Proactive Detractor Detection Framework Based on Message-Wise Sentiment Analysis Over Customer Support Interactions

J. S. Salcedo-Gallo[1], J. Solano[1], H. García[1], D. Zarruk-Valencia[1, 2], and A. Correa-Bahnsen[1]

[1]Rappi AI Research
[1]{sebastian.salcedo, jesus.solano, javier.garcia, alejandro.correa}@rappi.com
[2]davidzarruk@gmail.com

Abstract

In this work, we propose a framework relying solely on chat-based customer support (CS) interactions for predicting the recommendation decision of individual users. For our case study, we analyzed a total number of 16.4k users and 48.7k customer support conversations within the financial vertical of a large e-commerce company in Latin America. Consequently, our main contributions and objectives are to use Natural Language Processing (NLP) to assess and predict the recommendation behavior where, in addition to using static sentiment analysis, we exploit the predictive power of each user's sentiment dynamics. Our results show that, with respective feature interpretability, it is possible to predict the likelihood of a user to recommend a product or service, based solely on the message-wise sentiment evolution of their CS conversations in a fully automated way.

1 Introduction

Since the introduction of the Net Promoter Score (NPS) (1) in the early 2000s, it has become an important metric for measuring customer recommendation behavior across multiple industries (2; 3; 4; 5; 6; 7; 8; 9; 10; 11; 12; 13; 14; 15). This score is based on a rather straightforward question, namely *'How likely would you be to recommend us to your friends or family?'* Moreover, these surveys are massively conducted at a frequency ranging from monthly to an annual basis (11), or even immediately after individual interaction with Customer Support (CS) (6). In practice, the NPS score can be segmented into three main groups, namely *promoters, passives, and detractors* (1).

Previous studies on sentiment analysis related to NPS recommendation behavior have not been studied in chat-based customer relationship interactions, which have gained popularity given the massive outreach of widely adopted messaging applications and Client Resource Management (CRM) platforms. Furthermore, it is of compelling interest for companies to predict recommendation decisions to proactively and effectively correct service failures, and retribute customers with incentives, which ultimately could improve the company's ratings. Therefore, our main contribution and objective are to use NLP to assess and predict the recommendation behavior of the users where, in addition to using static sentiment analysis, we exploit each user's sentiment evolution throughout a particular conversation, inspired by the inherently dynamic nature of human conversations (16), while employing widely-known and robust ML transformed-based architectures (17).

In this work, we address the NPS binary classification problem, namely predicting whether a given user would be a *promoter* or not. We have a total number of 16.4k users and 48.7k CS conversations with a large e-commerce company in Latin-America for one specific city, where each particular recommendation rating is obtained directly from the users themselves in corresponding survey responses.

36th Conference on Neural Information Processing Systems (NeurIPS 2022).

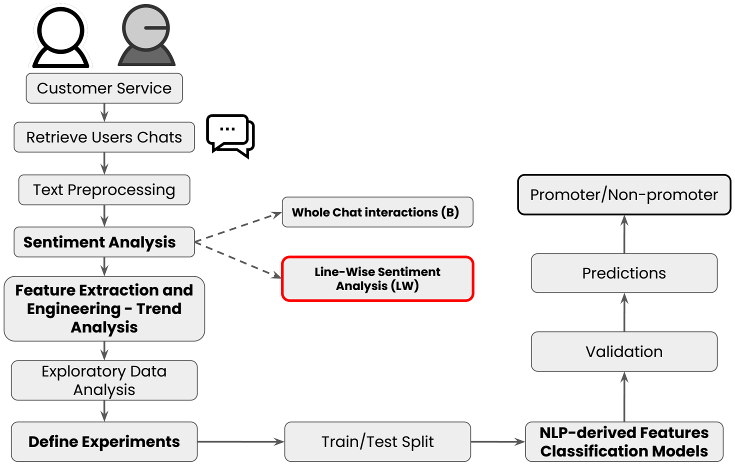

Customer Service
Retrieve Users Chats
Text Preprocessing
Sentiment Analysis
Whole Chat interactions (B)
Line-Wise Sentiment Analysis (LW)
Feature Extraction and Engineering - Trend Analysis
Exploratory Data Analysis
Define Experiments
Train/Test Split
NLP-derived Features Classification Models
Validation
Predictions
Promoter/Non-promoter

Figure 1: Our proposed framework for proactive *promoter/non-promoter* user detection for any chat-based user interaction.

With this, we have performed sentiment classification based on transformer-based architecture over each user's CS chat-based interactions to predict well in advance the recommendation behavior of particular users. Each turn-level inherent sentiment evolution feature is used to perform a classification task at hand. We draw inspiration from existing approaches for conversational discourse-aware response generation such as DialogBERT (18), DialoGPT (19), or Blender (20) for our analysis.

