Towards a Sentiment-Aware Conversational Agent

Isabel Dias
isabel.h.dias@tecnico.ulisboa.pt
INESC-ID, Instituto Superior Técnico, Universidade de
Lisboa
Portugal

Ricardo Rei
ricardo.rei@unbabel.com
Unbabel, INESC-ID, Instituto Superior Técnico,
Universidade de Lisboa
Portugal

Patrícia Pereira
patriciaspereira@tecnico.ulisboa.pt
INESC-ID, Instituto Superior Técnico, Universidade de
Lisboa
Portugal

Luisa Coheur
luisa.coheur@tecnico.ulisboa.pt
INESC-ID, Instituto Superior Técnico, Universidade de
Lisboa
Portugal

ABSTRACT

In this paper, we propose an end-to-end sentiment-aware conversational agent based on two models: a reply sentiment prediction model, which leverages the context of the dialogue to predict an appropriate sentiment for the agent to express in its reply; and a text generation model, which is conditioned on the predicted sentiment and the context of the dialogue, to produce a reply that is both context and sentiment appropriate. Additionally, we propose to use a sentiment classification model to evaluate the sentiment expressed by the agent during the development of the model. This allows us to evaluate the agent in an automatic way. Both automatic and human evaluation results show that explicitly guiding the text generation model with a pre-defined set of sentences leads to clear improvements, both regarding the expressed sentiment and the quality of the generated text.

KEYWORDS

dialogue systems, sentiment prediction, answer generation

1 INTRODUCTION

End-to-end data-driven conversational agents have become popular over the last years. Current works focus on ways to explicitly condition text generation models in order to produce certain attributes, such as emotions [12, 39].

In fact, generating answers with the appropriate emotion can be particularly important in scenarios as, for instance, customer support. In the example of Figure 1, while both answers can be considered correct, a reply that expresses an appropriate emotion may be more satisfying for the user, rather than the generic reply.
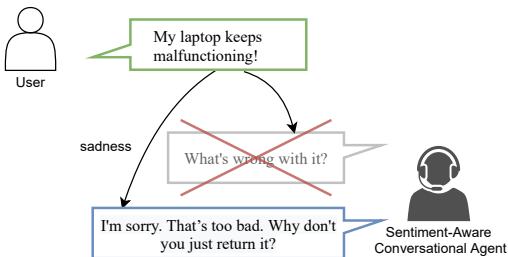

Figure 1: Dialogue generated by Baseline vs. SACA conditioned on *Sadness* sentiment.

As highlighted in [20], in human-human conversations the emotions expressed in two subsequent utterances from different speakers often change and are important when predicting the "correct emotion for an upcoming response before generation". According to [36], the drawback of current approaches is that they can not be used in a dialogue setting, given that they do not introduce a mechanism to automatically predict the next appropriate attribute.

In this paper, we propose an **end-to-end sentiment-aware conversational agent** that predicts the next appropriate attribute and generates its answers accordingly. Figure 2 depicts the proposed architecture, which is based on two main models:

- A **reply sentiment prediction model**, which predicts the appropriate reply sentiment that should be expressed by the conversational agent in the next utterance;
- A **text generation model**, which generates a sentence that is context-aware and expresses the predicted sentiment.
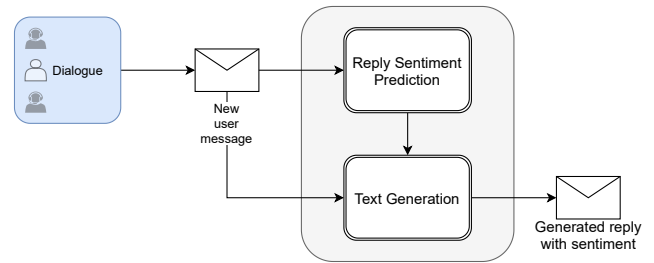

Figure 2: Proposed sentiment-aware conversational agent.

We also propose to use a **sentiment classifier** to evaluate the generated sentences. This allows us not only to understand whether the model is expressing the correct sentiment, but also to do so in an automatic manner during the development of the model. Furthermore, we explore **retrieval augmentation** in order to bias the classification models towards the correct sentiment label.

In addition, we study the best way to add information regarding the appropriate sentiment, in order to guide the text generation model. To this end, several **sentiment's lexicons**, such as simple sentiment tags or sentences capturing the different sentiments are considered; as we will see, feeding the model with a small set of sentences expressing emotions leads to the best results. In fact, our results show that: a) using the appropriate sentiment, improves,

Sentence	Label	Representation	Label
Does it cost anything?	Neu	`[CLS]Does it cost anything?[SEP]`	Neu
Yeah 20$ per month.	Neu	`[CLS]Yeah 20$ per month.[SEP]Does it cost anything?[SEP]`	Sur
Ohh!	Sur	-	-

Table 1: Example of a dialogue from the EmotionPush dataset along with the sentence representation and corresponding label for the reply sentiment prediction task. Neu corresponds to the label *Neutral*, and Sur to the label *Surprise*.

indeed, the scores in the generation process; b) that the evaluation based on the sentiment classification model correlates well with the human evaluation performed; c) although following current state of the art architectures, there is still plenty of room for improvement in the reply sentiment prediction model; and that d) the retrieval augmentation process did not significantly improve results.

We test our models with the EmotionPush [11] and DailyDialog [21] corpora, which use as labels the six Ekman's basic emotions [5], and neutral.

2 THE SENTIMENT AWARE AGENT

In this section, we describe the proposed sentiment classification model, reply sentiment prediction model, and text generation model, conditioned on the predicted sentiment and on the past conversation context. We also describe the retrieval augmentation techniques used.

