Creating Corpora for Seq2Seq Tone Rephrasing Using Social Media Posts

Paulo Cavalin
IBM Research
Rio de Janeiro, Brazil
pcavalin@br.ibm.com

Marisa Vasconcelos
IBM Research
São Paulo, Brazil
marisaav@br.ibm.com

Marcelo Grave
IBM Research
São Paulo, Brazil
marcelo.grave@ibm.com

Claudio Pinhanez
IBM Research
São Paulo, Brazil
csantosp@br.ibm.com

***Abstract*—We present a methodology to use Twitter posts to create a parallel corpus which can be used to train Seq2Seq neural networks for a tone rephrasing task. Given that people tend to post texts expressing opinions or emotions of varied intensities regarding given real-world events, the main idea is to create corpus containing pairs of posts with opposite tone but about the same topic. By doing so we overcome the main limitation of current tone rephrasing methods: the lack of appropriate parallel training corpora. We explore different methods to create the datasets, including some which require some level of manual labelling. The results show that a completely automatic generation from Twitter data yields training datasets which are better than those with manual interventions, and good enough for Seq2Seq models to outperform non-Seq2Seq models trained with similar data.**

***Index Terms*—Seq2seq, corpora, rephrasing, social media**

I. Introduction

In this paper, we explore the use of Twitter data to generate parallel corpora to be used in the training of *Seq2Seq* models [1], [2], focusing on the task of tone rephrasing. Tone rephrasing can be defined as converting a text presenting a tone τ_1 to another tone τ_2, while keeping the meaning of the text. For instance, tone rephrasing can be the conversion of a text that presents a negative sentiment to another text that will present a positive sentiment, while the underlying meaning of the text will remain unchanged.

Our interest on tone rephrasing is, first and foremost, because it is not a simple task for Seq2Seq systems, since the algorithm has to strike a fine balance between changing some words, re-structuring part of the sentence, but at the same time keeping the basic meaning intact. In addition, we also see an increasing number of applications which need “fine tuning” of verbal tone, ranging from social media filters to conversational agents. Although there is a lot of work on detecting negative and hate speech in social media, most of it seems to be directed at removing those posts. If good rephrasing systems were available, social media conversation could be toned down (perhaps even to make it appropriate for younger audiences) without the risks of full censorship of ideas.

Over recent years, Seq2Seq neural networks have drawn significant attention in language-to-language translation problems [1], [2]. Rephrasing tasks can also potentially benefit from such type of approach, although currently this faces limitations because of the lack of sufficiently large datasets. Other approaches [3]–[5] based on algorithms which learn on non-parallel datasets (e.g., *Set2Set)* usually lack of a more fine-grained parallel comparison of examples, and tend to fail in generating more complex rephrasings, since they are likely to produce mostly vocabulary alterations.

Social media platforms have been used as datasets in a variety of domains such as sentiment analysis classification tasks. On Twitter, for instance, people tend to react with different emotions to real-world events, and to express themselves by means of a corresponding tone [6], [7]. For example, during a presidential election, some people may post text supporting some of the candidates, while other people may write posts against him/her, sometimes violently. Given that users can express quite different emotions regarding the same topic, by means of changes in the tone, we decided to explore whether, by processing Twitter data with appropriate tools, one could generate corpora which might be used to train Seq2Seq models for tone rephrasing, and whether the accuracy of Seq2Seq using such corpora would surpass that of Set2Set approaches.

In summary, in this paper, we describe and test an approach for building parallel corpora from Twitter posts. The method takes as input a set of individual posts (tweets) and presents as outputs a corpus containing pairs of posts, where each pair contains two posts with opposite tones. We here focus on rephrasing from a positive to negative tone and the other way around. However, if tone classifiers which can deal with other classes of tones are available, our approach can be extended to other types of tone conversions.

In the proposed corpora building method, we take advantage of the time window of controversial topics, semantic similarity methods, and tone classification tools such as sentiment analysis. More specifically, given a post from a selected controversial topic, we build a cluster of texts based on finding other posts with similar content, given a pre-defined time window. Once that cluster is built, tone classification is applied on each post, and then by pairing each post of a cluster with other from the the opposite tone we generate a set of pairs which can be used to train Seq2Seq systems. Further post-filtering can be applied to those pairs, in order to eliminate those that present too different structures. By scaling up this processing to a collection of topics, a large corpus can be built.