Furthermore, we propose a message-wise sentiment evolution analysis of the customer-sent messages throughout the conversation. This message-wise approach consists of obtaining a sentiment score for each message the user sends in a particular chat-based CS conversation. Thus, allowing us to analyze the trend and overall sentiment evolution and their relation with the recommendation behavior of any particular user. Our results show that overall, message-wise evolution analysis is substantially superior in predicting the recommendation behavior of individual users compared to a traditional review-based approach of computing the overall sentiment of the complete interactions. To the best of our knowledge, this is the first message-wise sentiment evolution analysis over chat-based CS interactions to predict the recommendation behavior of users, which is highly extensible for any particular domain and any specific business-driven metric. For instance, N-class or binary problems, such as the Satisfaction Score (CSAT), Churn score, Recency-Frequency-Monetary (RFM) score, or even fraud detection, to name a few.

2 Background

The most relevant advances in algorithms to predict the NPS score are based on online reviews (5), word of mouth (WOM) (2), social media data analytics (21), explicit quantitative experience attributes and service ratings (11; 14), among others(13). These previous approaches used simple regression models (22), tree-based classification approaches such as simple Decision Trees (11), as well as Support Vector Machines (11), Deep Neural Networks (6), and probabilistic approaches (14). Moreover, for obtaining quantitative insights from text, there is evidence of manually and empirically performed analysis based on explicit human annotations to assess the negative, neutral, and positive polarity of eWOM messages, and their correlation with *detractors, passives, and promoters* recommendation decisions (2). Also, other classification approaches for *'Recommend'* or *'No-recommend'* behavior have considered more robust modeling such as bag-of-words and aspect-based sentiment analysis over online airline reviews (7; 5). However, on one hand, the actual predictive powers of combining qualitative and quantitative user-generated content on the recommendation behavior were not explored and are limited to correlation and statistical analysis so far. On the other hand, in the case of online reviews, where authors (5) used overall-sentiment, aspect-specific-sentiment, and bag-of-words, they indeed address the performance on the prediction task of recommending/non-recommending. However, their results and approach are not generally comparable with ours, given the fundamental difference between chat-based interactions and online reviews (5; 11; 6; 4; 12). Consequently, the actual predictive powers of inherent characteristics obtained from the sentiment evolution in chat-based interactions were yet to be explored and assessed in general, and in a real-world environment.

An earlier study (7) showed that positive emotions have a positive relationship with customer outcomes, whereas negative emotions had a negative relation with recommendation behavior regardless of the nature of the service. As such, the author made relevant contributions from a theoretical point of view. However, the actual predictive powers on a classification task were not addressed nor explored.

Finally, authors in (13) stated that, for future work, ensembles and optimization procedures should be applied for getting predictive recommendation decisions in the scope of the NPS prediction problem. As such, we here explore the use of an ML Ensemble model, more specifically Gradient Boosting Trees. All of this, in the light of a real-life dataset, while studying the impact of using a random under-sampling technique for training in order to improve generalization.

3 Detractor Detection Framework Based on Sentiment Analysis

In this work, we propose a framework for proactive detection of the recommendation behavior in the scope of the NPS classification problem considering the user's chat-based CS interactions, as shown in Figure 1. The main idea lies in the hypothesis that the trend and sentiment evolution of chat-based conversations can better capture the overall behavior of the customers toward their recommendation decisions. Our method consists on classifying the sentiment of each message with language modeling based on ML transformers architecture. For instance, let us consider a user that started a conversation rather neutral, then swung to rather negative consecutive messages while the issue is being exposed, to then end the conversation rather positively as the issue might have been effectively solved. In this case, the user is presumably more likely to recommend the service as a result of good customer experience or low frustration.

3.1 Sentiment analysis of CS conversations

Specifically, for the sentiment measure $SS(\cdot)$ we use a sentiment classifier based on the Bidirectional Encoder Representations from Transformers (BERT) architecture (17), which is a transformers-based ML technique for NLP applications such as sentiment classification (23), question answering (24), masked language models(25), Next Sentence prediction tasks (26), among others (27). In this case, we used a specific pre-trained model that is fine-tuned and intended for direct use as a sentiment classification model for product reviews in English, Dutch, German, French, Spanish, or Italian. Regardless of the language, this model has a sufficiently robust off-by-one accuracy ranging from 93% to 95%, corresponding to the percentage of reviews where the number of stars the model predicts differs by a maximum of 1 from the number given by the human reviewer in a 0-4 scale.