2.1 The Sentiment Classification and Reply Sentiment Prediction Models

For both sentiment classification and reply sentiment prediction we used pre-trained Transformer models, such as BERT [3]. We improved upon this base model, by using previous dialogue context. In order to add context to the input of the model, we took advantage of a particularity of the BERT architecture, the [SEP] token, by considering that the first sentence after the [CLS] token is the sentence we are trying to classify, and it is followed by its context. Thus, given a dialogue $D = (s_1, s_2, ..., s_n)$, with n equal to the number of sentences in the dialogue, in order to classify the sentence s_i with x sentences as context, the input to the model is $concat(s_i, s_{i-1}, ..., s_{i-x})$.

Regarding the reply sentiment prediction model, similarly to the contextual sentiment classification, we also made use of the [SEP] token to separate the different sentences that are part of the input. However, there are two main differences: first, the input of the reply sentiment prediction model corresponds only to the previous context sentences; secondly, the gold label is the sentiment of the upcoming sentence. In this task we can also use an arbitrary number of sentences as context. We will use as default setup the last two previous subsequent sentences. In Table 1 we can observe an example of the input of the model.

2.2 The Text Generation Model

The text generation model receives the sentiment predicted by the reply sentiment prediction model, as well as the previous context of the conversation, to generate a sentence that is not only appropriate given a context, but also expresses the predicted sentiment. To do so, we adapted part of the work in [35], that proposes a model that is able to leverage a set of sentences that describe a given persona in order to generate text that is coherent with that

persona. We adapted this concept and developed a **sentiment lexicon knowledge base**, which was used to make the conversational agent sentiment-aware. In order to guide the model towards an appropriate sentiment, we concatenated the desired sentiment's lexicon to the input of the model. The sentiment lexicon for each sentiment was built using the following techniques:

- **Tag**: use the sentiment's name (e.g., anger, joy, etc.). We will use this model as a baseline for the sentiment conditioned text generation model;
- **TF**: retrieve the 40 most frequent n-grams from the set of training sentences of each sentiment. We assume $1 \leq n \leq 3$;
- **TFU**: the same as the **TF** approach, but removing n-grams that are not unique to each sentiment. We assume $1 \leq n \leq 3$;
- **TF-IDF**: retrieve 40 n-grams with the highest TF-IDF score. We assume $1 \leq n \leq 3$;
- **Random Sample**: select at random a sentence from the training set labelled with the sentiment we want to generate;
- **Sentiment Sentences**: use a pre-defined set of sentences representative of each sentiment. The set we are going to use can be observed in Table 2. This set of short sentences was created using always the same sentence structure ("I am..." and "That is..."). For the DailyDialog corpus we do not include the *non-neutral* label.

Sentiment	Sentence 1	Sentence 2
Anger	I am angry.	That is so annoying!
Disgust	I am disgusted.	That is repulsive!
Fear	I am frightened.	That is scary!
Joy	I am happy.	That is delightful!
Neutral	I am ok.	That is ok.
Non-neutral	I am not ok.	That is not ok.
Sadness	I am sad.	That is so upsetting.
Surprise	I am surprised.	That is so amazing!

Table 2: Sentences used to represent each sentiment.

2.3 The Retrieval Augmentation Model

Due to the recent success of retrieval augmentation approaches in NLP tasks [4, 19, 25], including sentiment classification [26], we also explored a mechanism that relies on nearest neighbors. Following [26], the first step was to find the nearest training example for each train/development/test example, and the corresponding label. To do so, we used the Sentence Transformer [29][1] library to create sentence embedding representations of all examples using the `paraphrase-distilroberta-base-v1` model. Next, we used

---

[1]https://www.sbert.net

the FAISS [14][2] library to build an index with the sentence embeddings that belong to the train set in order to find the closest training examples. In particular, we chose the Euclidean distance to calculate the distances between the examples. After retrieving the nearest neighbors information, the second step is to incorporate it in the Transformer model. We started by initializing an extra set of embeddings, which we will call Sentiment Embeddings (SE), one for each sentiment label, that are trained along with the model. We initialized the embedding of each sentiment with the average of the sentences' corresponding to a given sentiment. E.g., the sentiment embedding for *joy* was initialized with the average of the embeddings corresponding to the sentences labelled with the sentiment *joy* in the training set. Then, for each example, we incorporated the nearest training example label in the Transformer model, by concatenating the corresponding SE to its output, after pooling, which will be the input to the classification layer.

3 EXPERIMENTAL SETUP

This section describes the datasets, evaluation metrics, and setups used during this work[3].

3.1 Datasets

We tested our models in the EmotionPush [11] and DailyDialog [21] corpora. The former is composed of 1000 private conversations from Facebook Messenger. The latter, with dialogues in daily life scenarios, contains 13118 multi-turn dialogues, divided in 10 themes such as: Finance, Politics, Health, Work, etc. Both corpora use as labels the six Ekman's basic emotions [5], and neutral. The EmotionPush dataset has an additional label, non-neutral, to represent sentences in which the annotators did not reach consensus.

3.2 Evaluation Metrics

We evaluated the sentiment classification and the reply sentiment prediction models using the micro (m) and macro (M) F1 averages. Given that both corpora are highly unbalanced, with the *neutral* label composing over 80% of the examples, we also evaluated our models both with the majority class and without the majority class (micro/Macro-No Majority Class metric which we refer to as m/M-NMC-F1). This allowed us to have a better understanding of how the models are performing on less represented sentiments.