By means of experiments comparing different implementations of the aforementioned methodology, we argue in this pa-

978-1-7281-6926-2/20/$31.00 ©2020 IEEE

per that the proposed approach is promising since the Seq2Seq models we created have been able to beat both a proposed baseline and a state-of-the-art Set2Set method in numerous metrics. In addition, we also show that by making use of publicly-available tone classification tools, the methodology is scalable to larger sets of social media data for building even larger training corpora.

II. Related Work

Several methods have been proposed for converting a text to another. In recent years, great progress has been made with deep learning for tasks such as translation [2] and paraphrasing [1]. While in translation a text needs to be converted from one language to another, in paraphrasing, the text needs to be re-written to another one, with different words and grammatical structure. In both cases, though, both the meaning and the tone (tone, mode, style, and sentiment are terms which possess similar meaning, but we use simply tone hereafter) should be kept the same, since there is no intention in changing that from the input.

There are many applications which can benefit if text changes its meaning or tone but preserving the basic semantics. Converting from one tone to another has different applications and interest in approaches for carrying out such task has emerged in the recent years [3]–[5]. Some work has been done in this area, for instance in the conversion of offensive language to non-offensive [5], and the generation of customizable affective text [3].

In conversational systems, it is widely accepted that people recognize and assign emotions to computers [8]. In many contexts, it is desirable to change the way information is given to users to match the users' emotional state. For example, [9] found, in an experiment using driving simulators, that if there is a mismatch between the mood of the driver (induced artificially before the experiment) and the emotions expressed by the voice of the car assistant, not only likeness of the car assistant system decreases, but the number of accidents double. Similarly, adaptive dialog systems may be improved if there is technology which converts a neutral version of an utterance into a more thankful, apologetic, or assertive version.

In general, people have a remarkable preference towards interacting with and evaluating positively people who are similar to them [10], and the same is true when interacting with computers [8], [9]. Although, it is relatively simple to create different versions of a text with specific expressive emotional traits, this is often a time-consuming task which also can not be used for real-time data. We see thus, an increasing demand for automatic tone rephrasing systems such as the ones discussed in this paper.

Differently from translation and paraphrasing, tone rephrasing suffers from the lack of parallel corpora (that is, different versions of the same text rephrased in different tones) to train end-to-end deep learning methods. As a consequence, both corpora and approaches proposed for the task have been generally non-parallel [3]–[5].

Therefore, creating better corpora for tone rephrasing is key for further progress in this area. We believe that both the research community and industry can benefit from approaches which can improve the process of creating corpora for style transfer. Such approaches can be focused not only on generating more precise training sets, either parallel or non-parallel, but also on scaling up to the larger portfolio of tones needed in commercial applications.

TABLE I
Statistics of the Twitter data

Number of matches	6
Total number of tweets	1,772,999
Total number of RTs	471,987
Average tweet length (chars)	100.6
Average tweet length (words)	15.20

III. Twitter Data

We use Twitter social data publicly shared during the FIFA 2018 World Cup as the source for the dataset used in this work. Sports datasets are in general rich in terms of the high level of emotions and opinions, since the fans' posts and Twitter are often a back-channel reflecting their reactions to a live broadcast [6].

We collected tweets using keyword related to the event such as the names of the players, a set of soccer-related words (e.g., "penalty", "kick", "goal"), the names of the countries involved in each match, and the official hashtags of the tournament (e.g., #WorldCup2018,#fifaworldcup2018), using the sampling method provided by the *Twitter Streaming API*. For each match, the collection period lasted three hours, starting half an hour before the beginning of the game. For this work, we use data from 6 matches: Brazil vs. Costa Rica, Denmark vs. Australia, England vs. Tunisia, France vs. Uruguay, South Korea vs. Germany, and Russia vs. Egypt. All the matches summed up 1.8 million tweets (including up to 470 thousand retweets). We chose those games because they presented some of the most controversial events of the tournament [11]. Greater detail about the collected data is presented in Table I.

Figure 1 shows the sentiment polarity distribution for Brazil vs. Costa Rica match. Notice that even being a match with many controversial moments, almost 45% of tweets were classified as neutral while the remaining tweets are distributed over negative and positive polarity values.

IV. Corpora Building Method

Our proposed methodology consists of several processing stages applied to a corpus of social media posts, which, in the end, result in a corpus of pairs of texts which can be used to train and evaluate Seq2Seq machine learning models for tone rephrasing. Figure 2 illustrates the main stages of our approach.