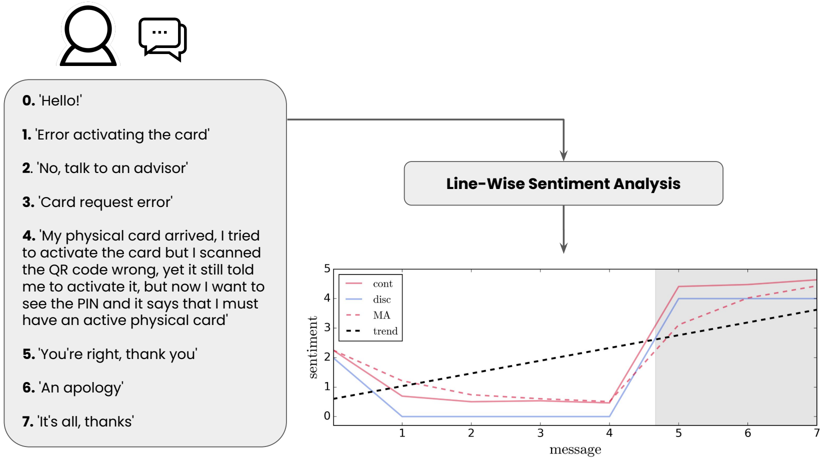
Once we obtain the message-wise sentiment per user's responses in each specific longest conversation, we perform manual feature extraction and feature engineering based on the retrieved sentiment classification. For the conversation $\hat{c} \in C_i$ we construct the **discrete sentiment line** as the curve associated to the vector $disc = MWS(\hat{c})$. More precisely, for a given message m the model return the number of stars associated to m as the value $SS(m)$. This model also returns the probability $\mathbb{P}(SS(m))$. Given that this 'discrete' approach ($MWS(\hat{c})$) might result in a very step-like behavior in general, we therefore opted for having a 'continuous' sentiment, namely the actual star having the largest probability score plus the actual score associated with that particular star. Formally, we define the **continuous sentiment curve** as the curve associated to the vector:

$$cont(\hat{c}) := (SS(\hat{m}_1) + \mathbb{P}(SS(\hat{m}_1)), .., SS(\hat{m}_N) + \mathbb{P}(SS(\hat{m}_N))) \tag{1}$$

where $\hat{c} := (\hat{m}_1, ..., \hat{m}_N)$.

To further smooth the sentiment message-wise series, we apply an exponential weighted mean function to Eq. 1, so that we obtain a more flexible evolution onset, more suitable for trend and concavity analysis:

$$MA_j := \alpha \cdot cont(\hat{c})_j + (1 - \alpha) \cdot MA_{j-1} \tag{2}$$



0. 'Hello!'

1. 'Error activating the card'

2. 'No, talk to an advisor'

3. 'Card request error'

4. 'My physical card arrived, I tried to activate the card but I scanned the QR code wrong, yet it still told me to activate it, but now I want to see the PIN and it says that I must have an active physical card'

5. 'You're right, thank you'

6. 'An apology'

7. 'It's all, thanks'

Figure 2: We depict a typical example of user-sent messages to CS in a particular interaction. The blue solid line represent the 'discrete' sentiment, which corresponds to the star rating computed with the sentiment classifier per each user's message. The red solid line represent the 'continuous' sentiment, which corresponds to the discrete star rating plus the actual score associated with that particular star. The red dashed line represents an exponential weighted mean function (MA), and the black dashed line represent a linear fit over the smoothed MA curve. The shaded region corresponds to the last third of the conversation, which we have used to extract some features for our analysis. We only plot the linear fit to the conversation, even though concavity and descriptive statistics features are also computed from this message-wise analysis.

which we depict as a red dashed line in Fig. 2, where we consider an static decay parameter of $\alpha = 2/3$. We apply a simple linear regression to capture the trend, which in general can give an approximation for overall evolution of the conversation. Then this value slope is used as a feature for our classification model.

Finally, we define the experiments, namely the sampling technique for unbalanced treatment, target labels, and the classification model to train. For this, we use a widely-known ML classification algorithm, namely Gradient Boosting Trees (XGBoost) (28) and Random Search hyper-parameter tuning for the classifier (29). The proposed framework is very flexible as it could be easily extended to any chat-based interactions to predict any business-driven metric or N-class customer rating classification task, such as Satisfaction Score (CSAT), Churn score, Recency-Frequency-Monetary (RFM) score, or even fraud detection.