Regarding the text generation model, our choice of automatic evaluation metrics aims to measure if the model is able to generate a sentence that expresses a desired sentiment without compromising the quality of the text. Therefore, first, regarding the quality of the generated text, we focused on Perplexity (*PPL*) [15], and Sentence Embedding Similarity (*SES*) [36]. Second, to evaluate if the generated text is expressing the appropriate sentiment, we used the sentiment classification model to classify the generated sentences and evaluate them using the aforementioned F1 metrics.

3.3 Training Setup

The models were implemented using PyTorch Lightning [6] and the HuggingFace Transformers [34] library.

The sentiment classification and reply sentiment prediction models were trained for a maximum of 40 epochs, using the cross entropy loss, with 4 validation steps per epoch, stopping after 10 consecutive validation steps without improvement. The checkpoint used to evaluate the model was the one that achieved the highest validation macro-F1 value. We follow the work by [10] and use the Adam optimizer [18] with a discriminative learning rate of 1×10^{-3}, except for the Transformer model that has a learning rate of 5×10^{-6}. For the Transformer model we apply a layer-wise learning rate decay of 0.95 after each step. We apply a dropout [30] of 0.4 to the sentence embeddings during training. We use a real batch size of 16 whenever the GPU's memory allowed it, but used gradient accumulation to always simulate a batch size of 32.

The text generation models were trained for a maximum of 40 epochs, using the negative log-likelihood loss, with 4 validation steps per epoch, stopping the training after 12 consecutive validation steps without improvement. We use the Adam optimizer [18] with a learning rate of 5×10^{-6}. The checkpoint used to evaluate the model was the one that achieved the lowest validation negative log-likelihood loss value. Due to computational constraints, we always use the two most recent context sentences from the dialogue as input to the model. We use a real batch size of 4 whenever the GPU's memory allowed it, but we used gradient accumulation to always simulate a batch size of 16. All other hyperparameters were kept as default.

3.4 Models Setup

Considering sentiment classification, and by using the development set, we empirically found the following optimal setup: a RoBERTa-large model [22], that receives as input the concatenation of the sentence to be classified, with the last previous context sentence; a linear classification layer that receives as input the concatenation of the [CLS] token embedding of the last 4 hidden layers (*concat4* pooling); and uses the retrieval augmentation method previously described. As a baseline we considered a RoBERTa-large model with a linear classification layer, that receives as input the sentence to be classified.

Regarding the reply sentiment prediction task, and by also using the development set, the optimal setup found was: a RoBERTa-large model, that receives as input a concatenation of the last four context sentences; and a linear classification layer. The Retrieval augmentation did not improve performance and therefore was not used for this task. As a baseline we considered a RoBERTa-large model with a linear classification layer, that receives as input the last two context sentences.

Finally, our conditioned text generation model consists on a DialoGPT-small model, with the concatenation of the desired sentiment's lexicon to the input of the model.

4 RESULTS

4.1 Sentiment Classification

We start by presenting the results of our Sentiment Classification model (from now on "SentClass" Model) (Table 3).

---

[2]https://faiss.ai
[3]The code is publicly available here: The repository will be made available.

[4]with Contextual Augmentation

EmotionPush				
Model	**m-F1**	**m-NMC-F1**	**M-F1**	**M-NMC-F1**
Baseline	**78.9**	57.6	45.4	39.2
SentClass	**78.9**	**58.2**	**54.1**	**49.2**
+ RoBERTa-base	-0.7	-0.8	-4.2	-4.7
+ CLS	-0.9	-0.9	-1.6	-1.7
- Context 1	-2	-2.6	-8.8	-9.9
- Ret. Aug.	-0.7	+0.3	+3.1	+3.6
DailyDialog				
Model	**m-F1**	**m-NMC-F1**	**M-F1**	**M-NMC-F1**
Baseline	84.5	56.5	48.1	41.0
SentClass	**85.0**	**58.1**	**51.0**	**44.4**
- Context 1	-0.6	-1.6	-2.5	-3.0
- Ret. Aug.	-0.6	-1.3	-1.0	-1.2
KET [38]	-	53.4	-	-
ELECTRA[4] [17]	-	57.9	-	-
COSMIC [7]	-	58.5	51.1	-

Table 3: Baseline vs. SentClass Model + Ablation Study.

Regarding the experiments done on the EmotionPush dataset, when compared with the baseline, our model improves all metrics, except the m-F1, where it maintains the same value. More notably, it is able to improve the M-F1 metric by 8.7 points. These improvements are also noticeable on the M-NMC-F1 metric, where our model improves 10 points.

In order to further validate our results we performed an ablation study using the test set, also displayed in Table 3, with four experiments defined as follows: **+RoBERTa-base**, where we replace RoBERTa-large by a RoBERTa-base; **+CLS**, where we replace the *concat4* pooling by the embedding of the [CLS] token of the last hidden layer; **-Context 1**, where we no longer use context in the input of our model; **-Ret. Aug.**, where we remove the retrieval augmentation methods from the model. Results showed that removing the context is what impacts the model the most, which tells us it was the most significant addition to our model. Additionally, replacing the RoBERTa-large by the RoBERTa-base and the *concat4* by the *CLS* pooling option, also worsens all metrics. Interestingly, in the test set, using retrieval augmentation worsened our results.

Regarding the results on the DailyDialog corpus, since on the ablation study performed on the EmotionPush corpus we found that removing context and retrieval augmentation had the most impact on the model, we focused only on those changes for this corpus. We can observe that all metrics improve significantly when compared to the baseline. In particular, the m-F1 metric improves by 0.5 points, and without the majority class by 1.6 points. Regarding the M-F1 metric, it improves by 2.9 points and without the majority class by 3.4 points. Both removing the context and retrieval augmentation worsens the results. We should also notice that our model outperforms all approaches except when compared with the work from [7] that makes use of an additional large knowledge base that captures certain aspects such as personality, or emotion interactions, although the performance is quite similar[5].

