 provides an overview of our model architecture.

For our sentence encoder, we use a pretrained and frozen BERT encoder (Devlin et al., 2018). Due to computational constraints, we do not finetune BERT and instead use the BERT tokenizer and encoder off-the-shelf to tokenize sentences and get per-token encodings. The BERT encoder is pretrained to learn linguistic features and relationship between sentences, however is not explicitly trained to encode emotion information.

We train a 4-layer bidirectional LSTM (Graves and Schmidhuber, 2005) network on top of the BERT encodings to learn the task-specific information required to classify emotions in the input tokens, and we specifically choose a bidirectional network since it aggregates information across the entire sentence and thus allows each token to have access to information from the rest of the sentence when predicting its emotion. This is especially important for the emotion segmentation task since words next to each other are likely to have a similar emotion, and having neighboring context can help predict the emotion correctly for words that otherwise do not express an emotion. The last component of our network is a linear layer that takes in the per-token output of our LSTM network and maps it to an emotion class which is the final output of our model.

5.3 Inference

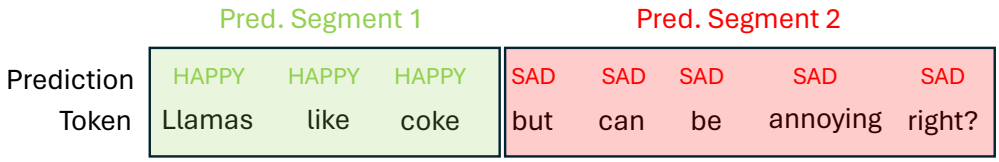


Figure 3: During inference, our model predicts a per-token emotion. We use the per-token emotion annotation to segment the sentence into different parts i.e. consecutive tokens that have the same predicted emotion belong to the same segment. Thus, our simple per-token classification approach allows us to perform sentence-level emotion classification.

The task we are approaching is emotion segmentation i.e. we want to segment sentences into multiple emotions. We use a simple heuristic to go from the predicted per-token emotion prediction to segmented emotions wherein we consider consecutive tokens with the same predicted emotion as a predicted segment. Figure 3 provides an example of this process.

6 Experiments

In this section, we first introduce our baselines and evaluation metrics in sections 6.1 and 6.2 respectively. We then compare the performance of our proposed approach against several baselines in section 6.3. We also perform an ablation study examining the potential benefit of using a bidirectional LSTM model against a unidirectional model in section 6.4. Lastly, we perform a qualitative analysis of our proposed model on our test in section 6.5.

6.1 Baselines

We consider several methods in addition to the proposed approach described in section 5. Our baselines include n-gram and neural network models built on top of a frozen BERT encoder similar to our proposed approach for fair comparison.

N-gram models. We implemented 1-gram, 2-gram and 3-gram models as baselines. These models do not require training and instead divide each input sentence into n equal sized and consecutive segments i.e. a 1-gram model will always predict 1 segment while a 3-gram model will always predict 3 segments. We compute the BERT encoding of each target emotion (i.e. the BERT encoding of emotions such as "happy", "sad", etc.) and then assign each of the n segments in a sentence an emotion using the highest cosine similarity with the emotion embeddings. Here, each segment is represented by a single embedding that is the average of the token embeddings in that segment.

Neural network models. We train two types of neural network models on top of BERT as baselines. Our first baseline is a softmax regression model that operates on each token output by the frozen BERT model and predicts an emotion class. The second baseline is a multi-layer perceptron (MLP) that again operates per-token and predicts an emotion class for each token. The architecture of the MLP can be found in appendix A.

Implementation Details. All methods are trained for 10000 iterations with an AdamW optimizer (Loshchilov and Hutter, 2017), a learning rate of $\alpha = 0.001$ and a batch size of 32. We train all models on a single NVIDIA T4 GPU with an approximate training time of about 7-8 hours. We did train two models on the same GPU at once for some of our results which likely increased training time. All models are trained using a categorical crossentropy loss wherein we predict a per-token emotion class. The pretrained BERT tokenizer and encoder used in our project was taken from Hugging Face (Wolf et al., 2019) and ignored the case of letters. The n-gram models do not require training and are thus directly evaluated on the test set. We compared validation results at 5000 and 10000 iterations for checkpoint selection for the test evaluation.