3.2 Baseline Model

In order to assess the relevance of the here proposed framework for chat-based CS interactions, we obtain a baseline model that mimics an empirical sentiment estimation for batch text of the complete interaction. First, for our baseline model, we consider the static sentiment of each complete interaction. Let C_i be the set of complete conversations of the user u_i. For each $c \in C_i$ we compute the static sentiment $SS(c)$ using the architecture described in section 3.1. Therefore, we compute the descriptive statistics variables such as the mean, minimum, maximum, and median sentiment of the vector $(SS(c))_{c \in C_i}$, as well as the number of CS interactions, $|C_i|$. Accordingly, with this baseline set-up features we can train a model, following procedure described in Section 4.2, that gives an estimation of the probability of a given user to be a *promoter* based on overall static sentiment evaluation. However, it is very relevant to make clear that our main contribution is not just to use sentiment analysis and Natural Language Processing (NLP) to predict recommendation behavior, but rather to use highly mature components (17) in a novel way for capturing the actual sentiment evolution throughout the CS conversations, as presented in Sec. 3.1.

3.3 Line-wise Sentiment Analysis Model

We perform granular, flexible and general message-wise sentiment analysis to assess the inherent sentiment dynamic nature of conversations and how it relates to the recommendation behavior of a given user. To the best of our knowledge, there is no scientific evidence showing that the actual predictive powers of sentiment evolution in chat-based interactions have been explored and assessed regarding general binary classification tasks.

As such, we assess the predictive power of quantitative message-wise sentiment analysis features extracted from individual CS conversations. Consider a conversation $c \in C_i$ and its vector representation $c := (m_1, ..., m_N)$, where m_j is the j-th user message or turn appearing in the conversation, thus neglecting agent responses. We begin considering the vector of message-wise sentiment as $MWS(c) := (SS(m_1), ..., SS(m_N))$. Figure 2 shows an example of a typical CS conversation, where each message has an associated sentiment on the message-wise panel on the right. It can be seen that the conversation started rather neutral, then there were some issues exposed in the middle part of the conversation, which were effectively solved by the agent.

Let $\hat{c}$ be the longest conversation in C_i. Once the message-wise sentiment vector $MWS(\hat{c})$ is computed, we obtain descriptive statistics features, as well as concavity analysis related to the mean of the numerical second derivative and the overall trend fitted as a linear slope. With this, we are able to capture not only the overall trend, but also the evolution of the conversation in terms of concavity analysis. A more detailed description of this process appears in Section 3.1. We look at the evolution of the message-wise sentiment vector $MWS(C_i)$ over time. This includes both the number of messages belonging to each specific discrete sentiment class, as well as the descriptive statistics features extracted from the last third of the conversation. For the here presented framework, the overall sentiment of the complete interaction is compared against the granular sentiment evolution of the chat-based messages.

4 Evaluation

4.1 Dataset

Our dataset consists of the CS conversations and NPS surveys conducted in the second semester of 2021, of users within the financial vertical of a large e-commerce company for one specific city. In our case study we consider around 16.4k users having CS interactions whom also have filled the NPS survey, for a total number of interactions to the CS center $\bigcup C_i \approx$ 48.7k+. Also, the mean number of interactions correspond to 2.39. Therefore, we predominantly base our analysis on the largest conversation of a given user, while still considering general features for the overall CS interactions. Moreover, the mean number of messages for the largest conversations per user corresponds to 13.85 messages. We use only the customer responses, thus ignoring the agent's prompts. After querying the conversations, we then apply a simple preprocessing step, namely removing special characters and blank messages. As for the count of each recommendation behavior annotated by the users themselves in our dataset, we have a total of 10701 *promoter* users, whereas 5700 *non-promoter* ones, making it a slightly imbalanced classification problem. It is of compelling relevance to point out that we use only features extracted from text, thus preventing ourselves from using behavioral data or other specific service ratings.

4.2 Experimental Setup

We use Random Search for hyper-parameter tuning of the XGBoost classifier used in this work, where both the number of folds in a (Stratified)-KFold and the number of parameter settings that are sampled for each classifier were set to 10. For our experimental setup, we have a total of 16401 users, for which we use a user-wise train/test split ratio of 80% / 20% for the train and test set, respectively. We also explore the impact of sampling the train set over the predictive power in terms of validation metrics such as the F1-score, and Specificity (30). Moreover, we also assess the influence of *passive* (2470) users on the predictive power of the algorithm, so that we additionally obtain classification results considering only *promoters* (10701) and *detractors* (3230) users. In that regard, we consider three experiments in our setup, namely our (1) static-like sentiment baseline **(B)**, (2) message-wise sentiment evolution analysis considering *passive* users **(B + LW)** , and (3) message-wise sentiment evolution ignoring *passive* users **(B + LW{NP})**.

Table 1: XGBoost Classification results in terms of Area Under the Curve (AUC), Kolmogorov-Smirnov (KS) and Macro F1 score for our three experiments on the NPS binary classification task. As such, we have depicted the abbreviations for each experiment as: baseline **(B)**, our message-wise sentiment evolution analysis including *passive* users **(B + LW)**, and line-wise sentiment evolution ignoring *passive* users **(B+LW{NP})**.