---

[5]For EmotionPush we can only compare our results with the ones from [16], although the metric used is the Unweighted Accuracy. We also outperform this work: 62.2 vs. 70.7 (ours). All the other works that use this dataset provide different splits.

4.2 Reply Sentiment Prediction

In Table 4 we can observe the comparison of our reply sentiment prediction (RSP) model. As described in Section 3.4, we use a RoBERTa-large which receives as input a concatenation of the last four context sentences. As a baseline, we considered the same model, but receiving as input the last two context sentences.

EmotionPush				
Model	**m-F1**	**m-NMC-F1**	**M-F1**	**M-NMC-F1**
Baseline	**69.0**	18.0	15.0	5.5
RSP Model	66.5	**19.6**	**17.8**	**8.8**
+ RoBERTa base	+3.4	-6.8	+2.5	-3.1
+ CLS	+2.6	-4.9	-1.6	-2.1
+ Context 2	+0.2	-3.4	-1.3	-1.5
DailyDialog				
Model	**m-F1**	**m-NMC-F1**	**M-F1**	**M-NMC-F1**
Baseline	**80.7**	40.1	33.8	24.6
RSP Model	80.4	**42.8**	**35.0**	**26.1**
+ RoBERTa base	-1.2	-1.5	-2.9	-3.2
+ CLS	+0.7	-1.3	-0.4	-0.6
+ Context 2	-0.2	-1.7	+0.5	+0.5

Table 4: Baseline vs. RSP Model + Ablation Study.

We can start by observing that only the m-F1 does not improve when compared to the baseline. Nonetheless, our model outperforms the baseline in all other metrics (m-NMC-F1 by 6.7 points, M-F1 by 2.4 points, and M-NMC-F1 by 2.9 points). This shows that our model is better at generalizing for less represented sentiments.

In order to further validate our results, we performed another ablation study, which can also be observed in Table 4. In this study, all models have a higher m-F1 than the RSP model. On the remaining metrics all perform worse than the RSP model. These results show how our model is doing a trade-off between a lower F1 in the majority class and a higher F1 on the less represented sentiments.

Regarding the DailyDialog corpus, the results obtained show that our introduced changes also improve the baseline. Furthermore, we can observe that even using a larger number of training examples, the reply sentiment prediction task is still hard to perform well in. Finally, regarding the ablation study, we can observe that all introduced changes result in an improvement on most metrics.

4.3 Selecting the Best Sentiment Lexicon

In order to understand which sentiment lexicon resulted in the best performance, we applied each lexicon to a Dialo-GPT-small text generation model, concatenating the appropriate sentiment lexicon to the input of the model, as described in Section 2.2. This experiment was done using the EmotionPush dataset. The results can be observed in Table 5.

Regarding the perplexity, we can observe that all approaches, except the *TF-IDF* and the *Sentiment Sentences*, perform similarly to the baseline (*None*). When compared to the baseline, the *TF-IDF* approach worsens the perplexity by 28.5 points, while the *Sentiment Sentences* approach improves this metric by 6.1. Furthermore, the *Sentiment Sentences* approach also improves the SES metric by 1.2 points. More interestingly, this approach improves significantly the metrics that are evaluating the expressed sentiment. When

Sentiment Lexicon	PPL	SES	F1			
			m	m-NMC	M	M-NMC
None	92.4	16.8	42.5	15.6	13.2	6.4
Tag	90.1	17.7	45.6	16.8	14.2	7.2
TF	91.2	16.4	45.5	14.4	12.6	5.6
TFU	90.7	16.3	45.0	14.7	13.8	6.7
TF-IDF	120.9	16.4	41.7	13.6	12.6	5.7
Random Sample	94.3	16.8	44.7	20.1	15.0	8.3
Sentiment Sentences	**86.3**	**18.0**	**62.7**	**42.3**	**29.1**	**22.4**

Table 5: Results of concatenating sentiment lexicon to the input of the DialoGPT-small text generation model using the EmotionPush dataset.

compared to the baseline, the m metric improves by 20.2 points, the m-NMC by 26.7 points, the M by 15.9 points, and the M-NMC by 16 points. These results show how this model is not only generating better sentences, due to the improvements on the perplexity and the SES, but also expressing the correct sentiment more times, which can be concluded by the improvements on the metrics that evaluate the expressed sentiment.

4.4 Text Generation

In Table 6, we can observe the results obtained in the text generation experiments. For these, we consider two baselines: the first (*BL*) is a DialoGPT-small not conditioned on sentiment; and the second (*Tag*) is a DialoGPT-small conditioned on the sentiment tags. Additionally, we show the results obtained using our best performing model, (*SM*) a DialoGPT-small conditioned with the pre-defined set of sentiment sentences defined in Table 2.

Both *Tag* and *SM* models are able to improve the perplexity. Furthermore, these models perform exceptionally well when compared to the *BL* model on the sentiment-related metrics, increasing all metrics significantly. Nonetheless, the *SM* approach yields the best overall results.

Emotion-Push						
	PPL	SES	m-F1	m-NMC-F1	M-F1	M-NMC-F1
BL	85.0	17.3	47.6	22.6	17.1	10.3
Tag	**78.9**	16.7	61.4	43.9	24.2	17.4
SM	79.4	**18.0**	**64.3**	**45.5**	**33.0**	**26.6**
DailyDialog						
	PPL	SES	m-F1	m-NMC-F1	M-F1	M-NMC-F1
BL	9.9	26.7	77.9	30.6	24.9	14.5
Tag	**9.5**	**28.5**	83.6	51.2	42.6	34.6
SM	9.6	27.3	**84.6**	**53.1**	**48.5**	**41.4**

Table 6: Results with the three different models.

