Frustration Level Annotation in Latvian Tweets with Non-Lexical Means of Expression

Viktorija Leonova[1], Janis Zuters[2]
Department of Computer Science, University of Latvia, 19 Raina Blvd., Riga, Latvia
[1]viktorija.leonova@gmail.com
[2]janis.zuters@lu.lv

Abstract

We present a neural-network-driven model for annotating frustration intensity in customer support tweets, based on representing tweet texts using a bag-of-words encoding after processing with subword segmentation together with non-lexical features. The model was evaluated on tweets in English and Latvian languages, focusing on aspects beyond the pure bag-of-words representations used in previous research. The experimental results show that the model can be successfully applied for texts in a non-English language, and that adding non-lexical features to tweet representations significantly improves performance, while subword segmentation has a moderate but positive effect on model accuracy. Our code and training data are publicly available[1].

1 Introduction

Dramatically increasing data storage and processing capacities have resulted in an explosion of available data, and of potential uses for the many kinds of knowledge or insights that could theoretically be extracted from that data. The development of Web 2.0 has created unprecedented amounts of text, and, in recent decades, images and videos. But the sheer volume of the available data is problematic due to a shortage of human resources (time and attention) available for analyzing or even just browsing through it all, as described in (Verma et al, 2016). From the very start, as soon as machine learning techniques appeared, they were immediately applied to text analysis. Neural networks have proven particularly suitable for such tasks, but to learn to approximate human judgements, these methods generally need a large amount of manually annotated training data to obtain a viable model. Creating such corpora is a very time-consuming task, especially if it has to be done from scratch. For languages with a relatively low number of speakers, such as Latvian, this is a particularly pressing problem, because of the small number of textual corpora available, so that researchers generally need to create their own datasets.

Much of the information of importance to be captured from the Internet is by its nature related to emotions. Recommendation systems, conversational “chat bots”, automated customer support assistants, various tools for analyzing, monitoring or enhancing personal well-being or mental health, not to mention systems for targeting or crafting advertising or marketing messages — all can benefit from understanding the emotional state of the recipient or sender of a given message. But recognition of emotions in text generally requires considerable investment of human resources; and is difficult to automate in principle. Perception of emotions is largely subjective, and the emotions perceived from the same source by different people differ. Further, in written communication, and especially communication in social networks, the words alone do not capture all of the emotional information available in the message. For example, the emotional tone of a sentence may vary depending on whether it was followed by period as the absence of periods is a characteristic feature of internet jargon usage, as shown in (Khalifa, 2020), which may carry a different emotional charge.

In this work, we propose the method for automatic annotation of the emotional charge found in text with the help of non-lexical means of expression — typographical marks, emojis and

[1] Source code is available at https://github.com/Lynx1981/dfrustration/

Proceedings of Recent Advances in Natural Language Processing, pages 814–823
Sep 1–3, 2021.
https://doi.org/10.26615/978-954-452-072-4_093

likewise. In addition, to be able to do so for Latvian, we propose an annotated dataset that was used for model training and its results.

2 Background and Related Works

As machine learning methods, and neural networks, in particular, have developed over the past couple of decades, they naturally started to be applied to recognizing emotions, especially following the demonstrated successes of neural networks in image recognition as in (Giacinto et al., 2016). Emotion recognition in speech, as the easier task, began to appear as early as 2000 in (Nicholson et al., 2020). We begin to see published research on the recognition of emotions in text only beginning with (Alm et al., 2005). The majority of works focus on the classification of text according to its predominant emotion, which means that the text or part of the text was classified as containing one of the basic emotions. The most commonly used is Ekman's emotion classification scheme, described in (Ekman, 1992), which contains six basic emotions: anger, fear, sadness, disgust, surprise, and joy — either in the standard way, or with various extensions, such as in (Yao et al., 2014), or reductions, as in (Lee and Wang, 2015). Another model used to classify emotions, in two variants, is the Russell circumplex model, described in (Russell and Mehrabian, 1977), which is a two- or three-factor model where each emotion is represented in a two-dimensional (valence-arousal) or three-dimensional space, with axes of dominance, valence and arousal. Although there are numerous works that use these emotion classification models – for example, the three-factor model was used in (Parthasaraty and Busso, 2017) and the two-factor model in (Yu et al., 2016) — these are not as common as works using a categorical list of basic emotions.