6.2 Evaluation Metrics

We evaluate emotion segmentation in two main ways: segmentation quality and emotion classification. We measure segmentation quality by using intersection over union (IoU) where we compute the amount of overlap between the groundtruth (GT) segment and the predicted segment after removing all whitespace characters and punctuation. We define a positive match between a predicted segment and the GT segment as one with IoU > 0.5, and use this to measure the precision and recall of our proposed emotion segmentation algorithm. To measure emotion classification, we compute the accuracy of the predicted emotion only on GT segments that have a match in the predicted segments. Note that each predicted segment can only be matched to a single GT and there is no double counting.

6.3 Comparisons with Baselines

We compared our proposed model to baselines and present evaluation metrics in Table 1. Since the IoU, segmentation precision and segmentation recall of our model are competitive scores, we see that our model does extremely well on the segmentation task compared to baselines. Although, the 2-gram model performs competitive on the segmentation recall metrics, we should note that by design this model segments an input sentence into two equal sized parts. Since the test set is dominated by sentences with two segments, the 2-gram is expected to be competitive on segmentation metrics such as recall. The relatively low emotion accuracy thus indicate that unlike our proposed approach it is not able to classify emotions as accurately as desired. We also outperform the softmax

regression and MLP baselines which indicate that input aggregation across time likely contributes to the segmentation task as would be expected.

Model	IoU ↑	Seg-Precision ↑	Seg-Recall ↑	Emotion Accuracy ↑
Ours	**0.9230**	**0.9440**	**0.9447**	**0.5021**
Softmax regression	0.6907	0.2510	0.6912	0.4209
MLP model	0.7238	0.2994	0.7346	0.4223
1-gram	0.4669	0.8991	0.4930	0.1197
2-gram	0.7872	0.8416	0.9230	0.2556
3-gram	0.7272	0.5304	0.8725	0.2465

Table 1: Quantitative results using the test set composed of data with multiple segments distribution.

One concern with reporting metrics on an imbalanced test set where segments of length 2 are overrepresented is that the metrics may not be representative of performance on segments of different length. This is a valid concern in model evaluation and to alleviate this concern we also independently evaluated each of the models in Table 1 on segments of different lengths. When independently evaluating models on sentences with n segments, we see that the corresponding n-gram models does well in segmentation as expected but is unable to classify with a competitive emotion accuracy i.e. a 2-gram model performs well in segmentation of sentences with 2 segments. However, these models do not generalize in performance across different segment lengths i.e. a 2-gram model does not do well on segmenting sentences with 1 or 3 segments. Although there is a little variance in results across segments of different lengths, the general trend in the main results are seen when independently evaluating segments of different lengths, thus reaffirming the validity of the results in Table 1. We refer readers to appendix B for detailed metrics on different segment lengths.

6.4 Ablation Study Comparing LSTM Architectures

We perform an ablation study to examine our design choices for our model, namely we vary both the number of stacked LSTM layers, and whether the LSTM is unidirectional or bidirectional. We present our results in Table 2. Our results indicate that our proposed bidirectional LSTM model with 4 layers performs better than a smaller model with 2 layers across most metrics and also supports our hypothesis of using bidirectionality for segmentation in LSTMs. We do see a significant increase in segmentation precision when bidirectional LSTMs are used showing the utility of context aggregation across time in making better segmentations in this challenging task.

Model	IoU ↑	Seg-Precision ↑	Seg-Recall ↑	Emotion Accuracy ↑
LSTM-Uni (2 layers)	0.9051	0.8201	0.9447	0.5084
LSTM-Bi (2 layers)	0.9153	0.9058	0.9426	**0.5133**
LSTM-Uni (4 layers)	0.8943	0.8124	0.9342	0.4958
LSTM-Bi (4 layers)	**0.9230**	**0.9440**	**0.9447**	0.5021

Table 2: Quantitative results using the test set composed of data with multiple segments distribution.

6.5 Qualitative Analysis

From our analysis of results in sections 6.3 and 6.4, it was clear that our proposed bidirectional LSTM model with 4 layers achieves the most competitive performance. We also perform a qualitative analysis on our proposed model to examine its strengths and potential limitations. Figure 4 shows examples of model outputs on the test set and compares them to ground truth.

