Proceedings of the LREC2020 Industry Track , pages 15–20
Language Resources and Evaluation Conference (LREC 2020), Marseille, 11–16 May 2020

Measuring the Polarity of Conversations between Chatbots and Humans: a use Case in the Banking Sector

Guillaume Le Noé-Bienvenu, Damien Nouvel, Djamel Mostefa
OrangeBank, Inalco, OrangeBank
guillaume.lenoe.bienvenu@gmail.com, damien.nouvel@inalco.fr, djamel.mostefa@orangebank.com

Abstract

This paper describes a study on opinion analysis applied to both human to chatbot conversations, but also to human to human conversations with data coming from the banking sector. Applying a polarity classifier model to conversations provides insights and visualisations of the satisfaction of users at a given time and its evolution. We also conducted a study on the evolution of the opinion on the conversations started with the chatbot and then transferred to a human agent. This work illustrates how opinion analysis techniques can be applied to improve the user experience of the customers but also detect topics that generate frustrations with a chatbot but also with human experts.

Keywords: Text Mining, Opinion Analysis, Chatbot, Polarity

1. Introduction

1.1. Scope and Aim

Orange Bank is a mobile bank launched in late 2017 and for which the main channel of communication with its customers is Djingo, a text chatbot. Available 24/7 by chat, Djingo, Orange Bank virtual advisor, is the customers first point of contact. Since the launch of Orange Bank in November 2017, more than 2,5 million conversations have been initiated by our clients with Djingo (an average of 100,000 conversations per month), 50% of which are handled entirely by the virtual advisor (without any redirection to the Customer Relationship Centre). Since Djingo is the first point of contact of Orange Bank clients, all chat conversations with a human agent started with Djingo. We are hence able to measure the evolution of the polarity within the same conversation between a customer and Djingo and then between the customer and the human operator.

In this context, opinion mining may be used to deliver in real time an understanding of the customer relationship for a given service. It could also be used to detect annoyance, irritation or angriness at an early stage of the conversation with Djingo in order to quickly redirect the user to a human expert. In this situation, opinion mining is also useful to detect topics and to provide insights about customer's satisfaction.

Our work focuses on the evolution of customer's opinion, both on conversations or messages within conversation. We implemented an opinion detector that has been evaluated, and plugged into the history of online conversations between customers and chatbot or human support desk. This work provides the customer support service visualisations of the evolution of customer's satisfaction depending on themes, as well as information on how much the bot and humans give satisfaction to the customers.

1.2. State of the Art

1.2.1. Opinion Analysis

Whereas a lot of work has been done in the opinion analysis field, most of it was directed towards product reviews, e.g. identifying the sentiment linked to the aspects of an object or its entities (Liu, 2012), but a few work was done towards written conversations, especially with a chatbot. (Hancock et al., 2019) used the estimation of user satisfaction to improve the learning process of the chatbot. Tools to work on polarity and emotions based on rules such as VADER (Hutto and Gilbert, 2014) or SentiWordNet (Esuli and Sebastiani, 2006) are freely usable, but remain only for the English language. For French, resources are also available, such as the CANÉPHORE Corpus (Lark et al., 2015), but remain mostly specific to tweets. In this paper, we present a few cases (mostly graphs) in which opinion analysis could help giving valuable information with written talks. We focus on the polarity, defined by Zhang and Ferrari (2014) as the property of a text being positive, negative or neutral.

1.2.2. Text Classification

Text classification is a well known task in NLP, and a reasonably efficient technique to perform it consists of using a TF-IDF (Salton and Buckley, 1988) representation of the data combined with a support vector machine classifier (SVM) on it. This approach has since be giving satisfactory results. (Joachims, 1998; Pang et al., 2002; Lilleberg et al., 2015) Deep learning methods can also be used for text classification. In particular, convolutional neural networks obtain very high scores for text classification (Kim, 2014), but require more time and examples for training. Also, the winners of many challenges in NLP for the French language used TF-IDF+SVM models as the one used for DEFT 2015 (Thierry Hamon, 2015) or during the Hackatal 2018 [1]).