As computing power and data grew, the classification of emotions began to shift from a purely qualitative to a more quantitative approach: the intensity of emotions, not just the presence of specific emotions in the text, began to be studied. Some authors even create automatic tools for classifying emotions with intensity, for example the Weka package presented in (Bravo-Maquez et al., 2019) for four emotions (anger, fear, joy and sadness), or another opensource emotion computing framework presented in (Duppada and Hiray, 2017) for the same four emotions.

Interestingly, however, none of these classifications include frustration, or dissatisfaction, in the list of emotions, even if the emotion itself is well known to everyone and plays an important role, for example, in assessing quality of service in (Stauss et al., 2005). In customer service, however, 'dissatisfaction' is typically used instead of the word 'frustration', but this does not change the content of the term. There are very few works on annotating frustration, and they are relatively old, for example, (Klein et al., 2002) and (Hone, 2006), where the authors discussed the possibility of reducing human frustration in dialogue with the help of an emotional or empathic agent. In (Kapoor et al., 2007), the authors achieved good results in identifying frustration, but they used a complex multimodal system that measured the physical parameters of human (in their case, student) behavior, such as the pressure on the chair and the speed of the mouse. In short, the recognition of frustration from text has not been adequately studied. A recent paper (Hu et al., 2018) is a relatively rare example in which intensities for eight differing emotional "tones" (anxious, frustrated, impolite, passionate, polite, sad, satisfied, and empathetic) were annotated; however, the goals and methods of this work and ours were significantly different: while we examine the effect of words and non-lexical means of expression on perceived frustration, they look for correlations between user and support worker emotional tones; also, their approach uses a seq2seq ("sequence to sequence") neural model, while we employ a model based on an architecturally simpler, fully-connected feedforward neural classification network.

The works we have mentioned have one feature in common: the classification of emotions in them is based on the analysis of words, or lexical means of expression. Several studies have been devoted to compiling lexicons, such as (Staiano and Guerini, 2014) and (Strapparava and Valitutti, 2004), but we have not been able to find a lexicon of non-lexical features. (Aman and Szpakowicz, 2007) mentions the use of non-lexical means of expression for the classification of emotions (using Eckman's scheme) but does not give a list or description of these means. With the growing popularity of Twitter as a source of data, several works are making use of emojis, like (Wood and Ruder, 2016) or hashtags, like (Das and Bandyopadhyay, 2010). The article (Mohammad and Kiritchenko, 2015) is devoted to the construction of a hashtag lexicon for the automatic classification of emotions, however, there is one problem with their use: most Twitter messages do not contain any hashtags, at least in the domain of customer support conversations. Textual features such as the use of exclamation and question marks were used in (Hasan et al., 2014), and message length in (Roberts et al., 2012). The use of non-lexical (and non-linguistic) means of expression for the classification and intensity of emotions is much more developed for voice communication. An example is (Hautasaari, 2019), where both non-lexical features such as speaking speed or number and length of pauses, and non-linguistic features such as inhalation and exhalation are used to classify emotions into eight classes.

Attempts to use other linguistic features, such as the ratio of sentence parts to each other, like in (Devillers and Vidrascu, 2006) or the number of word separators or word separator sequences as in (Perikos and Hatzilygeroudis, 2016) in addition to a basic bag-of-words representation, are being made, but a systematic review and study of such features is still lacking.

3 Preceding Work

We use two gauge points for appraising the performance of our model: first is the baseline accuracy, obtained by always predicting the most frequent ground-truth rating, and second — the results obtained by an equivalent model based solely on lexical features and not employing any sort of input processing. A paper by (Zuters and Leonova, 2020) presented such a model for predicting frustration intensity level based on lexical features only. There, the authors employed a fully connected feedforward neural network with 64 hidden units. This network took as input a bag-of-words representation of the input text, using a subset vocabulary constructed during the training phase. In order to construct this, for every word in the dataset that was encountered in more than 2 entries, the following statistics were calculated: the average value of frustration intensity of the entries this word was found in, and the standard deviation of this value. Entries that were annotated as not rated (“n”) or missing a rating value were ignored. The standard deviation is a main criterion for constructing the bag-of-words “best words” vocabulary, based on reasoning that the lower the standard deviation of the frustration rating, the more characteristic the specific word is for the given frustration intensity.