The first three rows of Figure 4, shows instances where our model can often get perfect segmentation and emotion classification accuracies. Nevertheless, the last three rows do demonstrate failure cases with specific behaviours. For instance, the fourth and fifth rows showcases instances of under segmentation and over segmentation between the ground truth and the predicted outputs, causing

LSTM Bidirectional Networks: Analyzing Four-Layer Trends for Accurate and Inaccurate Results.				
Sentences	Predicted Segments	Ground Truth Segments	Predicted Emotions	Ground Truth Emotions
my cat was so cute when he chased his tail in circles	my cat was so cute when he chased his tail in circles	my cat was so cute when he chased his tail in circles	Happy	Happy
i am really scared of heights but i want to try skydiving someday	i am really scared of heights but i want to try skydiving someday	i am really scared of heights but i want to try skydiving someday	Fearful, Happy	Fearful, Happy
i love spending time with my family during the holidays but sometimes i just need a break from all the chaos	i love spending time with my family during the holidays but sometimes i just need a break from all the chaos	i love spending time with my family during the holidays but sometimes i just need a break from all the chaos	Happy, Neutral, Sad	Happy, Neutral, Sad
my teacher gave me a surprise pop quiz and i was completely unprepared i am so stressed out	my teacher gave me a surprise pop quiz and i was completely unprepared i am so stressed out	my teacher gave me a surprise pop quiz and i was completely unprepared i am so stressed out	Surprised, Fearful	Surprised, Fearful, Angry
my teacher gave me a really hard assignment and i am not sure i can do it	my teacher gave me a really hard assignment and i am not sure i can do it	my teacher gave me a really hard assignment and i am not sure i can do it	Neutral, Sad, Neutral, Fearful	Neutral, Angry
my cat scratched my favorite armchair and now it is ruined	my cat scratched my favorite armchair and now it is ruined	my cat scratched my favorite armchair and now it is ruined	Surprised, Sad	Sad, Angry

Figure 4: Qualitative results using the test set composed of data with multiple segments distribution, containing good and bad instances.

discrepancies that lead to emotion and segmentation precisions to decrease. Lastly, the last row illustrates that the model also has a change to perform perfect segmentation accuracy against the ground truth, but there is a chance it might get incorrect emotions.

Another important issue demonstrated by the last row is that often emotions can be ambiguous and text cannot convey them. Although the example is labeled as sad and angry in the ground truth data, it is equally plausible that it can be represented as surprised and sad, or surprised and angry. Using solely text to determine emotion has its limitation, since the manner of communicating something also conveys the speaker's emotion. Lastly, the context surrounding where the sentence in question is being used, can change the author's tone, and therefore the emotions conveyed. These are important things to keep in mind when considering the evaluation of models for emotion classification and segmentation tasks.

7 Conclusion

We explored the challenging problem of segmenting sentences into parts with distinct emotions. Our contributions are two-fold. We generated one of the first large-scale datasets with complex multi-emotion segmentation, to the best of our knowledge, in sentences for the training and evaluation of models for this task. We then proposed a method that combines pretrained sentence encoders with bidirectional LSTMs to segment sentences into segments with distinct emotions. Our method achieves competitive performance on most metrics, however we do notice a need to further improve emotion classification accuracy. Future work could include a few directions such as validating the data we generated with humans annotations for increasing trust in our dataset and correcting potentially incorrect large language model generated annotations. A second interest direction would be performing a human study of the sentences in the dataset and seeing the variance with which people attribute emotions to a given sentence. This could inform the development of better emotion classification models. Lastly, there is a need to explore more architectures that may work well for this task. Although, we did not train transformer-based methods due to time and memory constraints, there is a possibility that examining different models could lead to better performance on our task.

Acknowledgements. We would like to thank Prof. Diyi Yang for helpful discussions and feedback. We used Together AI credits for prompting large language models and Google Cloud credits for cloud computing resources used in the project.

References

Dana Alon and Jeongwoo Ko. 2021. Goemotions: A dataset for fine-grained emotion classification.

Dipankar Das and Sivaji Bandyopadhyay. 2009. Sentence level emotion tagging. In *2009 3rd International Conference on Affective Computing and Intelligent Interaction and Workshops*, pages 1–6.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Paul Ekman. 1992. An argument for basic emotions. *Cognition & emotion*, 6(3-4):169–200.

Alex Graves and Jürgen Schmidhuber. 2005. Framewise phoneme classification with bidirectional lstm networks. In *Proceedings. 2005 IEEE International Joint Conference on Neural Networks, 2005.*, volume 4, pages 2047–2052. IEEE.