1.3. The Djingo Chatbot

Djingo is Orange Bank's conversational agent, available 24/7 for its 3,000 daily users. It is able to understand 390 intentions and has more than 1,000 answers adapted to the user's needs. Djingo is used both as a Frequently Asked Questions (FAQs) system (products marketed e.g. withdrawal fees, time to deliver a cheque book, etc.) and as an assistant to perform actions related to the customer account (ordering a cheque book, blocking the card, etc.). FAQ-oriented answers are usually the same for all customers, whereas requests performing an action trigger an operation

[1] https://hackatal.github.io/2018/

that depends on the account.

For example, if a user wishes to order a checkbook, Djingo will check if the user is identified, if there is currently no checkbook order, if the user can order it, and so on. At each step, depending on the elements received through a programmatic interface (APIs), Djingo provides the user with an appropriate answer. During the conversation, themes and intentions are detected by the IBM Watson module. To date, there are about 60 themes: Orange-Bank, app-site-info, app-site-problem, insurance-info, termination insurance, etc. Conversations can include several themes. If the user asks a question that Djingo does not have the answer to, or detects that the user is unable to make himself understood, he suggests that the user should be redirected to an advisor.

2. Opinions for messages and conversations

2.1. Chatbot Corpus

The corpus used in this article consists of 1,566,060 unique conversations from November 2017 to March 2019, containing 5,775,227 messages. Most of the messages sent by the users contain a small number of words (around 4.6 words per message) and are often describing the question using simple words. The size of the lexicon is quite important with around 144k entries due to important number of misspellings and typos.

2.2. Annotation

As we focus on the polarity of messages, we built a gold-standard, by manually annotating 3,053 randomly picked user messages from the corpus. Each message is considered as positive, negative or neutral, following the 2015 DEFT annotation guide (Thierry Hamon, 2015).

The annotation was made by two different annotators, giving a Cohen's kappa coefficient of 0.72. One particular issue during the annotation process was the case of greeting messages. We notice that in our data set, the user uses greetings for 83.96% of the conversations with a human agent, and only 18.99% of those with the chatbot. This gives us a clear indication of the behaviour of the user depending on the interlocutor. From an opinion perspective, we then assumed those greetings were positive and annotated them accordingly.

Table 1 gives examples of annotated data.

Unsurprisingly, the annotations are unbalanced: 5.01% of the messages are positive, 73.96% of them neutral and 21.03% negative. This was expected as users usually come with problems and questions regarding bank services and operations. Indeed, the company wants to maximise the satisfaction of users at the end of the interaction, while limiting the number of agents hired for this task.

2.3. Classification

This annotated data set was then divided over a train (4/5) and test parts (1/5). The train data was then pre-processed by computing a TF-IDF transformation. We tested several classical machine learning models using the sklearn API (Buitinck et al., 2013). Results are reported in Table 2.

Message (*translated*)	Annotation
Merci orange pour les 80 euros *Thank you orange for the 80 euros*	positive
Merci, bonne soirée *Thank you, have a nice evening*	positive
OK, super ! *Okay, great!*	positive
Je souhaiterai ouvrir un compte *I'd like you register an account*	neutral
Savoir si ma demande a été traitée *Find out if my request has been processed*	neutral
Quelles sont vos offres pour les étudiants ? *What are your offers for students?*	neutral
Cela ne repond pas a la question *This doesn't anwser the question*	negative
Non merci je suis très contrariée *No, thanks, I'm very upset.*	negative
Vous servez à rien *You're useless.*	negative

Table 1: Example of annotated messages

ML classifier	Precision	Recall	F1
SVM	0.90	**0.81**	**0.85**
MaxEnt	**0.92**	0.75	0.82
Nultinomial Naive Bayes	**0.92**	0.63	0.70
SGDClassifier	0.91	0.79	0.84