Entry	**No. of occur.**	**Avg. value**	**St. dev.**
offer	7	2.5714	0.7284
offered	3	3.3333	0.4714
offering	3	3.3333	0.9428

Table 1: Statistical metrics of different forms of the word “offer”.

Table 1 provides an excerpt from such a vocabulary: for each word, the following numbers are provided: the number of occurrences of the word in the dataset, the average value and the standard deviation of this value.

For each set of training data, a vocabulary was constructed, and then used for preprocessing each input entry for calculating predictions. The output of the model was a value from 0 to 4, representing the predicted frustration intensity. The performance of this model is used as a baseline for comparison.

4 Model

Here, we propose a new model that uses as input a set of features based on non-lexical means of expression in addition to the basic bag-of-words representation, and also employs subword segmentation for input preprocessing. By adopting

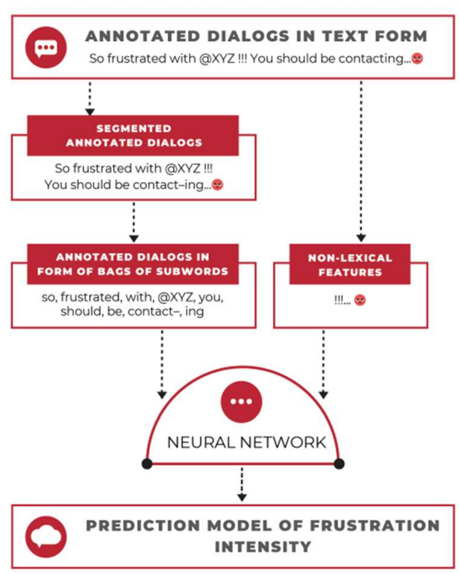

Figure 1: Model schema.

these techniques, we demonstrate a significant improvement in prediction accuracy, as discussed in detail in the Results section. A schematic of the model is given in Figure 1.

It can be seen that a user message in the model is used to construct two types of features: lexicon-based and non-lexical means of expression-based. We use the same method of lexicon construction as developed for the baseline model which was described in the Previous Work section. However, the new model differs in that it applies subword segmentation of the user message prior to constructing the bag-of-words representation (which thus becomes a 'bag-of-subword-units'), and it also adds features to the model input based on a range of non-lexical means of expression. We next discuss these in detail.

4.1 Non-Lexical Means of Expression

While there is no shortage of research based on annotating emotions on the basis of lexical means of expression, and considerable effort has been dedicated to developing lexicons and word embeddings to improve the results, non-lexical means of expressions are used but very sparingly. It is true that in text-based social media — as opposed to, for example, in personal conversations— non-verbal signs of emotion such as intonation or facial expressions are naturally absent, and thus mostly lexical means are used for emotion identification from text. However, people creatively use the means available to compensate, at least partially, for the absence of such non-verbal means. To this end, built-in and homemade emoticons, or "smileys", sometimes composed of typographic marks, as well as typographic marks themselves (e.g. quotes for sarcasm), and also hashtags and the like, are very commonly used.

(Mohammad and Bravo-Marques, 2017) showed that hashtags consistently increase the perceived intensity of emotions in Twitter messages in English, which suggests that making use of these and other non-lexical means of expression to improve automatic annotation has promise. We also would like to mention that not all emojis are used equally by all users. Some appear situationally and play an illustrative role by commenting the text in the form of an image. However, emoticons are not the only means of expression used to express emotions in a text. Traditional forms are also used, such as punctuation marks of all kinds, and conversational features such as two-, three- or more -fold repetition of letters, as well as more Internet-specific ones such as uppercase writing, among others.

4.2 Feature Selection

In the dataset we have constructed, we have identified a number of non-lexical means of expression (NLME), for each of which we calculated the correlation with the median human-annotated level of frustration. The correlation served as a selection criterion for selecting NLME for further feature construction. The original correlation table can be found in the accompanying GitHub repository, along with the other source files. Contrary to our expectations, we found that means of expressions such as hashtags do not possess predictive value for the level of frustration.