Siddique Latif, Muhammad Usama, Mohammad Ibrahim Malik, and Björn W. Schuller. 2023. Can large language models aid in annotating speech emotional data? uncovering new frontiers.

Ilya Loshchilov and Frank Hutter. 2017. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*.

Walaa Medhat, Ahmed Hassan, and Hoda Korashy. 2014. Sentiment analysis algorithms and applications: A survey. *Ain Shams Engineering Journal*, 5(4):1093–1113.

Desmond C. Ong, Zhengxuan Wu, Zhi-Xuan Tan, Marianne Reddan, Isabella Kahhale, Alison Mattek, and Jamil Zaki. 2021. Modeling emotion in complex stories: The stanford emotional narratives dataset. *IEEE Transactions on Affective Computing*, 12(3):579–594.

Soujanya Poria, Navonil Majumder, Rada Mihalcea, and Eduard Hovy. 2019. Emotion recognition in conversation: Research challenges, datasets, and recent advances. *IEEE Access*, 7:100943–100953.

Tianyi Tang, Junyi Li, Wayne Xin Zhao, and Ji-Rong Wen. 2022. Context-tuning: Learning contextualized prompts for natural language generation.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. 2023. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, et al. 2019. Huggingface's transformers: State-of-the-art natural language processing. *arXiv preprint arXiv:1910.03771*.

A MLP Architecture

We summarise our MLP architecture in table 3.

Layer	Configuration
1	Linear(D, D // 2) ReLU()
2	Linear(D // 2, D // 4) ReLU()
3	Linear(D // 4, D // 4) ReLU()
4	Linear(D // 4, n_classes)

Table 3: Summary of MLP architecture. Here, $D = 768$ refers to the dimension of the encoded BERT tokens for the input sentence.

B Test results for different segment lengths

We report test results for different lengths of segments in the test set. The test set used here is the same as that used for generating the metrics reported in Table 1. However, we segregate the data by segment length over here.

B.1 Segment length of 1

Model	IoU ↑	Seg-Precision ↑	Seg-Recall ↑	Emotion Accuracy ↑
Ours	0.9894	0.9553	**1.0**	**0.5362**
Softmax regression	0.6593	0.1913	0.6511	0.4043
MLP model	0.7221	0.2496	0.7362	0.4553
1-gram	**1.0**	**1.0**	**1.0**	0.2936
2-gram	0.5925	0.5000	**1.0**	0.2723
3-gram	0.4658	0.1106	0.3319	0.0936

Table 4: Quantitative results on test set with sentences consisting of only 1 segment.

B.2 Segment length of 2

Model	IoU ↑	Seg-Precision ↑	Seg-Recall ↑	Emotion Accuracy ↑
Ours	**0.9176**	0.9506	0.9390	**0.5067**
Softmax regression	0.6919	0.2589	0.6907	0.4246
MLP model	0.7250	0.3116	0.7373	0.4290
1-gram	0.3773	**1.0**	0.5	0.1109
2-gram	0.9114	**0.9911**	0.9911	0.2661
3-gram	0.7562	0.6600	0.9900	0.2772

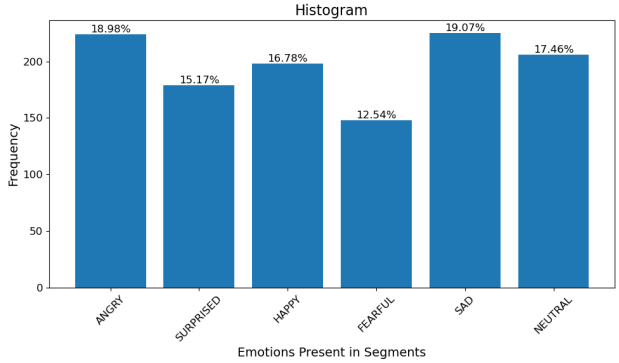
Table 5: Quantitative results on test set with sentences consisting of only 2 segments.