Table 2: Performance of Opinion Classifier (macro)

As the SVM classifier provides the best F1 score, we ran a grid search on several parameters to optimize this model configuration. We obtained an average 0.85 F1 macro score (0.91 F1 micro). The neutral class obtains the best score (0.95 F1), while positive and negative classes have much lower F1 scores (0.82 and 0.76, respectively). Those results were obtained using the NLTK TweetTokenizer (Bird et al., 2009), without any other preprocessing (no lemmatization, case is kept as it is) and linear kernel for the SVM. Finally, the model was used to classify all messages of the corpus.

3. Conversation Polarity by Themes

3.1. Rules to Predict Conversations Polarity

To have a global view of user experience, one needs to compute an opinion score for each conversation. As the data was annotated by messages, simple rules were implemented to predict the polarity of an entire conversation based on the opinion of its messages. A conversation is then:

- **neutral** when all messages are such,
- **positive** when at least one of its messages is such and the remaining is neutral or positive,
- **negative** when at least one of its messages is such and the remaining is neutral or negative,
- **mixed** otherwise.

	Number of messages	%	Number of conversations	%
Positive	460,744	3.98	190,057	7.30
Neutral	9,903,323	85.50	1,746,296	67.07
Negative	1,218,890	10.52	541,549	20.80
Mixed	_	_	125,641	4.83
Total	1,1582,957	100	2,603,543	100

Table 3: Proportion of messages and conversations in the corpus

Using these simple rules, table 3 shows the proportion of messages and conversations in the corpus. These rules allowed us incidentally to get strongly oriented conversations (e.g. a conversation where nearly all of its messages are negative would be very negative).

3.2. Histogram

The first representation we get from this labelling is the proportions of the conversation classes (positive, negative, neutral and mixed) depending of the detected themes. Figure 3 (annex) shows those proportions for December 2018. For instance, *the app_site* theme (related to the behaviour of the Bank's application) has more than 50% of its conversations being negative where the *cheque* theme remains globally neutral, this can be explained by the fact that this operation is rarely problematic. The representation of polarity gives us a rough idea of where to improve the user's experience. This type of plot can also be drawn for a different time scale (year, day, etc.).

3.3. Heatmap

In the previous section, we presented a way of drawing the proportions of the conversation classes for a particular timelapse. However, this type of plot does not give us information about the evolution of this proportions across a time scale. E.g. on Figure 3, the *app_site* theme has a strong part of negative conversations but one can wonder if those proportions were similar through the year, whether it was due to a temporary failure, or if it was a general trend.
In order to represent a potential evolution of those proportions, we proposed a heatmap showing this evolution of the opinion by theme. To get a polarity score as a single numerical value for each case, a rule was implemented, consisting of adding the neutral and positive proportions of conversation and subtracting the negative. This was given by the following formula:

$$PS(th,t) = \frac{N(neu,th,t) + N(pos,th,t) - N(neg,th,t)}{NTotalConversations(th,t)}$$

Where

- *th*: the theme of the conversation
- *t*: a date
- *N(pol, th, t)*: the number of polarity (pol as negative, positive or neutral) conversations of the theme *th* at time *t*

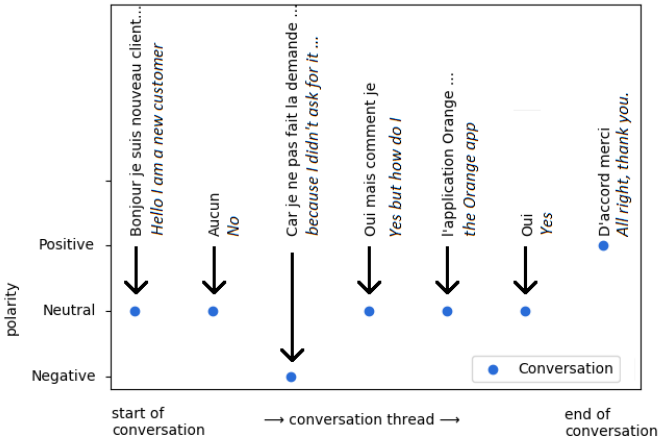

Figure 1: Single Conversation Polarity Graph

- *NTotalConversations(th, t)*: the total number of conversations of the theme *th* at time *t*