The same is true for emoticons expressing seemingly positive feelings, such as smiling, laughing faces and similar. Having looked more closely at the examples containing such emoticons, we concluded that the most likely reason for the absence of such correlation is that these are used to denote sarcasm as often as not. An interesting fact is that, while the tendency is preserved for self-made emoticons constructed from typographic marks, such as "(-:", it is less pronounced. The final list of selected features looks as follows:

- Message length
- Number of exclamation marks
- Number of exclamation marks normalized in relation to the message length
- Number of question marks
- Number of dots
- Number of commas
- Number of quotation marks (single, double, and reversed quotes)
- Number of uppercase words of length 5 and more
- Number of repeated letter "a" sequences
- Number of Twitter built-in emojis

- Number of positive emojis, constructed from typographic marks
- Number of negative emojis, constructed from typographic marks
- Presence of a picture in the message
- PTAC mention in the message (PTAC stands for Consumer Protection Service in Latvian)

After having selected the promising features, we tested different combinations of those and found the following. Firstly, there is no single feature dominantly responsible for the improved performance. The best feature, namely, the number of exclamation marks, gave only 46.8% accuracy. Secondly, exclusion of the worst features, whose inclusion individually gives worse results than the model without any NLME features at all (quotes, repeating letters, negative smileys made of typographic marks, presence of a picture in the message), also decreases performance by about 0.5%. This means that all the listed features are necessary to achieve the maximal performance.

4.3 Segmentation

Another technique that we employed in order to improve performance is preprocessing of the input data with a subword segmentation tool. The reasoning behind this is that Latvian is closer to being a synthetic language than an analytical one, and thus each word is present in the dataset in a multitude of forms — differing for every combination of case, number, gender, and other grammatical categories.

Entry	No. of occur.	Avg. value	St. dev.
lg	8	2.0	0.0
publisk	5	3.0	0.0
neiet	4	2.0	0.0
mac	4	3.0	0.0
piezvan	4	3.0	0.0
neierobežots	5	2.0	0.0
|isku	4	3.0	0.0
izmēģin	4	3.0	0.0
nov.	4	3.0	0.0
pagāj	3	3.0	0.0

Table 2: Ten best segmented entries. The vertical line | indicates a subword unit that gets joined to whatever precedes it.

To alleviate this effect, segmentation has been successfully employed in machine translation field. Table 2 shows the top ten entries from the word dictionary, illustrating the principle.

As the vocabulary for subsequent frustration level annotation is constructed automatically based on the distribution of the ratings for specific words, segmentation should facilitate classifying the same words in different grammatical forms together, potentially improving prediction accuracy. It also allows to unify a number of forms under a single entry, reducing data sparsity. As we analyzed these entries in comparison with the whole-word dictionary, we saw, that: 1) brand names ("lg", "mac") and unchanged ("nov.", "neierobežots") words preserved their place on top; one entry ("-isku") is a word ending that couldn't have been in an unsegmented dictionary; two ("publisk-", "pagāj-") are new developments — they originally appeared as whole words less than three times and were thus previously ignored, but are now included by virtue of serving as a root form for multiple related word-forms; and, finally, the remaining three ("izmēģin-", "piezvan-" and "neiet") were reranked, for similar reasons. The improvements in performance due to subword segmentation are discussed in the Results section.

4.4 Additional Processing

In addition to subword segmentation, we also explored other input preparation methods. Specifically, removing diacritical marks in original entries in order to unify spelling variations differing only in their presence or absence, and replacing abbreviations for time, speed and other units of measure, as well as popular sources of spelling variations, with their full forms. The effect of these two methods, even applied cumulatively, was found to be disputable at best and provided, no real improvement of the model accuracy.