As a first step, we collect data from Twitter (as described in the previous section) using keywords to query the Streaming

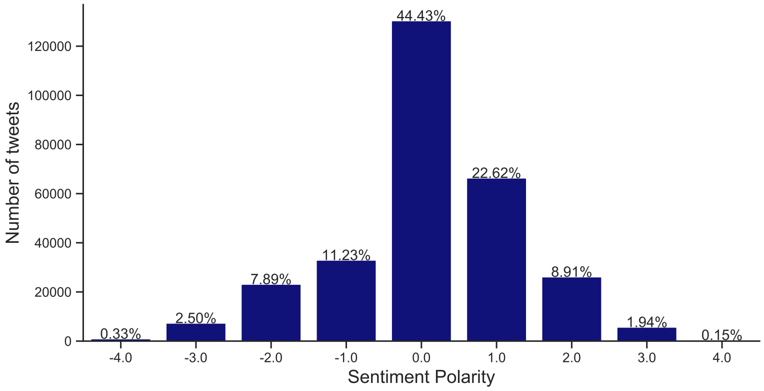
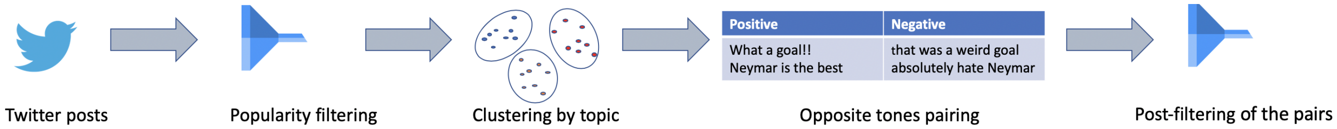


Fig. 1. Sentiment Polarity of Brazil vs. Costa Rica.



Positive	Negative
What a goal!! Neymar is the best	that was a weird goal absolutely hate Neymar

Fig. 2. Proposed Methodology for Corpus Creation

API for a large event, in our case, soccer matches. After collecting all data, we select the most posted tweets (e.g., retweets) of a match and then perform a search for texts with similar semantic content to create clusters of posts. That search is based on semantic similarity computation, for which we make use of a *word embedding similarity*, with word embeddings built from a combination of Wikipedia and Twitter posts. The goal of this step is to build clusters of posts with share similar content, i.e. the same topic, to those of the keys ones (e.g., most popular). Before computing similarity, though, the texts are pre-processed by means of normalization steps such as removal of stop-word and converting entity names, such as country and player name, to a single special token defined as *ENTITY*.

Once the clusters have been computed, the next step lies in assigning a tone class or a tone score to each post in the cluster. In this work we apply sentiment analysis by means of the *SentiStrength* tool [12]. SentiStrength is a well-established method which implements a combination of learning methods to assess the sentiment strength of the opinion in social media posts. Given a piece of text, the tool returns a score between -4 (negative polarity) and +4 (positive polarity). We performed an evaluation of the quality of the classification provided by the tool in our datasets and we found a precision and recall of about 91% which makes it suitable for the purposes of this paper.

Once the tone of each sentence has been computed, we perform the Cartesian product of the sets of posts with opposite sentiments (e.g., positive and negative) within each cluster (i.e., posts with similar semantic content), and generate the first corpus with pairs of posts, which we denote as the *CP* corpus.

On the *CP* corpus, we apply different filtering methods which although produce smaller corpora, create cleaner training sets. In other words, we narrow down the resulting corpus to contain only more similar pairs of text. We consider two different filtering approaches. The first one consists of keeping only the most similar pairs by considering the *cosine similarity distance* between the *bag-of-word vectors* built on each pair of posts. Given that the word embedding-based similarity focuses more in semantic similarity, the bag of words-based similarity puts more weight on lexical comparisons and thus complements the other method. In addition, we consider a second type of filtering based on *Jaccard similarity distance*.

V. Corpora Building Method Validation

In this section, we present a validation of the proposed methods, considering both the Seq2Seq model and a baseline method. The idea is to evaluate which methods perform best to create corpora of pairs of texts, and also to understand whether the Seq2Seq model is promising for tone rephrasing or not compared with the baseline.

As we saw, there are different strategies to create a corpus depending on how the initial set of clusters of posts (cartesian product pairs) is filtered. In this paper we consider the following six different methods of creating a parallel corpus from a set of Twitter posts, including some cases where manual filtering was also used:

- **CP SentiStrg**: generated using the cartesian product of posts from the same cluster but with opposite sentiments, computed only with the SentiStrength tool;

TABLE II
NUMBER OF PAIRS AND AVERAGE SENTIMENT IN EACH DATASET COMPUTED FROM THE SOCCER GAMES.