Figure 4 (annex) reports the heat map from November 2017 to March 2019. The bluer the case is the higher proportion of positive conversations the corresponding theme has. Conversely the red cases indicate negative conversations. One can then watch the changes in the proportions of cases throughout the months. For instance, we clearly see that the *Bonus* theme in March 2018 had its lowest polarity score, but its polarity score increased in the next few months. As in the previous section, this plot can also be drawn for a different time scale.

3.4. Graph of Polarity

We have been then studied the way polarity of messages changes for a single conversation, especially when the user switches from a chatbot to an agent. In order to have a visual output, we converted the polarity (negative, neutral, positive) of each message of the conversation to an integer (0 for negative, 1 for neutral, 2 for positive). This provides us with a list of integers that we can plot on a basic polarity graph, as reported in Figure 1.
Since the conversations do not have the same length (different number of user messages), we converted the lists of integers representing the polarity of the user messages into lists of floats of fixed size. The size of the output lists can be modified as an optional parameter.[2] We then compute the average of each point of the list. Figure 2 show the result of the output with a padding of dimension 20.
On Figure 2, we first notice that for both types of users (redirected and non-redirected or full IA), the conversation starts with the same polarity (neutral) on average. After the first third of the conversation, people who are not redirected see the polarity of their conversation stagnate around a value slightly below neutral, while people who will be redirected see the polarity of their conversation decrease until an agent takes over. As soon as people are cared for by

[2]Code available at `https://github.com/GuillaumeLNB/perso/blob/master/rounding.py`

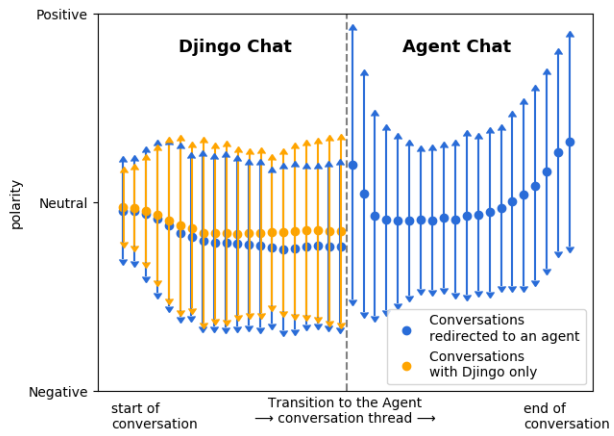

Figure 2: Polarity graph

a counsellor, the polarity of the conversation takes a more positive trend (signs of politeness such as "hello" are labelled as positive and are more present in conversations with a human being). This is followed by a more neutral phase, which generally corresponds to the advisor's information gathering. At the end of the conversation, the trend is clearly becoming positive, we hypothetize that satisfying solutions are being proposed by the human agent.

4. Discussion

There are however some limitations to the approaches discussed in this paper. First of all, the classification is based on annotation, and it is quite difficult to annotate into only three polarity classes. In the example: "*Mon épouse est décédé et je souhaite réaliser une demande de succession / My wife has died and I want to make a succession request*", the user of the conversational agent reports a past event as well as the willingness to take action. However, the part "*Mon épouse est décédé / My wife died*" would have been annotated as negative, while the part "*je souhaite réaliser une demande de succession / I wish to make an estate application*" would have been annotated neutral. A new class "positive-negative mix" could have been used as in DEFT 2018 [3], but would have required a much more subtle and fine-grained annotation work.