5 Dataset

In this work, we have used a completely new Latvian dataset, developed specifically for the purpose of testing the performance of our proposed frustration annotation model against the old one used as a gauge point. Following the example of many other recent researchers, we selected Twitter as a data source of choice. Four major Latvian internet and telecommunication service provider accounts were chosen for collecting conversations of users with customer support representatives. Those accounts are: (@mans_tet), (@mans_LMT), (@Bitelv) and (@tele2Latvija), which belong to companies Tet, LMT, Bite and Tele2, respectively. To provide for not just the possibility of appraising a frustration level at a certain moment of time, but also to allow studying the dynamics of frustration changes from one user turn to the next, the selected conversations contain no less than two user turns, with at least one customer support turn between those. As a turn we consider a sequence of messages that belong to one party in the conversation (not interrupted by any other users' messages). An essential criterion was that the dialogs should be in Latvian, as obtaining a dataset in Latvian was the primary goal of collecting this dataset of tweets. The conversations were collected manually, in order to ensure that the criteria are met and that eligible conversations were not excluded just because another user replied to a tweet from the conversation, if such intervention did not actually affect the initial dialog. The resulting dataset consists of 283 dialogs with 688 user turns and 531 customer support representative turns. Of those 688 user turns, 9% had a median frustration value of 0, 19% of 1, 31% of 2, 30% of 3 and, finally, 11% of 4.

The resulting collection was post-processed and saved in a unified and anonymized format, as described in the Experimental Setup section.

Each user's turn in the dataset is followed by three values, representing the level of frustration assigned to this turn by three independent annotators. Each value represents a frustration level measured on a scale of 0 to 4, or can be "n" if the annotator judged that a level of frustration could not be determined from the text of the user message, for example, in case of the user simply stating their address. A final option is a missing value, for example if the text was in a language other than Latvian or could not be understood by an annotator for some other reason.

For English, in order for the results to be comparable with the baseline, we have employed the same dataset that was used in (Zuters and Leonova, 2020). The dataset represents a small subset of the Kaggle Customer Support dataset, where approximately 400 consistent dialogues between a support and a user were isolated and annotated for frustration by three independent annotators on the scale of 0..4. In total, this dataset contained 843 user turns, of which 18% had median frustration value of 0, 15% of 1, 28% of 2, 27% of 3 and 11% of 4. Thus, the baseline accuracy of the most frequent value was 31% for Latvian and 28% for English, and the most frequent frustration values were 2 and 3, respectively.

6 Experimental Setup

The model was implemented on Python as a neural network with RELU activation function and a softmax layer. It is using the following metaparameters: number of hidden units, number of epochs. The model, using both non-lexical and lexical means of expression, whether it was using subword segmentation of the input text or not, was trained on the same set of data.

We tested the model on different vocabulary sizes, fixing the number of epochs as 100 and the number of hidden units as 64, and found that the tendency of vocabulary sizes 50 and 500 to underperform has been preserved. We further explored the effect of hyperparameter changes, and also some additional input processing methods. We tried different network sizes (number of hidden units in the dense feedforward layer), including 32, 128, and 256, and concluded that 32 hidden units were not sufficient, while 128 and 256 gave suboptimal results with 0.75%, 0.5% and 1.5% decline in accuracy, respectively.

The model takes as its input a file with a collection of dialogs between users and customer support representatives in an anonymized format. Information such as Twitter user ids and message ids were removed from the dialogs, and any included sensitive information such as e-mail or customer number, is replaced with generic placeholders, following the example of the Kaggle Customer Support dataset. All user messages in the dataset are tagged with either "USER:" or "SUPP:" to denote whether they belong to a client/user or to the company's customer support representative, respectively. Consecutive messages coming from a single party are joined together, forming a single "turn" — a sequence of messages uninterrupted by another party, so that the model works under the assumption that each two consecutive messages in a dialog belong to the different parties. For all experiments reported here, we use a single value per turn, calculated as a median of three ratings given by the annotators, which allows keeping the aggregated annotation value as an integer, thus enabling us to perform classification rather than regression. The number of occurrences of each of the median ratings for values 0 to 4 is [59, 131, 210, 205, 83], respectively, with 2 being the most frequent rating with 210 instances.

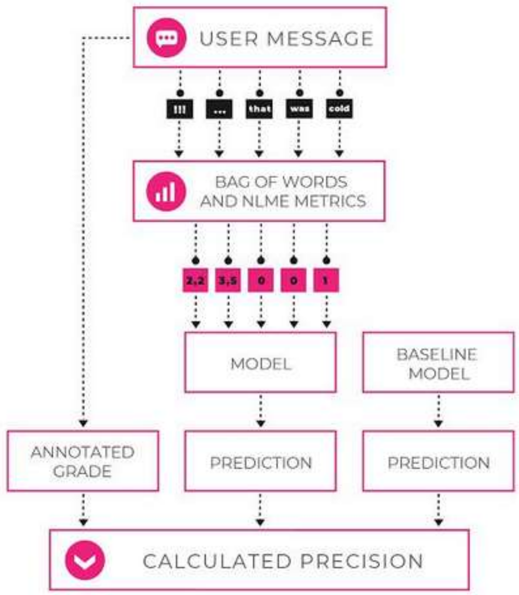

Figure 2: Frustration intensity prediction.