	XGBoost		
Experiment	**AUC**	**KS**	**Macro F1**
B	0.5513	0.0801	0.54
B + LW	0.6199	0.1843	**0.58**
B + LW{NP}	**0.6455**	**0.2389**	**0.58**

We then validate the performance of the classification models for the different experiments using the Area Under the Curve (AUC) (31), Kolmogorov-Smirnov (KS) metric (32), and Macro F1 score (30) as our validation metrics for one-to-one comparison.

5 Results and Discussion

Table 1 shows the results of our classification task, namely predicting whether the given customer would be a *promoter* (1) or not (0). Results are presented by considering Random Undersampling only over our train set for dealing with imbalanced nature of our dataset. We show results for our baseline **(B)** approach, as well as for our dynamic line-wise sentiment analysis considering **(B + LW)** and ignoring **(B + LW{NP})** *passives* users.

First, the results presented in Table 1 for our baseline **(B)** show that this model is only marginally doing better than a random guesser. Therefore, the overall and static sentiment of complete CS interactions does not present strong predictive power considering only CS chat-based interactions. Nonetheless, they do have comparable performance if compared against to previous works where explicit service ratings were used as features, exhibiting a similar Macro F1 score of about 0.55 for a similar problem in the scope of the NPS classification task (33; 11). The observed low statistical accuracy of our baseline and previous approaches suggests that there should be additional attributes not captured in the sentiment of the whole interaction or surveys that explain the recommendation behavior of the users, which is the problem we are trying to tackle here.

The results for our **(B + LW)** experiment are significantly better than those exhibited by our baseline **(B)** and also than those reported in previous works, in terms of the AUC, KS, and F1 scores. This suggests that the message-wise sentiment evolution analysis and derived features improve the predictive power of the classification algorithm at hand, having performance gains in terms of the AUC of about 10-14% in any case. Moreover, when observing the validation metrics for the **(B+LW{NP})** experiment, it is evident that the AUC and the KS metrics further increase, so that the model is separating better the *detractors* from *promoters*, even when the problem had a higher class imbalance. This might be because the input space for *passive* users is more diverse and sparse than for both detractors or promoters. Thus, suggesting that there might not be clear patterns to learn from the *passives* population so that extensive inspection in that regard is encouraged in future works.

Figure 3 show a plot of the scorecard for our XGBoost model for the **(B+LW)** experiment. It can be seen that overall, the relative number of users classified as *non-promoters* does increase monotonically as the score decreases, which is a good indicator in this case. The score represents the probability of a sample belonging to the *promoters* class. In that regard, the model is capable of separate between each class, though it is not entirely sure about none, given that the bins with scores higher (lower) than 0.8 (0.1) are empty. This can be due to a fundamental selection bias, as users contact CS only when they have issues with the product or service. Correspondingly, their recommendation behavior might be biased and lowered by itself considering the inherent nature of the CS interactions. However, the here presented results, are significantly better to those obtained in previous works (11) considering explicit quantitative service ratings, for instance.

In addition, we use the SHapley Additive exPlanations (SHAP) (34) algorithm to obtain the feature-level interpretability of our ensemble model. SHAP values can tell us the extent and relationship to which each feature in a model has contributed to the final prediction. Figure 4 shows the SHAP values for our **(B + LW)** experiment. These values are in very good agreement with a rather intuitive

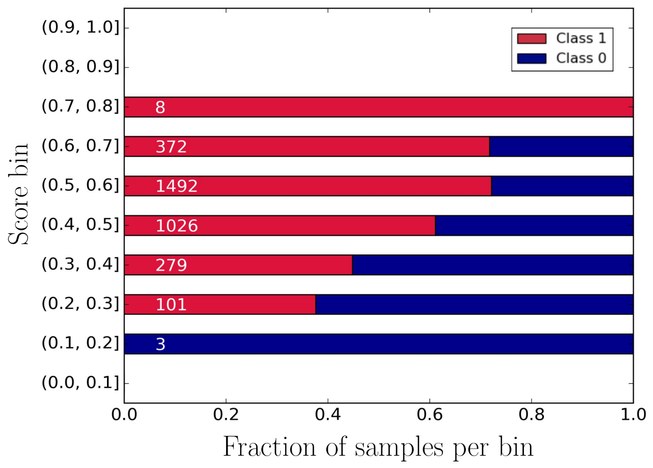


Figure 3: Relative distribution of number of samples belonging to each respective *promoters/non-promoters* class per score bin. The score represents the probability of each sample belonging to the *promoters class* (Class 1) according to the trained XGBoost model for our **(B+LW)** experiment. Numbers inside each bin represent the number of samples classified per each score interval.

reasoning, as well as previous similar studies based on text reviews (7). For instance, we can see that a higher number of messages from a customer to the CS center leads to a lower value on the classification task, i.e., a lower probability of being a *promoter* user. This is very sensible given that if a single client has sent a very large number of messages, it might indicate angriness or frustration, then leading to a rather poor customer experience, and thus a low recommendation decision. A similar interpretation can be extended to the number of messages having the lowest possible sentiment in our discrete sentiment scale, as well as the other corresponding features.