Contrarily to the results observed for the EmotionPush corpus, the results obtained on the DailyDialog corpus, show that the *Tag* and the *SM* models perform more similarly. However, it is relevant to mention that the *SM* model also outperforms the *Tag* model on the sentiment metrics, more significantly on the macro metrics.

As previously mentioned, the EmotionPush corpus is retrieved in an online chat context, which means the text is very informal,

while the DailyDialog corpus was built from websites that are used to practice English, which makes the corpus more formal and fluent. This aspect could influence the quality of the generation models. In particular, the fact that the *Tag* and *SM* models fine-tuned with the DailyDialog corpus perform similarly could be an indication that the quality of the data used makes the models fine-tuned for this corpus not as dependent on the provided set of sentences, and a more simple option, such as a sentiment tag, is enough to guide the models. In contrast, the informality of the EmotionPush corpus could be making the models fine-tuned for this dataset more reliant on full sentiment sentences in order to generate good quality text conditioned on a sentiment.

4.5 Sentiment-Aware Conversational Agent

In order to fully evaluate the sentiment-aware conversational agent, we considered three models:

- the *Baseline*, the DialoGPT-small model, which is not conditioned on sentiment;
- the *Oracle*, the DialoGPT-small + pre-defined set of sentences model. Since this model is conditioned on the gold sentiment label, it represents the proposed sentiment-aware conversational agent if the RSP model was perfect;
- *SACA*, the proposed Sentiment-Aware Conversation Agent. This agent consists on the best performing models obtained empirically. The RSP model (RoBERTa-large + four context sentences), and the text generation model conditioned on the RSP model's predictions (DialoGPT-small + pre-defined set of sentences).

Emotion-Push						
	PPL	SES	m-F1	m-NMC-F1	M-F1	M-NMC-F1
Baseline	85.0	17.3	47.6	22.6	17.1	10.3
Oracle	**79.4**	**18.0**	**64.3**	**45.5**	**33.0**	**26.6**
SACA	80.3	16.6	49.8	21.6	15.7	8.4
DailyDialog						
	PPL	SES	m-F1	m-NMC-F1	M-F1	M-NM-F1C
Baseline	9.9	26.7	77.9	30.6	24.9	14.5
Oracle	**9.6**	27.3	**84.6**	**53.1**	**48.5**	**41.4**
SACA	9.7	**27.6**	77.9	30.8	26.0	15.8

Table 7: Baseline + Oracle + Sentiment Aware Conversational Agent (SACA) results.

The results obtained can be observed in Table 7. The first conclusion we can take is that the appropriate sentiment, improves, indeed, scores in the generation process, as the *Oracle* outperforms the *Baseline*. Nevertheless, for both datasets, the introduction of the reply sentiment prediction is still far from the *Oracle*. This proves the need for a better reply sentiment prediction model, which is key for the proper functioning of the proposed conversational agent. It should be noticed, however, that using the model with the pre-defined set of sentiment sentences in the *SACA* still improves the *perplexity* when compared to the *Baseline*, which shows that it is the set of sentences that is being concatenated to the input that is improving the *perplexity*, and not whether the model is receiving

the correct sentiment or not. Regarding the *sentence embedding similarity*, on the EmotionPush corpus it is noticeable that the *SES* is better when the sentiment metrics and the *perplexity* achieve better results. On the DailyDialog corpus this is not as perceptive, given the low fluctuation of the metric between the evaluated models.

An example of the sentences generated by the developed sentiment-conditioned text generation model can be observed in Table 8. For the same context sentence we present the sentences generated by the model conditioned on each of the target sentiments. Additionally, we also show the sentence generated by the baseline, which is not conditioned on any sentiment. The model is capable of generating fluent and adequate answers for all sentiments. Furthermore, most sentences generated express the desired sentiment. Another interesting aspect is that the model only repeated keywords from the pre-defined set of sentiment sentences twice on a sentence generated conditioned on the *fear* emotion. Interestingly, the *fear* sentiment is the most underrepresented sentiment in the corpus, which might have made the model rely more on the provided set.

Example 1	
Context	How do you feel?
Baseline	I feel great, thanks.
Anger	I'm sick of this city.
Disgust	I'm very nauseous. I've never felt so nauseous.
Fear	I'm scared. I'm scared of the future.
Joy	I like it very much.
Sadness	I'm really sorry, I'm really not feeling well.
Surprise	Oh, my god! My eyes are burning!

Table 8: Example of sentences generated by the sentiment-conditioned text generation model. Words in red are present in the pre-defined set of sentences.

4.6 Human Evaluation

As mentioned in [36], automatically evaluating empathetic conversational agents is a challenging task given the limitations of the automatic metrics. In particular, the most common text generation metrics evaluate the word/lexical overlap between the gold and generated sentences, and in a dialogue setting there can be many correct answers. Furthermore, the experiment previously reported on Table 8, further motivates the challenge of relying on these metrics to evaluate sentiment-aware conversational agents: feeding to the model the exact same context and conditioning it on different sentiments drastically changes the outcome of the generated sentences. For that matter, regarding the evaluation of our model, the most important aspects to evaluate were the adequacy given the previous context, and whether it expressed the desired sentiment. For this evaluation we were able to gather answers from seven annotators with a proficient English level.