To determine the optimal selection of features, used in the model, we have used a two-step process. To appraise the performance of the NLME features, for comparison we used the model with bag-of-words input features only. Figure 2 provides its principal schema of operation.

In the first step, we identified all potential features in the dataset and calculated a correlation table with the median annotated value, leaving only those that had at least a weak correlation. In the second step, we have run the model with a different combination of those features. First, we have used none and all the features for a benchmark, and then tested every feature in isolation to verify that no single feature would give a comparable accuracy. Since none did, we next tried to exclude features that in isolation gave worse results than a model using bag-of-words only. However, the removal of underperforming features (those which in isolation give results below the performance of the bag-of-words-only model) resulted in decreasing the overall performance by 0.4% (z-score = 1.32).

For segmentation of Latvian text, we have applied a GenSeg tool, described in (Zuters and Strazds, 2019) to preprocess the dialog file, so that the input now consisted of the messages in an already segmented form, leaving the rest of the process exactly as before — so that the run of the model on segmented versus unsegmented data differed only in the input file. Having fixed the metaparameters at 64 hidden units and vocabulary size 100, we found that subword segmentation improved the resulting model accuracy by 1.25% (z-score = -4.02).

7 Results

As our goal was to test improvements on Latvian data, and additionally adapt the original model to the specific challenges of the Latvian language, we created a reference point for comparison by training the baseline neural model using our Latvian language dataset. As discussed in the Preceding Work section, we defined a first baseline for prediction as the accuracy achieved by always assigning the most frequent annotation value in the corpus. However, different from the English dataset, the most frequent value in the Latvian dataset is 2 (with a distribution of values [59, 131, 210, 205, 83]), and the corresponding baseline accuracy is 30.5% The baseline neural model uses 64 hidden units, vocabulary size 100 and a bag-of-words input representation.

	C1	C2	C3	C4	RM	BM
Accura-cy	48.8	48.4	47.5	46.9	42.2	30.5

Table 3: Prediction results for the NLME model in comparison with different configurations (for Latvian). C1 - NLME model with all features, C2 - NLME model without subpar features, C3 - NLME model with all features and no segmentation, C4 - NLME model with a single best feature, RM – reference model, BM – baseline model.

Table 3 summarizes the results for the most prominent configurations of the features. It can be seen that using the combination of the best features increases the accuracy by 1.5% compared to using the single best feature, while adding the underperforming features improves the result by another 0.4%.

Using input preprocessed with the GenSeg segmentation tool gives the highest performance, yielding a total 49% accuracy, which is an 8% improvement over the old model, of which 1.25% can be attributed to the subword segmentation.

Diacritics removal and unification of abbreviations did not have any discernible effect on the model performance. The best results are achieved using 64 hidden units and vocabulary size 100.
Summarizing the results of our experiments, we can conclude that using our proposed model for prediction of frustration intensity significantly increases the accuracy of predictions.

8 Conclusions

In this paper, we have proposed a new neural network-based model for frustration intensity prediction for customer messages in the context of conversations with customer support representatives, as well as a new dataset of such conversations in Latvian, used to train and evaluate the model performance. A baseline model used for comparison employs a bag-of-words representation as input, constructed on the basis of vocabulary that is dynamically built during the training phase. The words selected for inclusion in the vocabulary are the ones that have the least standard deviation for annotated frustration intensity values. Our proposed method differs, first and most importantly, by including also features constructed on the basis of non-lexical means of expression, and, secondly, by performing close-to-morphological segmentation as a preprocessing step to make vocabulary construction more coherent. We also examined a few other methods of input processing, namely, removal of diacritics, and unifying popular variations in spelling, but these did not yield any improvements in results, even when used together.