Furthermore, when interpreting variables such as the last sentiment of the longest conversation, the slope of the linear fitting (Fig. 2), or the average sentiment over all CS conversations, we can see that the larger these values, the more likely a person to recommend the service is. This is very relevant given that a high slope value, means that a given conversation started rather neutral/negative but ended rather positive, and this might be associated with low customer frustration and then better recommendation behavior overall, thus validating our sensible hypothesis stated in Section 3.

Very relevantly, it can be observed that the concavity analysis demonstrated that rather low acceleration values (i.e., low mean value of the numerical second derivative) leads to rather low SHAP values, whereas rather convex-like sentiment series (high acceleration values) leads to higher SHAP values. This is also in very good agreement with intuitive reasoning, as for a twice-differentiable function, if its acceleration is positive, then the line-wise series is concave upward. Likewise, if the second derivative is negative, then the graph is concave downward. Therefore, the more convex-like the line-wise sentiment series is, the more likely the user to be a *promoter* is. For instance, the overall evolution of a convex-like conversation would be, in general, a conversation starting very gentle and positive, then swinging through a rather neutral-negative stage, then ending rather positively. Therefore such a convex-like pattern is also associated with better recommendation behavior, whereas concave-like shape results generally in the opposite outcome.

The more noisy and volatile a series is, the more likely the user is to be a *promoter*, as confirmed by the coefficient of variation. This might be due to the fact that non-*promoter* users might tend to have a more negatively biased behavior, causing also the overall standard deviation of the distribution to be small. Furthermore, for the sum of other features in the last row of Fig. 4 we can see that their overall aggregated SHAP values do not have strong predicting power, yet can further segment our target population.

In general, the here presented results represent the first analysis and assessment of the actual predictive powers of the message-wise sentiment evolution throughout chat-based CS interactions on the recommendation behavior of individual users. As such, this work represents a highly flexible and interpretable framework for proactive intervention of *promoter/non-promoter users* which is

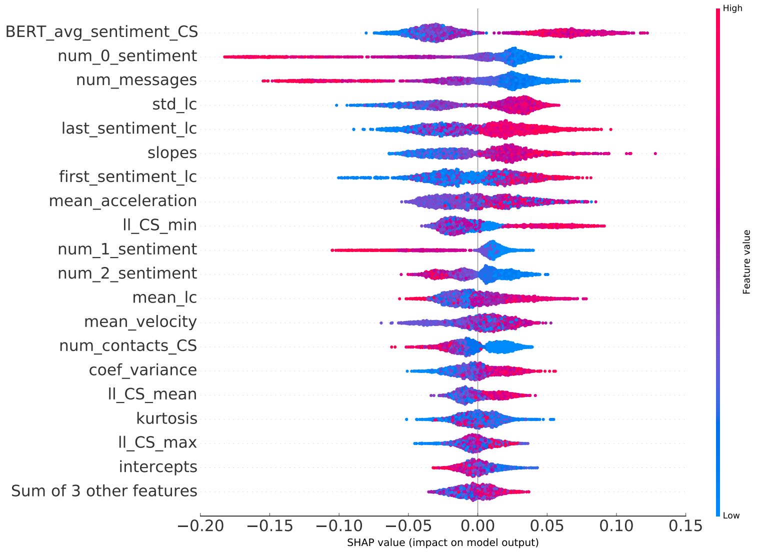


Figure 4: SHapley Additive exPlanations (SHAP) values for individual features of the line-wise sentiment analysis considering *passive* users (B + LW) experiment.

particularly relevant, given that in our case-study the NPS survey is sent periodically once every few months. Therefore, the framework could be used in other classification tasks regarding other business-driven metrics or customer ratings based on chat-based interactions in any particular domain.

Finally, we would like to discuss that even though we do not have a clear baseline model present in the literature to which we can fairly compare against, we do observe that our baseline is comparable in performance to different available studies in the field (5; 11; 7; 6). Generally, the lack of baselines or benchmark comparisons are grounded in the fact that there are no publicly available datasets containing chat-based conversations with NPS annotations obtained directly from the users; as these datasets typically belong to private companies, highly regulated by the authorities, as in our study case. However, there is no foreseeable fact that would prevent the here presented results and employed ML techniques to be generally extensible to any industry domain.