Dataset	**Pairs**	S^+	S^-	ΔS
CP SentiStrg	54,004	1.35	-1.66	3.01
CP SentiStrg fltr	1,593	1.29	-1.69	2.98
CP revised	58,040	0.81	-1.16	1.97
CP manual fltr	28,454	0.73	-1.18	1.91
Fltr Similarity	1,335	0.49	-1.17	1.66
Fltr Jaccard	409	0.63	-0.82	1.45

- **CP SentiStrg fltr**: generated by filterin the set of pairs from *CP SentiStrg* with a second pass of similarity, using bag-of-words and cosine distance;
- **CP revised**: similar to *CP SentiStrg*, but in this case the sentiment labels of posts in the clusters are manually inspected and corrected if there was some inconsistency, prior to generate the corpus;
- **CP manual fltr**: similar to *CP revised*, but with a second pass of manual inspection to remove inconsistent pairs of posts (for instance, pairs of text with the same sentiment);
- **Filtered Similarity**: similar to *CP SentiStrg fltr*, but with a second pass of similarity filtering is applied to *CP revised*;
- **Filtered Jaccard**: similar to *Filtered Similarity*, but with the second similarity filter applied on *CP revised* being the Jaccard similarity.

All methods had as input four clusters of posts, with a total of 3,666 posts and an average of 916 post per cluster. From those, a total of 1,239 were recognized by the SentiStrength tool with positive sentiment, 717 negative, and the remaining ones neutral. It is worth mentioning that those sentiment ratings were used for both *CP SentiStrg* and *CP SentiStrg fltr*. For the other datasets, the ratings were also manually corrected, resulting in 1,484 positive and 704 negative poste.

In Table II we present a summary of each of the evaluation datasets, containing the number of pairs generated, the average positive and negative sentiment, denoted S^+ and S^-, and the difference between S^+ and S^-, denoted ΔS. Note that, the largest the value of ΔS, the more opposite are the average sentiment in the set. That said, we observe that the set with largest ΔS is *CP SentiStrg*, but its corresponding filtered version, i.e. *CP SentiStrg fltr*, has also a similar range of sentiment. The manual filtering yielded a decrease in ΔS of about one third, which was further reduced by the additional filtering used in the last 4 datasets.

A. Evaluation Metrics

We use four evaluation metrics to assess the quality of the generated sentences considering the different aspects and challenges of the rephrasing task as discussed before. The metrics are:

Novelty (Nvlt): we want to assess how different the generated sentence is from the input sentence, i.e., how much of the input has been actually rephrased. Based on the metric described in *Wang et al.* [13], we computed the novelty of each generated sentence using the *Jaccard distance* φ as:

$$Nvlt(G_i) = 1 - \varphi(G_i, I_i),$$

where G_i is the generated sentence and I_j is the sentence used as input.

Content Preservation (Prvt): we use the content preservation metric proposed by *Fu et al.* [14] which evaluates the similarity between the training sentences and the generated sentences. It is defined extracting features from word embeddings between sentences from the training set and sentences in the test set[1].

BLEU score: we used the *BLEU score* to assess the similarity between ground-truth candidate sentences and the generated sentence [15]. It is a score between 0 and 1 which is computed counting matching n-grams in the candidate sentence to n-grams in the generated sentence[2].

Sentiment Conversion Metrics: given that it might not be trivial to generate ground-truth data and evaluate text conversion approaches in this particular application using standard metrics, we defined a metrics which evaluates the text conversion capability. Given the input sentiment S_{in}, the expected sentiment S_{exp}, and the generated sentiment S_{gen}, the *sentiment conversion ratio* ΔS_{ratio} is computed as:

$$\Delta S_{ratio} = \frac{\Delta S_{gen}}{\Delta S_{exp}} \times 100,$$

where $\Delta S_{gen} = S_{gen} - S_{in}$ and $\Delta S_{exp} = S_{exp} - S_{in}$ are the change in tone produced by the algorithm, and the change in tone expected to be achieved, respectively. Notice that $S_{in} = S^+$ and $S_{exp} = S^-$ if we are converting from positive to negative tone, and the opposite when rephrasing from negative to positive.

With ΔS_{ratio} we expect to be able to measure how close to the expected sentiment a given method has been able to get. In that case, values close to 100 means that the generated sentiment is very close to the expectation. Also, other values can indicate some exaggeration in the conversion, i.e. for values greater than 100, or even the incapacity to do any conversion with negative values.