Performance was assessed by performing a leave-one-out cross exhaustive cross-validation, that is, by computing accuracy (as percentage of correct predictions) obtained after training the model on all data except one entry using this one entry for prediction, and repeating this process for all the entries in turn, averaged across five runs. We compare against a similar neural model that only uses lexical features as input. While such a model achieves a 10% improvement over the baseline accuracy obtained by always predicting the median frustration rating of the dataset, our proposed model achieves an improvement in accuracy of 18% over the baseline and 8% over the old model. We have conducted ablation studies to evaluate the contribution attributable to input preprocessing using close-to morphological segmentation, and also adjusted hyperparameters, and found that the segmentation is responsible for approximately 1.25% of the improvement.

In addition, we have presented a new dataset in Latvian, that contains dialogs between users and customer support specialists. The dataset in total has 283 dialogs with 688 user turns and 531 customer support representative turns, with each dialog containing no less than two user turns separated by a support representative turn, and with all user turns manually annotated for frustration intensity level. This dataset was used for training and assessing the reference neural network-based model for frustration prediction and its improved version.

Acknowledgments

The research has been supported by the European Regional Development Fund within the joint project of SIA TILDE and University of Latvia "Multilingual Artificial Intelligence Based Human Computer Interaction" No.1.1.1.1/18/A/148.

References

Verma, Jai Prakash, Smita Agrawal, Bankim Patel, and Atul Patel. "Big data analytics: Challenges and applications for text, audio, video, and social media data." *International Journal on Soft Computing, Artificial Intelligence and Applications (IJSCAI)* 5, no. 1 (2016): 41-51.

Khalifa, Nooriddin S. "Internet Chat as 'Jargon'." Journal of Language Studies 3, no. 1 (2020): 149-175.

Giacinto, Giorgio, and Fabio Roli. "Design of effective neural network ensembles for image classification purposes." *Image and Vision Computing* 19, no. 9-10 (2001): 699-707.

Nicholson, Joy, Kazuhiko Takahashi, and Ryohei Nakatsu. "Emotion recognition in speech using neural networks." *Neural computing & applications* 9, no. 4 (2000): 290-296.

Alm, Cecilia Ovesdotter, Dan Roth, and Richard Sproat. "Emotions from text: machine learning for text-based emotion prediction." In *Proceedings of human language technology conference and conference on empirical methods in natural language processing*, pp. 579-586. 2005.

Ekman, Paul. "An argument for basic emotions." *Cognition & emotion* 6, no. 3-4 (1992): 169-200.

Yao, Yuanlin, Shuwei Wang, Ruifeng Xu, Bin Liu, Lin Gui, Qin Lu, and Xiaolong Wang. "The construction of an emotion annotated corpus on microblog text."

(In Chinese) *Journal of Chinese Information Processing* 28, no. 5 (2014): 83-91.

Lee, Sophia, and Zhongqing Wang. "Emotion in code-switching texts: Corpus construction and analysis." In *Proceedings of the Eighth SIGHAN workshop on Chinese language processing*, pp. 91-99. 2015.

Russell, James A., and Albert Mehrabian. "Evidence for a three-factor theory of emotions." *Journal of research in Personality* 11, no. 3 (1977): 273-294.

Parthasarathy, Srinivas, and Carlos Busso. "Jointly Predicting Arousal, Valence and Dominance with Multi-Task Learning." In *Interspeech*, vol. 2017, pp. 1103-1107. 2017.

Yu, Liang-Chih, Lung-Hao Lee, Shuai Hao, Jin Wang, Yunchao He, Jun Hu, K. Robert Lai, and Xuejie Zhang. "Building Chinese affective resources in valence-arousal dimensions." In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pp. 540-545. 2016.

Bravo-Marquez, Felipe, Eibe Frank, Bernhard Pfahringer, and Saif M. Mohammad. "AffectiveTweets: A Weka package for analyzing affect in tweets." (2019): 1-6.

Duppada, Venkatesh, and Sushant Hiray. "Seernet at emoint-2017: Tweet emotion intensity estimator." *arXiv preprint arXiv:1708.06185* (2017).

Stauss, Bernd, Maxie Schmidt, and Andreas Schoeler. "Customer frustration in loyalty programs." *International Journal of Service Industry Management* (2005).

Klein, Jonathan, Youngme Moon, and Rosalind W. Picard. "This computer responds to user frustration: Theory, design, and results." *Interacting with computers* 14, no. 2 (2002): 119-140.