6 Conclusions

In this work, we propose and evaluate our framework on the NPS classification problem in the field of CS interactions, showing its value, flexibility and interpretability in a real-world case study, where rating annotations were provided directly by the users in corresponding surveys. Our results show that it is possible to predict the recommendation decision of users based on dynamic sentiment classification of chat-based data sources employing transformer-based methods. Results show performance gains of about 10-14% obtained when considering a sentiment evolution analysis versus a purely aggregated, review-based, sentiment classification. Moreover, our explainable features explicitly allowed us to draw important and intuitive insights regarding the complex relations that do arise between the recommendation behavior of a given user and the sentiment evolution of their CS messages. More importantly, we here present a framework that can be easily extended to other prediction tasks in any business environment, considering any customer ratings of interest. We therefore hope that this work could spark interest among data-driven companies, and will inspire other works in this research area, where inherent attributes related to the evolution of conversations could be exploited to further develop state-of-the-art techniques and applications.

Acknowledgements

We would like to thank Luisa F. Roa, Ana M. Quintero, Jaime Acevedo, V. Maya, and R. Saavedra for their insightful ideas and valuable discussions during the development of this research project.

References

[1] F. F. Reichheld, “The one number you need to grow,” *Harvard Business Review*, vol. 81, pp. 46–54, 2003. [Online]. Available: www.hbr.org

[2] N. Raassens and H. Haans, “Nps and online wom: Investigating the relationship between customers’ promoter scores and ewom behavior,” *Journal of Service Research*, vol. 20, pp. 322–334, 8 2017.

[3] A. Ando, R. Masumura, H. Kamiyama, S. Kobashikawa, and Y. Aono, “Hierarchical lstms with joint learning for estimating customer satisfaction from contact center calls,” vol. 2017-August. International Speech Communication Association, 2017, pp. 1716–1720.

[4] J. Bockhorst, S. Yu, L. Polania, and G. Fung, “Predicting self-reported customer satisfaction of interactions with a corporate call center,” *ECML PKDD 2017: Machine Learning and Knowledge Discovery in Databases*, pp. 179–190, 2017. [Online]. Available: https://github.com/cyberyu/ecml2017.

[5] M. Siering, A. V. Deokar, and C. Janze, “Disentangling consumer recommendations: Explaining and predicting airline recommendations based on online reviews,” *Decision Support Systems*, vol. 107, pp. 52–63, 3 2018.

[6] J. Auguste, D. Charlet, G. Damnati, F. Bechet, and B. Favre, “Can we predict self-reported customer satisfaction from interactions?” *2018 IEEE International Conference on Acoustics, Speech and Signal Processing : proceedings*, pp. 1–5, 2018.

[7] S. Chatterjee, “Explaining customer ratings and recommendations by combining qualitative and quantitative user generated contents,” *Decision Support Systems*, vol. 119, pp. 14–22, 4 2019.

[8] S. Rose, R. Sreejith, and S. Senthil, “Social media data analytics to improve the customer services: The case of fast-food companies,” *International Journal of Recent Technology and Engineering*, vol. 8, pp. 6359–6366, 7 2019.

[9] B. Vanderheyden, Y. Xie, and M. Rachumallu, “Net promoter sentiment classifier using ohpl-al,” 2019, pp. 2494–2502.

[10] C. Lewis and M. Mehmet, “Does the nps® reflect consumer sentiment? a qualitative examination of the nps using a sentiment analysis approach,” *International Journal of Market Research*, vol. 62, pp. 9–17, 1 2020.

[11] I. Markoulidakis, I. Rallis, I. Georgoulas, G. Kopsiaftis, A. Doulamis, and N. Doulamis, “A machine learning based classification method for customer experience survey analysis,” *Technologies*, vol. 8, p. 76, 12 2020.

[12] S. Baehre, M. O’Dwyer, L. O’Malley, and N. Lee, “The use of net promoter score (nps) to predict sales growth: insights from an empirical investigation,” *Journal of the Academy of Marketing Science*, 2021.

[13] P. K. Jain, R. Pamula, and G. Srivastava, “A systematic literature review on machine learning applications for consumer sentiment analysis using online reviews,” p. 100413, 8 2021.

[14] I. Markoulidakis, I. Rallis, I. Georgoulas, G. Kopsiaftis, A. Doulamis, and N. Doulamis, “Multiclass confusion matrix reduction method and its application on net promoter score classification problem,” *Technologies*, vol. 9, p. 81, 11 2021.