B. The Baseline and Seq2Seq Models

For the evalution of the six different methods to generate the training corpora, two different methods are considered: a Seq2Seq neural network and a standar rule-based baseline tone rephrasing system we implemented. The baseline method consisted of replacing each adjective respectively found in every sentence of the test set by one of its antonyms randomly selected from a valid set. For this, we employed the intersection of all antonyms found for each adjective using the *WordNet synsets*[3], and all words of the training set are

[1] We use the https://github.com/fuzhenxin/textstyletransferdata implementation.

[2] We use the function sentence *bleu* from the *NTLK* Python package.

[3] We used the WordNet synsets from *NLTK* Python package.

TABLE III

RESULTS WITH THE BASELINE, WHERE IN BOLD WE HIGHLIGHT THE BEST RESULTS, AND UNDERLINED WE HIGHLIGHT CASES WHERE THE TRANSFORMATION EXCEEDED THE EXPECTED SENTIMENT.

Positive to negative (Baseline)									
Dataset	S_{in}	S_{exp}	ΔS_{exp}	S_{gen}	ΔS_{gen}	ΔS_{ratio}	Nvlt	Prvt	BLEU
CP SentiStrg	1.35	-1.66	-3.01	0.83	-0.52	17.3%	**0.09**	0.99	0.76
CP SentiStrg fltr	1.29	-1.69	-2.98	0.86	-0.43	14.4%	0.08	0.99	0.76
CP revised	0.81	-1.16	-1.97	0.44	-0.37	18.8%	0.07	0.99	0.75
CP manual fltr	0.73	-1.18	-1.91	0.12	-0.61	**31.9%**	0.07	0.99	0.78
Fltr Similarity	0.49	-1.17	-1.66	0.13	-0.36	21.7%	0.06	0.98	0.78
Fltr Jaccard	0.63	-0.82	-1.45	0.48	-0.15	10.3%	0.03	0.99	**0.82**
Negative to positive (Baseline)									
Dataset	S_{in}	S_{exp}	ΔS_{exp}	S_{gen}	ΔS_{gen}	ΔS_{ratio}	Nvlt	Prvt	BLEU
CP SentiStrg	-1.66	1.35	3.01	-1.31	0.35	11.6%	0.07	0.99	0.75
CP SentiStrg fltr	-1.69	1.29	2.98	-1.24	0.45	15.1%	0.07	0.99	0.80
CP revised	-1.16	0.81	1.97	-0.86	0.30	15.3%	**0.08**	0.99	0.78
CP manual fltr	-1.18	0.73	1.91	-0.90	0.28	14.6%	0.07	0.99	0.78
Fltr Similarity	-1.17	0.49	1.66	-0.86	0.31	**18.7%**	0.06	0.99	**0.83**
Fltr Jaccard	-0.82	0.63	1.45	-0.71	0.11	7.6%	0.02	0.99	0.82

TABLE IV

RESULTS WITH THE SEQ2SEQ MODEL, WHERE IN BOLD WE HIGHLIGHT THE BEST RESULTS, AND UNDERLINED WE HIGHLIGHT CASES WHERE THE TRANSFORMATION EXCEEDED THE EXPECTED SENTIMENT.

Positive to negative (Seq2Seq)									
Dataset	S_{in}	S_{exp}	ΔS_{exp}	S_{gen}	ΔS_{gen}	ΔS_{ratio}	Nvlt	Prvt	BLEU
CP SentiStrg	1.35	-1.66	-3.01	-0.46	-1.81	60.1%	**0.94**	0.59	0.71
CP SentiStrg fltr	1.29	-1.69	-2.98	-1.61	-2.90	**97.3%**	0.86	0.78	0.77
CP revised	0.81	-1.16	-1.97	-0.79	-1.60	81.2%	**0.94**	0.60	0.72
CP manual fltr	0.73	-1.18	-1.91	-0.31	-1.04	54.5%	0.93	0.55	0.72
Fltr Similarity	0.49	-1.17	-1.66	-1.40	-1.89	113.8%	0.88	0.74	**0.78**
Fltr Jaccard	0.63	-0.82	-1.45	-0.91	-1.54	106.2%	**0.88**	0.81	0.76
Negative to positive (Seq2Seq)									
Dataset	S_{in}	S_{exp}	ΔS_{exp}	S_{gen}	ΔS_{gen}	ΔS_{ratio}	Nvlt	Prvt	BLEU
CP SentiStrg	-1.66	1.35	3.01	0.36	2.02	67.1%	0.93	0.56	0.71
CP SentiStrg fltr	-1.69	1.29	2.98	1.04	2.73	**91.6%**	0.82	**0.7**8	0.78
CP revised	-1.16	0.81	1.97	0.01	1.17	59.7%	**0.94**	0.71	0.76
CP manual fltr	-1.18	0.73	1.91	0.00	1.18	62.8%	**0.94**	0.50	0.74
Fltr Similarity	-1.17	0.49	1.66	0.31	1.48	89.1%	0.80	0.77	**0.81**
Fltr Jaccard	-0.82	0.63	1.45	0.73	1.55	106.8%	**0.94**	0.59	0.71