This evaluation was done by sampling at random 40 inputs from the test set of each corpus and retrieving the corresponding replies generated by four of the developed architectures: *Baseline*, the DialoGPT-small model; *Baseline Oracle*, the DialoGPT-small + *tag* model; *Oracle*, the DialoGPT-small + pre-defined set of sentences model; and *SACA*, the sentiment-aware conversational agent.

In order to evaluate the adequacy of the reply we asked the annotators the following question: "Do the replies sound appropriate considering the context of the dialogue?". A similar process was followed to evaluate the sentiment of the sentences. Our goal with this evaluation was to assess if the model was able to generate sentences with the desired sentiment. To do so, we asked the annotators if the reply expressed the gold sentiment. The question asked was "Does the reply represent the <sentiment_name> emotion?". Additionally, given the multitude of sentiments present that often can be interchanged, we asked the annotators to consider whether the sentence being evaluated could be said to express the asked sentiment. For example, "What?" could be used to express *anger* or *surprise*, depending on the tone used. We use a 2-point Likert scale for both questions (*Yes* or *No*).

Model	EmotionPush		DailyDialog	
	Adequacy	**Sentiment**	**Adequacy**	**Sentiment**
Baseline	0.4292	0.325	0.4958	0.3708
Baseline Oracle	0.5042	0.5167	**0.6542**	0.7084
Oracle	**0.6167**	**0.6583**	0.6292	**0.7625**
SACA	0.3917	0.3542	0.5667	0.3292

Table 9: Average EmotionPush and DailyDialog Scores.

The results obtained can be observed in Table 9. Given that we are using a Likert scale of 2-points, the reported scores correspond to the ratio of positive answers. E.g., an adequacy score of 0.6 for the *Baseline* model means that 60% of the generated sentences by this model were considered adequate. Similarly, sentiment score also corresponds to the ratio of positive answers. It is clear that the model that achieves the best performance was the *Oracle*. In particular, this model improved the adequacy of the replies when compared to all other models. It also achieves the highest sentiment scores. It is also interesting to observe that, despite the accumulated error of the *SACA* due to the reply sentiment prediction model, this model still outperforms the *Baseline* on the sentiment metric. There also seems to exist a correlation between the sentiment metric and how adequate the replies are. The models that achieved a higher sentiment metric, also tend to be more adequate. Regarding the DailyDialog corpus, both *Baseline Oracle* and *Oracle* perform similarly. The improvements of both models when compared to the *Baseline* are considerable across all metrics.

	PPL	**SES**	**m-NMC**	**M-NMC**
EmotionPush				
Adequacy	0.18	0.9524	0.8015	0.8522
Sentiments	0.01	0.9519	0.7992	0.8328
DailyDialog				
Adequacy	-0.981	0.7992	0.8852	0.8589
Sentiments	-0.7345	0.4475	0.9966	0.9839

Table 10: Pearson correlation between the automatic metrics and the human evaluation metrics.

Although a correlation using four points (corresponding to pairs of the automatic and human metrics obtained for each evaluated model) is not ideal, and might not lead to statistically significant values, we finalize this analysis with the correlation between the

obtained automatic metrics (perplexity, sentence embedding similarity, micro/macro-F1 without the majority class), and the human evaluation (adequacy and sentiment) by using the Pearson correlation to measure the linear relationship between them. We can observe the results in Table 10. The correlation between the sentiment automatic metrics and the human evaluation metrics is generally very high for both corpora. It is relevant to mention that on the EmotionPush corpus, the correlation between the human evaluation metrics and the perplexity is very low, while on the DailyDialog corpus is close to perfect. This showcases the difference in the quality of the text in both datasets. Furthermore, we highlighted in Section 4.4 that models fine-tuned with formal and fluent data, such as the DailyDialog corpus, seem to perform well with simpler sentiment-conditioning approaches. In contrast, models fine-tuned with informal data, such as the EmotionPush corpus, seem to rely more on full sentiment sentences in order to perform well. This hypothesis is further validated with these experiments, since the annotators gave higher scores to the *Oracle* model fine-tuned for the EmotionPush corpus, while the annotations gathered for the DailyDialog corpus showed similar results for the *Oracle* and *Baseline Oracle* approaches.

The high correlation obtained in the human and automatic metrics highlights the importance of having a sentiment classification model as a metric to guide the development of the models.

5 RELATED WORK

5.1 Sentiment Classification

The current state-of-the-art for sentiment classification is to use dense contextual word embedding models, such as BERT [3], via transfer learning. Other works on sentiment classification made use of context, speakers, speech acts and topics information [17], or further pre-trained BERT with a dataset similar to the target sentiment-labelled dataset [13].

5.2 Reply Sentiment Prediction

One critical aspect of systems that deal with sentiment-aware text generation in a dialogue context is how to define the appropriate sentiment for the upcoming reply. Several works approach this task (*e.g.* [9] or [2]) by adding additional dialogue and textual features, such as the time between interactions, or the emotion of previous sentences, to aid the model. To simulate the emotion transition in humans, the work described in [33] makes use of the Valence-Arousal-Dominance emotion space, which encodes the emotion of words in a 3-dimensional vector space, to calculate the "emotion transition as the variation between the preceding emotion and the response emotion". Predicting the next sentiment can also be part of the text generation model, as described in Section 5.3.

5.3 Sentiment-Aware Text Generation

The current state-of-the-art in dialogue text generation consists in data-driven end-to-end models, which are capable of generating fluent, appropriate, and meaningful responses in a dialogue setting, by using previous context sentences as input to text generation models. One of the first successful approaches was proposed in [31], which leverages the Seq2Seq architecture to predict an upcoming

sentence. The work proposed in [35] applies this idea to the Transformer architecture, and fine-tunes the GPT-2 model using two additional special tokens that are used to separate the sentences belonging to different speakers in the model's input.