[15] M. Zaki, D. Kandeil, A. Neely, and J. R. Mccoll-Kennedy, “The fallacy of the net promoter score: Customer loyalty predictive model why this paper might be of interest to alliance partners,” *University of Cambridge*, 2016. [Online]. Available: www.cambridgeservicealliance.org

[16] A. Garas, D. Garcia, M. Skowron, and F. Schweitzer, “Emotional persistence in online chatting communities,” *Scientific Reports*, vol. 2, no. 1, p. 402, May 2012. [Online]. Available: https://doi.org/10.1038/srep00402

[17] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, “Bert: Pre-training of deep bidirectional transformers for language understanding,” 2018, cite arxiv:1810.04805Comment: 13 pages. [Online]. Available: http://arxiv.org/abs/1810.04805

[18] X. Gu, K. M. Yoo, and J.-W. Ha, “Dialogbert: Discourse-aware response generation via learning to recover and rank utterances,” 2021. [Online]. Available: www.aaai.org

[19] Y. Zhang, S. Sun, M. Galley, Y.-C. Chen, C. Brockett, X. Gao, J. Gao, J. Liu, and B. Dolan, “Dialogpt: Large-scale generative pre-training for conversational response generation,” 11 2019. [Online]. Available: http://arxiv.org/abs/1911.00536

[20] S. Roller, E. Dinan, N. Goyal, D. Ju, M. Williamson, Y. Liu, J. Xu, M. Ott, K. Shuster, E. M. Smith, Y.-L. Boureau, and J. Weston, “Recipes for building an open-domain chatbot,” 4 2020. [Online]. Available: http://arxiv.org/abs/2004.13637

[21] N. A. Vidya, M. I. Fanany, and I. Budi, “Twitter sentiment to analyze net brand reputation of mobile phone providers,” *Procedia Computer Science*, vol. 72, pp. 519–526, 2015.

[22] D. Vélez, A. Ayuso, C. Perales-González, and J. T. Rodríguez, “Churn and net promoter score forecasting for business decision-making through a new stepwise regression methodology,” *Knowledge-Based Systems*, vol. 196, p. 105762, 2020. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0950705120301684

[23] M. Munikar, S. Shakya, and A. Shrestha, “Fine-grained sentiment classification using bert,” in *2019 Artificial Intelligence for Transforming Business and Society (AITB)*, vol. 1, 2019, pp. 1–5.

[24] M. Zaib, D. H. Tran, S. Sagar, A. Mahmood, W. E. Zhang, and Q. Z. Sheng, “Bert-coqac: Bert-based conversational question answering in context,” *ArXiv*, vol. abs/2104.11394, 2020.

[25] H. Bao, L. Dong, F. Wei, W. Wang, N. Yang, X. Liu, Y. Wang, S. Piao, J. Gao, M. Zhou, and H.-W. Hon, “Unilmv2: Pseudo-masked language models for unified language model pre-training,” in *Proceedings of the 37th International Conference on Machine Learning*, ser. ICML’20. JMLR.org, 2020.

[26] W. Shi and V. Demberg, “Next sentence prediction helps implicit discourse relation classification within and across domains,” in *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*. Hong Kong, China: Association for Computational Linguistics, Nov. 2019, pp. 5790–5796. [Online]. Available: https://aclanthology.org/D19-1586

[27] Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer, and V. Stoyanov, “Roberta: A robustly optimized bert pretraining approach,” *ArXiv*, vol. abs/1907.11692, 2019.

[28] T. Chen and C. Guestrin, “Xgboost: A scalable tree boosting system,” *CoRR*, vol. abs/1603.02754, 2016. [Online]. Available: http://arxiv.org/abs/1603.02754

[29] R. G. Mantovani, A. L. D. Rossi, J. Vanschoren, B. Bischl, and A. C. P. L. F. de Carvalho, “Effectiveness of random search in svm hyper-parameter tuning,” *2015 International Joint Conference on Neural Networks (IJCNN)*, pp. 1–8, 2015.

[30] C. Goutte and E. Gaussier, “A probabilistic interpretation of precision, recall and f-score, with implication for evaluation,” in *Advances in Information Retrieval*, D. E. Losada and J. M. Fernández-Luna, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2005, pp. 345–359.

[31] J. A. Hanley and B. J. McNeil, “The meaning and use of the area under a receiver operating characteristic (roc) curve.” *Radiology*, vol. 143 1, pp. 29–36, 1982.

[32] *Kolmogorov–Smirnov Test*. New York, NY: Springer New York, 2008, pp. 283–287. [Online]. Available: https://doi.org/10.1007/978-0-387-32833-1_214

[33] I. Rallis, I. Markoulidakis, I. Georgoulas, and G. Kopsiaftis, “A novel classification method for customer experience survey analysis.” ICST, 6 2020, pp. 1–9.

[34] S. M. Lundberg and S. Lee, “A unified approach to interpreting model predictions,” *Proceedings of the 31st International Conference on Neural Information Processing Systems (NIPS)*, p. 4768–4777, 2017.