B.3 Segment length of 3

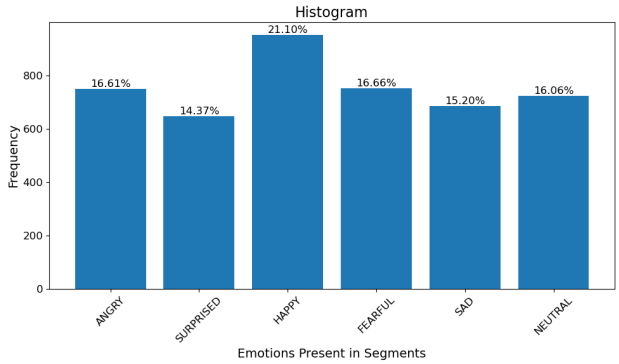
Model	IoU ↑	Seg-Precision ↑	Seg-Recall ↑	Emotion Accuracy ↑
Ours	**0.8861**	0.9144	0.9175	**0.4605**
Softmax regression	0.7125	0.2902	0.7251	0.4227
MLP model	0.7217	0.3117	0.7251	0.3746
1-gram	0.3141	0.1856	0.06186	0.0069
2-gram	0.5595	**0.9742**	0.6495	0.2096
3-gram	0.8485	0.9450	**0.9450**	0.2749

Table 6: Quantitative results on test set with sentences consisting of only 3 segments.

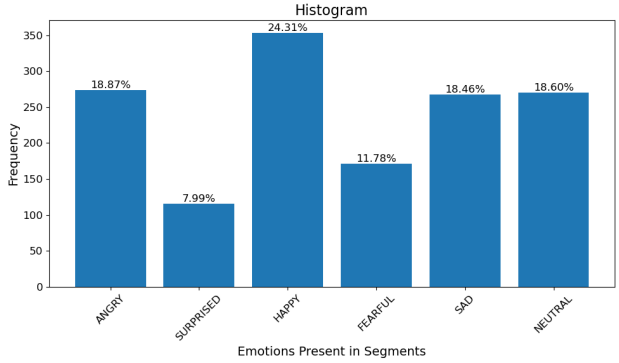
C Distribution of emotions by number of segments in sentence



(a) Distribution of emotions in sentences with 1 segment



(b) Distribution of emotions in sentences with 2 segments



(c) Distribution of emotions in sentences with 3 segments. Since the number of sentences with 3 segments is relatively scarce, we choose to retain a larger portion of the data at the cost of a more imbalanced distribution.

Figure 5: Emotion distribution by number of segments

D Prompt to LLAMA-2

We want to generate a dataset of examples for emotion segmentation in text. The only possible tags for emotions are <Angry>, <Surprised>, <Disgusted>, <Happy>, <Fearful>, <Sad>, and if there is no emotions in a segment use <Neutral>. For each emotion additionally tag it with a suffix -Start and -End. Each -Start must have a corresponding -End or the example is invalid. Here are in-context examples to learn from:

Some examples:

Sentence: Today was a bad day but at least I saw a llama.
Emotions: <Sad-Start>Today was a bad day<Sad-End><Happy-Start>but at least I saw a llama.<Happy-End>

Sentence: My teacher called on my friend in class to present and my friend *freaking* volunteered me instead.
Emotions: <Neutral-Start>My teacher called on my friend in class to presents<Neutral-End><Angry-Start>and my friend *freaking* volunteered me instead.<Angry-End>

Sentence: The pizza was delicious, but it gave me a stomachache.
Emotions: <Happy-Start>The pizza was delicious<Happy-End><Sad-Start>, but it gave me a stomachache.<Sad-End>

Sentence: I'm so done with this project it's taking forever, but I'm close to being done and I have to admit that I'm kind of excited to see the final result.
Emotions: <Angry-Start>I'm so done with this project it's taking forever<Angry-End><Neutral-Start>, but I'm close to being done<Neutral-End><Happy-Start> and I have to admit that I'm kind of excited to see the final result.<Happy-End>

Sentence: I can't believe I found a cockroach in the shower. It totally grossed me out and now I have to move it!
Emotions: <Surprised-Start> I can't believe I found a cockroach in the shower.<Surprised-End><Disgusted-Start> It totally grossed me out<Disgusted-End><Fearful-Start> and now I have to move it!<Fearful-End>

ONLY output exactly ten additional unique examples using this format and using EXACTLY two or three different emotion tags of <Angry>, <Surprised>, <Disgusted>, <Happy>, <Fearful>, <Sad> per sentence. DO NOT explicitly state an emotion in the sentence generated. Don't say "I feel <emotion>" or "it made me <emotion>"). The annotated emotions must correctly reflect the order and use of words in the sentence given.