Secondly, polarity is useful information, but does not indicate the subjectivity of the message. There is a significant difference between a user complaining about a particular Orange Bank service (example: *Ma carte bancaire ne marche pas / My credit card doesn't work*, negative polarity) and a dissatisfied user without a specific reason being stated (example: *Orange c'est vraiment de plus en plus pourri ! / Orange is really getting crap!*, negative polarity).

Thirdly, the transition from the polarity of the messages to the polarity of the conversation was carried out with a rule-based approach, creating a mixed class. This class does not take into account the intensity of certain messages. In the example in Table 4, the conversation has a mixed polarity

[3] https://perso.limsi.fr/pap/DEFT2018/annotation_guidelines/index.html

Message (*translated*)	Predicted Polarity
bonjour, *hello,*	positive
association loi 1901 peut elle ouvrir un compte chez vous? *Can a association loi 1901 open an account with you?*	neutral
compte + association oi 1901 *account + aossociation 1901 [l]aw*	neutral
je ne parle pas aux robots, connards *I don't talk to robots, assholes.*	negative

Table 4: Example of a conversation classified as mixed where it should have been negative

(presence of positive and negative), but remains very negative by the presence of the last message. An annotation at the level of the conversation would probably have classified this conversation as negative, but would not have made a difference between this very negative and a less negative conversation.

Finally, the thermal map display gives us an overview of the evolution of the polarity, but does not detail the reasons of this variation. In addition, we did not find a correlation for all themes between their monthly polarity scores and their redirection rates. We are wondering if this metric is suitable for comparing these data.

5. Conclusion

In this paper, we have presented several applications of opinion analysis on chatbot conversations. By developing a model for polarity analysis (positive, negative, neutral) using standard machine learning algorithms, we were able to use the data to highlight trends. A real corpus of more than 1.5 million of conversations between Orange bank customers and Djingo was used for this study. For privacy and confidential reasons, this corpus can not be shared at that time but it may be released in the future after anonymization of all personal data.

This analysis allowed us to look at which topics of the conversational agent show the most customer satisfaction or dissatisfaction, on a time scale. It also provides the opportunity to bring out very focused conversations (very positive or negative) from the corpus for educational purposes for customer relationship centre agents. Finally, this tool makes it possible to obtain a quantification of the customers' opinions on the spot.

6. Bibliographical References

Bird, S., Klein, E., and Loper, E. (2009). *Natural Language Processing with Python: Analyzing Text with the Natural Language Toolkit*. O'Reilly, Beijing.

Buitinck, L., Louppe, G., Blondel, M., Pedregosa, F., Mueller, A., Grisel, O., Niculae, V., Prettenhofer, P., Gramfort, A., Grobler, J., Layton, R., VanderPlas, J., Joly, A., Holt, B., and Varoquaux, G. (2013). API design for machine learning software: experiences from the scikit-learn project. In *ECML PKDD Workshop: Languages for Data Mining and Machine Learning*, pages 108–122.

Esuli, A. and Sebastiani, F. (2006). Sentiwordnet: A publicly available lexical resource for opinion mining. In *In Proceedings of the 5th Conference on Language Resources and Evaluation (LREC 06*, pages 417–422.

Hancock, B., Bordes, A., Mazaré, P., and Weston, J. (2019). Learning from dialogue after deployment: Feed yourself, chatbot! *CoRR*, abs/1901.05415.

Hutto, C. J. and Gilbert, E. (2014). Vader: A parsimonious rule-based model for sentiment analysis of social media text. In Eytan Adar, et al., editors, *ICWSM*. The AAAI Press.

Joachims, T. (1998). Text categorization with support vector machines: Learning with many relevant features.