considered as part of the valid set. Part-of-speech tagging [4] was used as a manner of extracting the adjectives from previously cleaned sentences.

For the Seq2Seq model, we implemented an *Encoder-Decoder* model with local attention [16], [17]. The *Recurrent Neural Networks (RNN)* for both the encoder and the decoder were implemented with a single embedding layer, and a *Gate-Recurrent Unit* layer, both with 256 hidden neurons, and dropout of 0.1. The models were trained for 25,000 epochs, applying teacher forcing with a ratio of 0.5, and the learning rate was set to 0.01. It is worth mentioning that determining which was the best method for implementing a Seq2Seq model was out of the scope of this work, and thus we focused on making use of a simple architecture in order to be able to train it even on small training sets.

C. Experimental Results

In Table III and Table IV, we present the results of the baseline and the Seq2Seq models, respectively, on each of the constructed datasets, for both *Positive to Negative* and *Negative to Positive* rephrasing tasks. For this evaluation, each dataset was randomly split into 70% of its samples for training and 30% for training, which allowed to directly and fairly compare the Seq2Seq model against the baseline.

The first clearly observed result is that the Seq2Seq model outperforms consistently the results of the baseline. By taking into account ΔS_{ratio}, the highest value achieved by the baseline is of 31.9%, while the Seq2Seq model reaches 54.5% in the same dataset, which is in fact the lowest ΔS_{ratio} achieved by the latter. In fact, the Seq2Seq model is able, with one of the tested datasets, to get to as high as 97.3% of ΔS_{ratio}, that is, almost achieving the full expected tone rephrasing.

Also, Seq2Seq outperforms by a high margin in Novely

[4] We used part-of-speech tagging also from the NLTK Python package.

TABLE V
COMPARISON OF DIFFERENT METHODS ON THE COMMON TEST, WHERE IN BOLD WE HIGHLIGHT THE BEST RESULTS, AND UNDERLINED WE HIGHLIGHT CASES WHERE THE TRANSFORMATION EXCEEDED THE EXPECTED SENTIMENT

Positive to negative									
Method	S_{in}	S_{exp}	ΔS_{exp}	S_{gen}	ΔS_{gen}	ΔS_{ratio}	Nvlt	Prvt	BLEU
Baseline	0.97	-1.40	-2.37	0.63	-0.34	14.3%	0.05	**0.99**	**0.77**
Seq2Seq	0.97	-1.40	-2.37	-1.07	-2.04	**86.1%**	0.74	0.82	0.77
Set2Set	0.97	-1.40	-2.37	-0.39	-1.36	57.4%	0.69	0.75	0.67
Set2Set-Yelp	0.97	-1.40	-2.37	-0.69	-1.66	70.0%	**0.91**	0.79	0.74
Negative to positive									
Method	S_{in}	S_{exp}	ΔS_{exp}	S_{gen}	ΔS_{gen}	ΔS_{ratio}	Nvlt	Prvt	BLEU
Baseline	-1.40	0.97	2.37	-1.07	0.33	13.4%	0.04	**0.99**	**0.84**
Seq2Seq	-1.40	0.97	2.37	0.09	1.49	**62.8%**	0.73	0.77	0.81
Set2Set	-1.40	0.97	2.37	-0.39	1.01	42.6%	0.80	0.72	0.72
Set2Set-Yelp	-1.40	0.97	2.37	1.94	3.34	140.9%	**0.90**	0.80	0.77

(Nvlt) but loses in Content Preservatio (Prvt). Nevertheless, the very high values of Prvt presented by the baseline indicate the poor ability of such approach for rephrasing. In terms of the BLEU score, we observe that the results are quite mixed, corroborating to our claim in the difficulty of generating ground-truth texts for this task.

Regarding the different datasets, it was very surprising that we have not been able to observe any substantial improvement in the metrics after we manually validated the sentiment classification. The best performances of the Seq2Seq model were achieved with the *CP SentiStrength Filtered* dataset, with 97.3% and 91.6% of ΔS_{ratio} for positive to negative and negative to positive rephrasing, respectively[5].