Regarding text generation with sentiment, the authors of [28] develop the EmpatheticDialogues dataset, fine-tune a GPT-2 model with it, and conclude that using a large-scale empathetic corpus enables the models to express appropriate emotions implicitly. Alternative recent approaches make use of mechanisms that condition the text generation model on a desired sentiment. Some use a pre-defined set of rules or heuristics [1, 12]; others, data-driven approaches [23, 27, 36]. In [1] it is proposed a conversational model which embeds words using the Valence-Arousal-Dominance emotion space [24], and explores decoding techniques that encourage diversity in candidate outputs. Furthermore, it designed a "training loss to explicitly train an affect-aware conversation model", following three heuristics: minimizing affective dissonance (generated emotion should be similar to the input's emotion); maximizing affective dissonance (generated emotion should not be aligned with the input's emotion); and maximizing affective content (generated emotion should avoid being neutral). In [12] it is adopted an "emotion mining from text" classifier, developed in [37], to classify the emotions expressed in previous conversation context, and used this information, together with pre-defined mapping rules defined by the authors, to decide which emotion should be expressed. However, in [36] it is argued that approaches that rely on a pre-defined set of rules or heuristics, such as the previously mentioned, are not supported by psychology literature, and therefore emotional interactions in human-human conversations should be explored by using data-driven language models with large-scale emotional corpora. Some works leverage a multi-task approach that jointly trains an emotion encoder and a text generation model, which is conditioned on the emotional state assessed by the emotion encoder [23, 27]. Similarly to our work, [36] incorporates an intent predictor, which is separately trained from the text generation model, with the goal of deciding the intent for the reply to be generated. That intent is predicted based on previous context, and is then encoded and fed to the text generation model. Our work mainly differs by having a different model architecture, and exploring smaller datasets. Interestingly, their analysis on the intent predictor reaches similar conclusions to ours, which further shows that the answer to achieve a better performance in the reply sentiment prediction task might not be directly related to the amount of data used to train the models, and other methods should be explored.

6 CONCLUSION AND FUTURE WORK

In this work we built a sentiment-aware conversational agent. In particular, we observed that using multiple context sentences on the input of the reply sentiment prediction model, and using a pre-defined set of sentiment sentences to condition the text generation model, improved performance when compared to baseline models. Furthermore, for text generation, we showed how our approach resulted in clear gains. Additionally, we observed how the reply sentiment prediction model is the bottleneck of the agent. Finally, we also performed a human evaluation on the developed models which corroborated the results obtained on the automatic

evaluation, highlighting the necessity of metrics that evaluate the sentiment expressed during development.

As future work, we consider that improving the reply sentiment prediction model is crucial for a better performance of the conversational agent. To do so, we propose to explore different ways to incorporate the retrieval augmentation methods [32], prompt-based learning [8], or hybrid approaches with both data-driven and pre-defined rules.

ACKNOWLEDGEMENTS

REFERENCES

[1] Nabiha Asghar, Pascal Poupart, Jesse Hoey, Xin Jiang, and Lili Mou. 2018. Affective neural response generation. In *European Conference on Information Retrieval*. Springer, 154–166.

[2] Chandrakant Bothe, Sven Magg, Cornelius Weber, and Stefan Wermter. 2017. Dialogue-based neural learning to estimate the sentiment of a next upcoming utterance. In *International Conference on Artificial Neural Networks*. Springer, 477–485.

[3] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. arXiv:1810.04805 [cs.CL]

[4] Emily Dinan, Stephen Roller, Kurt Shuster, Angela Fan, Michael Auli, and Jason Weston. 2018. Wizard of wikipedia: Knowledge-powered conversational agents. *arXiv preprint arXiv:1811.01241* (2018).

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

[6] William Falcon and The PyTorch Lightning team. 2019. PyTorch Lightning. https://doi.org/10.5281/zenodo.3828935

[7] Deepanway Ghosal, Navonil Majumder, Alexander Gelbukh, Rada Mihalcea, and Soujanya Poria. 2020. COSMIC: COmmonSense knowledge for eMotion Identification in Conversations. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: Findings*. 2470–2481.

[8] Xu Han, Weilin Zhao, Ning Ding, Zhiyuan Liu, and Maosong Sun. 2021. PTR: Prompt Tuning with Rules for Text Classification. *arXiv preprint arXiv:2105.11259* (2021).

[9] Jonathan Herzig, Guy Feigenblat, Michal Shmueli-Scheuer, David Konopnicki, Anat Rafaeli, Daniel Altman, and David Spivak. 2016. Classifying emotions in customer support dialogues in social media. In *Proceedings of the 17th annual meeting of the special interest group on discourse and dialogue*. 64–73.

[10] Jeremy Howard and Sebastian Ruder. 2018. Universal Language Model Fine-tuning for Text Classification. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. 328–339.

[11] Chao-Chun Hsu, Sheng-Yeh Chen, Chuan-Chun Kuo, Ting-Hao Huang, and Lun-Wei Ku. 2018. EmotionLines: An Emotion Corpus of Multi-Party Conversations. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*. European Language Resources Association (ELRA), Miyazaki, Japan. https://www.aclweb.org/anthology/L18-1252

[12] Chenyang Huang, Osmar R Zaiane, Amine Trabelsi, and Nouha Dziri. 2018. Automatic dialogue generation with expressed emotions. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*. 49–54.

[13] Yen-Hao Huang, Ssu-Rui Lee, Mau-Yun Ma, Yi-Hsin Chen, Ya-Wen Yu, and Yi-Shin Chen. 2019. EmotionX-IDEA: Emotion BERT–an Affectional Model for Conversation. *arXiv preprint arXiv:1908.06264* (2019).