Hone, Kate. "Empathic agents to reduce user frustration: The effects of varying agent characteristics." *Interacting with computers* 18, no. 2 (2006): 227-245.

Kapoor, Ashish, Winslow Burleson, and Rosalind W. Picard. "Automatic prediction of frustration." *International journal of human-computer studies* 65, no. 8 (2007): 724-736.

Hu, Tianran, Anbang Xu, Zhe Liu, Quanzeng You, Yufan Guo, Vibha Sinha, Jiebo Luo, and Rama Akkiraju. "Touch your heart: A tone-aware chatbot for customer care on social media." In *Proceedings of the 2018 CHI conference on human factors in computing systems*, pp. 1-12. 2018.

Staiano, Jacopo, and Marco Guerini. "Depechemood: a lexicon for emotion analysis from crowd-annotated news." *arXiv preprint arXiv:1405.1605* (2014).

Strapparava, Carlo, and Alessandro Valitutti. "Wordnet affect: an affective extension of wordnet." In *Lrec*, vol. 4, no. 1083-1086, p. 40. 2004.

Aman, Saima, and Stan Szpakowicz. "Identifying expressions of emotion in text." In *International Conference on Text, Speech and Dialogue*, pp. 196-205. Springer, Berlin, Heidelberg, 2007.

Wood, Ian, and Sebastian Ruder. "Emoji as emotion tags for tweets." In *Proceedings of the Emotion and Sentiment Analysis Workshop LREC2016, Portorož, Slovenia*, pp. 76-79. 2016.

Das, Dipankar, and Sivaji Bandyopadhyay. "Labeling emotion in Bengali blog corpus–a fine grained tagging at sentence level." In *Proceedings of the Eighth Workshop on Asian Language Resources*, pp. 47-55. 2010.

Mohammad, Saif M., and Svetlana Kiritchenko. "Using hashtags to capture fine emotion categories from tweets." *Computational Intelligence* 31, no. 2 (2015): 301-326.

Hasan, Maryam, Emmanuel Agu, and Elke Rundensteiner. "Using hashtags as labels for supervised learning of emotions in twitter messages." In *ACM SIGKDD workshop on health informatics, New York*, USA. 2014.

Roberts, Kirk, Michael A. Roach, Joseph Johnson, Josh Guthrie, and Sanda M. Harabagiu. "EmpaTweet: Annotating and Detecting Emotions on Twitter." In *Lrec*, vol. 12, pp. 3806-3813. 2012.

Hautasaari, Ari, Naomi Yamashita, and Ge Gao. "How non-native english speakers perceive the emotional valence of messages in text-based computer-mediated communication." *Discourse Processes* 56, no. 1 (2019): 24-40.

Devillers, Laurence, and Laurence Vidrascu. "Real-life emotions detection with lexical and paralinguistic cues on human-human call center dialogs." In *Ninth International Conference on Spoken Language Processing*. 2006.

Perikos, Isidoros, and Ioannis Hatzilygeroudis. "Recognizing emotions in text using ensemble of classifiers." *Engineering Applications of Artificial Intelligence* 51 (2016): 191-201.

Jung, Younghee, Kinam Park, Taemin Lee, Jeongmin Chae, and Soonyoung Jung. "A corpus-based approach to classifying emotions using Korean linguistic features." *Cluster Computing* 20, no. 1 (2017): 583-595.

Zuters, Janis, and Viktorija Leonova. "Adaptive Vocabulary Construction for Frustration Intensity Modelling in Customer Support Dialog Texts." International Journal of Computer Science & Information Technology (IJCSIT) Vol 12 (2020).

Mohammad, Saif M., and Felipe Bravo-Marquez. "Emotion intensities in tweets." *arXiv preprint arXiv:1708.03696* (2017).

https://twitter.com/mans_tet, accessed 21.08.2020.

https://twitter.com/mansLMT, accessed 21.08.2020.

https://twitter.com/Bitelv, accessed 21.08.2020.

https://twitter.com/Tele2Latvia, accessed 21.08.2020.

Zuters, Jānis, and Gus Strazds. "Subword segmentation for machine translation based on grouping words by potential roots." *Baltic Journal of Modern Computing* 7, no. 4 (2019): 500-509...