Those results may mean that although the SentiStrength tool may produce some classification errors, its accuracy is adequate to generate a corpus in an fully automated if proper filtering is done. In addition, we observe that filtering the datasets may results in corpora from which the Seq2Seq model can learn better, compared with a larger but noisier training sets.

VI. COMPARING SEQ2SEQ AND SET2SET

In order to verify our hypothesis that Seq2Seq models may outperform Set2Set models, provided there is an appropriate corpus, in this section we compare the results of the Seq2Seq model against the Set2Set approach presented in [4], and their performance in relation to the baseline model described before.

For both Seq2Seq and the baseline, we employ the *CP SentiStrg fltr* dataset, but here using all its 1,593 pairs for training. And for the Set2Set system, we consider two training sets: a) the *CP SentiStrg fltr* dataset, where the pairs are decomposed back into two distinct sets, i.e. one for each sentiment; and b) the *Yelp* dataset, which is a much larger corpus, containing a total of 444,010 samples. We refer to those two versions of Set2Set as *Set2Set* and *Set2Set-Yelp*, respectively.

For comparing the methods, we created a test set with 1,764 pairs by utilizing the same method used to create *CP SentiStrg fltr*, but applying it onto another set of clusters extracted from other 2018 World Cup matches which were not used for the evaluation presented in the previous section.

The results presented in Table V show that the Seq2Seq model achieves the largest value for ΔS_{ratio} in the positive to negative task. Similarly, Seq2Sew also achieves the highest ΔS_{ratio} in the negative to positive task, considering that the ΔS_{ratio} presented by Seq2Seq is closer to 100% than the ΔS_{ratio} presented by Set2Set-Yelp, i.e. 37.2% versus 40.9%, showing that the latter over-exaggerate in the conversion. Interestingly, we observe that Set2Set-Yelp has better results both in Nvlt and Prvt than Set2Set. This may be due to the larger training set of Set2Set-Yelp, though.

In Table VI we present some examples of rephrasing provided by each method. In those examples we can see that the Seq2Seq model generally produces well-formed sentences, even though a slight drift in meaning is observed in one of the examples. The two examples show the major weakness of the baseline method, which is the inability to rephrase in some cases. And with Set2Set and Set2Set-Yelp, we observe that the sentences can be well formed, but the meaning seems to have drifted too much from the input sentence.

In Table VII we present some of the errors made by the Seq2Seq model. Some of the mistakes are either by being unable to generate a grammatically correct sentence, as the first one, to the repeated rephrasing such as the second one which is the same as in Table VI, and even in the generation of meaningless sentences such as the third one. We believe, however, that by scaling up the dataset (for instance, by considering all matches of the World Cup) and generating large training corpora, those mistakes can be minimized.

VII. CONCLUSION

In this paper we presented the evaluation of different methods to use Twitter data to create corpora to train Seq2Seq neural networks for sentiment polarity transformation. The results obtained indicate that the methodology basically works and can be further improved by using larger input datasets and thus obtaining larger training sets, which are likely to produce better rephrasing models.

[5] We considered both undershooting ($\Delta S_{ratio} < 0\%$) and overshooting ($\Delta S_{ratio} > 100\%$) as an indication of possible problems with the model.

TABLE VI
SAMPLES OF REPHRASING PRODUCED BY THE EVALUATED METHODS.

	Positive to negative	Sentiment	Nvlt	Prvt
Input	France is winning this World Cup	1		
Baseline	*France is winning this World Cup*	1	0.00	1.0
Seq2Seq	*France has done nothing this world cup but flop and complain*	-1	0.86	0.78
Set2Set	*France has been so so far*	0	0.77	0.83
Set2Set-Yelp	*This is one of France*	0	0.77	0.87
	Negative to positive	S	Nvlt	Prvt
Input	France is losing. what a world cup	-2		
Baseline	*France is losing. what a world cup*	-2	0.00	1.0
Seq2Seq	*France has the world cup*	0	0.70	0.94
Set2Set	*France ricakeeper has been well good.*	1	0.75	0.77
Set2Set-Yelp	*France is what a mistake is what a oil.*	-1	0.50	0.86

TABLE VII
SAMPLES OF UNSUCCESSFUL REPHRASINGS BY SEQ2SEQ.