[14] Jeff Johnson, Matthijs Douze, and Hervé Jégou. 2019. Billion-scale similarity search with gpus. *IEEE Transactions on Big Data* (2019).

[15] Daniel Jurafsky and James H. Martin. 2009. *Speech and Language Processing (2nd Edition)*. Prentice-Hall, Inc., USA.

[16] Sopan Khosla. 2018. EmotionX-AR: CNN-DCNN autoencoder based emotion classifier. In *Proceedings of the sixth international workshop on natural language processing for social media*. 37–44.

[17] Jonggu Kim, Hyeonmok Ko, Seoha Song, Saebom Jang, and Jiyeon Hong. 2020. Contextual Augmentation of Pretrained Language Models for Emotion Recognition in Conversations. In *Proceedings of the Third Workshop on Computational Modeling of People's Opinions, Personality, and Emotion's in Social Media*. 64–73.

[18] Diederik P. Kingma and Jimmy Ba. 2015. Adam: A Method for Stochastic Optimization. In *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*.

[19] Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, et al. 2020. Retrieval-augmented generation for knowledge-intensive nlp tasks. *arXiv preprint arXiv:2005.11401* (2020).

[20] Xiang Li and Ming Zhang. 2018. Emotion Analysis for the Upcoming Response in Open-Domain Human-Computer Conversation. In *Asia-Pacific Web (APWeb) and Web-Age Information Management (WAIM) Joint International Conference on Web and Big Data*. Springer, 352–367.

[21] Yanran Li, Hui Su, Xiaoyu Shen, Wenjie Li, Ziqiang Cao, and Shuzi Niu. 2017. DailyDialog: A Manually Labelled Multi-turn Dialogue Dataset. arXiv:1710.03957 [cs.CL]

[22] Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. RoBERTa: A Robustly Optimized BERT Pretraining Approach. arXiv:1907.11692 [cs.CL]

[23] Nurul Lubis, Sakriani Sakti, Koichiro Yoshino, and Satoshi Nakamura. 2018. Eliciting positive emotion through affect-sensitive dialogue response generation: A neural network approach. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 32.

[24] Albert Mehrabian. 1996. Pleasure-arousal-dominance: A general framework for describing and measuring individual differences in temperament. *Current Psychology* 14, 4 (1996), 261–292.

[25] Nikita Moghe, Siddhartha Arora, Suman Banerjee, and Mitesh M Khapra. 2018. Towards Exploiting Background Knowledge for Building Conversation Systems. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*.

[26] Rita Parada Ramos, Patrícia Pereira, Helena Moniz, Joao Paulo Carvalho, and Bruno Martins. 2021. Retrieval Augmentation for Deep Neural Networks. In *2021 International Joint Conference on Neural Networks (IJCNN)*. IEEE, 1–8.

[27] Hannah Rashkin, Eric Michael Smith, Margaret Li, and Y-Lan Boureau. 2018. I know the feeling: Learning to converse with empathy. *ArXiv* abs/1811.00207 (2018).

[28] Hannah Rashkin, Eric Michael Smith, Margaret Li, and Y-Lan Boureau. 2019. Towards Empathetic Open-domain Conversation Models: A New Benchmark and Dataset. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*. 5370–5381.

[29] Nils Reimers and Iryna Gurevych. 2019. Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics. https://arxiv.org/abs/1908.10084

[30] Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. 2014. Dropout: a simple way to prevent neural networks from overfitting. *The journal of machine learning research* 15, 1 (2014), 1929–1958.

[31] Oriol Vinyals and Quoc Le. 2015. A Neural Conversational Model. arXiv:1506.05869 [cs.CL]

[32] Xinyu Wang, Yong Jiang, Nguyen Bach, Tao Wang, Zhongqiang Huang, Fei Huang, and Kewei Tu. 2021. Improving Named Entity Recognition by External Context Retrieving and Cooperative Learning. *arXiv preprint arXiv:2105.03654* (2021).

[33] Zhiyuan Wen, Jiannong Cao, Ruosong Yang, Shuaiqi Liu, and Jiaxing Shen. 2021. Automatically Select Emotion for Response via Personality-affected Emotion Transition. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*. 5010–5020.

[34] Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Perric Cistac, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander M. Rush. 2020. Transformers: State-of-the-Art Natural Language Processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*. Association for Computational Linguistics, 38–45. https://www.aclweb.org/anthology/2020.emnlp-demos.6

[35] Thomas Wolf, Victor Sanh, Julien Chaumond, and Clement Delangue. 2019. Transfertransfo: A transfer learning approach for neural network based conversational agents. *arXiv preprint arXiv:1901.08149* (2019).

[36] Yubo Xie and Pearl Pu. 2021. Generating Empathetic Responses with a Large Scale Dialog Dataset. *arXiv preprint arXiv:2105.06829* (2021).

[37] Ali Yadollahi, Ameneh Gholipour Shahraki, and Osmar R Zaiane. 2017. Current state of text sentiment analysis from opinion to emotion mining. *ACM Computing Surveys (CSUR)* 50, 2 (2017), 1–33.

[38] Peixiang Zhong, Di Wang, and Chunyan Miao. 2019. Knowledge-enriched transformer for emotion detection in textual conversations. *arXiv preprint arXiv:1909.10681* (2019).

[39] Hao Zhou, Minlie Huang, Tianyang Zhang, Xiaoyan Zhu, and Bing Liu. 2018. Emotional chatting machine: Emotional conversation generation with internal and external memory. In *Thirty-Second AAAI Conference on Artificial Intelligence*.