Input	Generated
This world cup!!	*ENTITY has done this this this so bad*
Interesting world cup	*ENTITY has done nothing this world cup but flop and complain*
This year FIFA world cup is disappointing	*ENTITY is*

Thus, as future work we shall focus both in processing more data and from other domains, exploring different corpora representing a varied set of topics. Additionally, we shall evaluate other Seq2Seq and attention-based models, and evaluate their impact on different rephrasing tasks. Similarly, once larger training sets become available, we shall focus on enhancing the current Seq2Seq architectures to tailor them specifically for tone rephrasing.

REFERENCES

[1] A. Prakash, S. A. Hasan, K. Lee, V. V. Datla, A. Qadir, J. Liu, and O. Farri, "Neural paraphrase generation with stacked residual LSTM networks," *CoRR*, vol. abs/1610.03098, 2016. [Online]. Available: http://arxiv.org/abs/1610.03098

[2] A. Nguyen Le, A. Martinez, A. Yoshimoto, and Y. Matsumoto, "Improving sequence to sequence neural machine translation by utilizing syntactic dependency information," in *Proceedings of the Eighth International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*. Asian Federation of Natural Language Processing, 2017, pp. 21–29. [Online]. Available: http://aclweb.org/anthology/I17-1003

[3] S. Ghosh, M. Chollet, E. Laksana, L. Morency, and S. Scherer, "Affect-lm: A neural language model for customizable affective text generation," *CoRR*, vol. abs/1704.06851, 2017. [Online]. Available: http://arxiv.org/abs/1704.06851

[4] T. Shen, T. Lei, R. Barzilay, and T. S. Jaakkola, "Style transfer from non-parallel text by cross-alignment," *CoRR*, vol. abs/1705.09655, 2017. [Online]. Available: http://arxiv.org/abs/1705.09655

[5] C. N. dos Santos, I. Melnyk, and I. Padhi, "Fighting offensive language on social media with unsupervised text style transfer," *CoRR*, vol. abs/1805.07685, 2018. [Online]. Available: http://arxiv.org/abs/1805.07685

[6] J. Gratch, G. Lucas, N. Malandrakis, E. Szablowski, E. Fessler, and J. Nichols, "Goaalll!: Using sentiment in the world cup to explore theories of emotion," in *Proc. of the ACII*, 2015.

[7] J. Bollen, A. Pepe, and H. Mao, "Modeling Public Mood and Emotion: Twitter Sentiment and Socio-Economic Phenomena," in *Proc. of ICWSM*, 2011.

[8] B. Reeves and C. I. Nass, *The media equation: How people treat computers, television, and new media like real people and places.* Cambridge university press, 1996.

[9] C. I. Nass and S. Brave, *Wired for speech: How voice activates and advances the human-computer relationship*. MIT press Cambridge, MA, 2005.

[10] H. Tajfel, "Social identity and intergroup behaviour," *Information (International Social Science Council)*, vol. 13, no. 2, pp. 65–93, 1974.

[11] T. National, "World cup 2018: Messi overreaction, salahś early exit and 10 worst moments so far," https://www.thenational.ae/sport/football/world-cup-2018-messi-overreaction-salah-s-early-exit-and-10-worst-moments-so-far-1.743271, 2018.

[12] M. Thelwall, K. Buckley, and G. Paltoglou, "Sentiment strength detection for the social web," *J. Am. Soc. Inf. Sci. Technol.*, vol. 63, no. 1, pp. 163–173, Jan. 2012.

[13] K. Wang and X. Wan, "SentiGAN: Generating Sentimental Texts via Mixture Adversarial Networks," in *Proc of the IJCAI*, 2018.

[14] Z. Fu, X. Tan, N. Peng, D. Zhao, and R. Yan, "Style transfer in text: Exploration and evaluation," *CoRR*, vol. abs/1711.06861, 2017. [Online]. Available: http://arxiv.org/abs/1711.06861

[15] K. Papineni, S. Roukos, T. Ward, and W. jing Zhu, "Bleu: a method for automatic evaluation of machine translation," in *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, 2002, pp. 311–318.

[16] K. Cho, B. van Merrienboer, Ç. Gülçehre, F. Bougares, H. Schwenk, and Y. Bengio, "Learning phrase representations using RNN encoder-decoder for statistical machine translation," *CoRR*, vol. abs/1406.1078, 2014. [Online]. Available: http://arxiv.org/abs/1406.1078

[17] M. Luong, H. Pham, and C. D. Manning, "Effective approaches to attention-based neural machine translation," *CoRR*, vol. abs/1508.04025, 2015. [Online]. Available: http://arxiv.org/abs/1508.04025