Kim, Y. (2014). Convolutional neural networks for sentence classification. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing, EMNLP 2014, October 25-29, 2014, Doha, Qatar, A meeting of SIGDAT, a Special Interest Group of the ACL*, pages 1746–1751.

Lark, J., Morin, E., and Peña Saldarriaga, S. (2015). Canéphore : un corpus français pour la fouille d'opinion ciblée. In *Actes de la 22e conférence sur le Traitement Automatique des Langues Naturelles*, pages 418–424, Caen, France, June. Association pour le Traitement Automatique des Langues.

Lilleberg, J., Zhu, Y., and Zhang, Y. (2015). Support vector machines and word2vec for text classification with semantic features, 07.

Liu, B. (2012). *Sentiment Analysis and Opinion Mining*. Morgan & Claypool Publishers.

Pang, B., Lee, L., and Vaithyanathan, S. (2002). Thumbs up: Sentiment classification using machine learning techniques. In *Proceedings of the ACL-02 Conference on Empirical Methods in Natural Language Processing - Volume 10*, EMNLP '02, pages 79–86, Stroudsburg, PA, USA. Association for Computational Linguistics.

Salton, G. and Buckley, C. (1988). Term-weighting approaches in automatic text retrieval. *Inf. Process. Manage.*, 24(5):513–523, August.

Thierry Hamon, Amel Fraisse, P. P. P. Z. C. G. (2015). Analyse des émotions, sentiments et opinions exprimés dans les tweets: présentation et résultats de l'édition 2015 du défi fouille de texte (deft), 06.

Zhang, L. and Ferrari, S. (2014). Intensité et polarité : un modèle opératoire articulant plusieurs travaux linguistiques. In *Langue française, /4 (num 184), p. . DOI : 10.3917/lf.184.0035.*, pages 35–54.

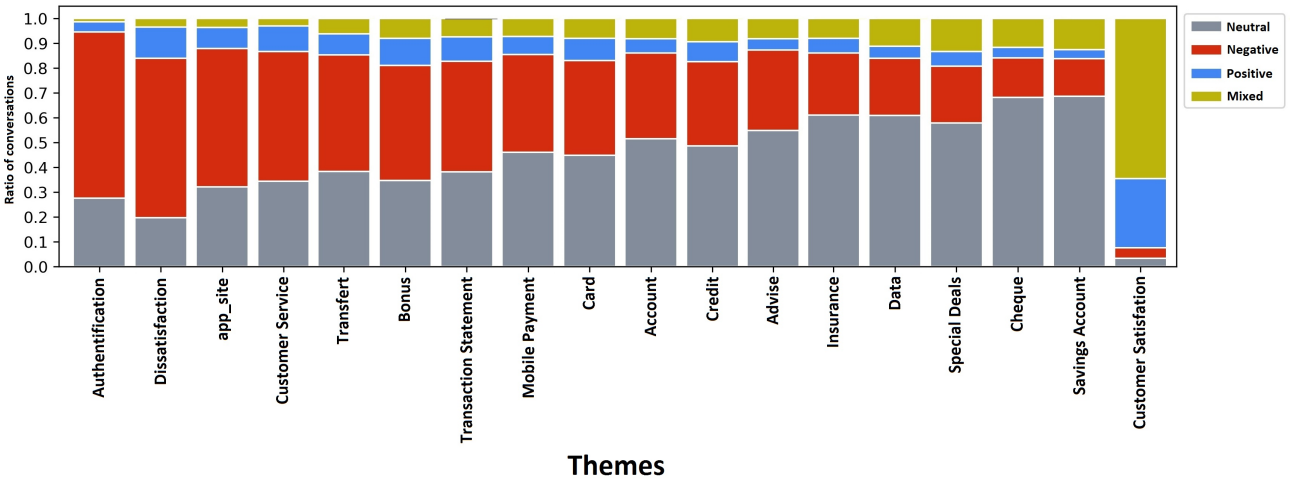

Figure 3: Basic polarity histogram

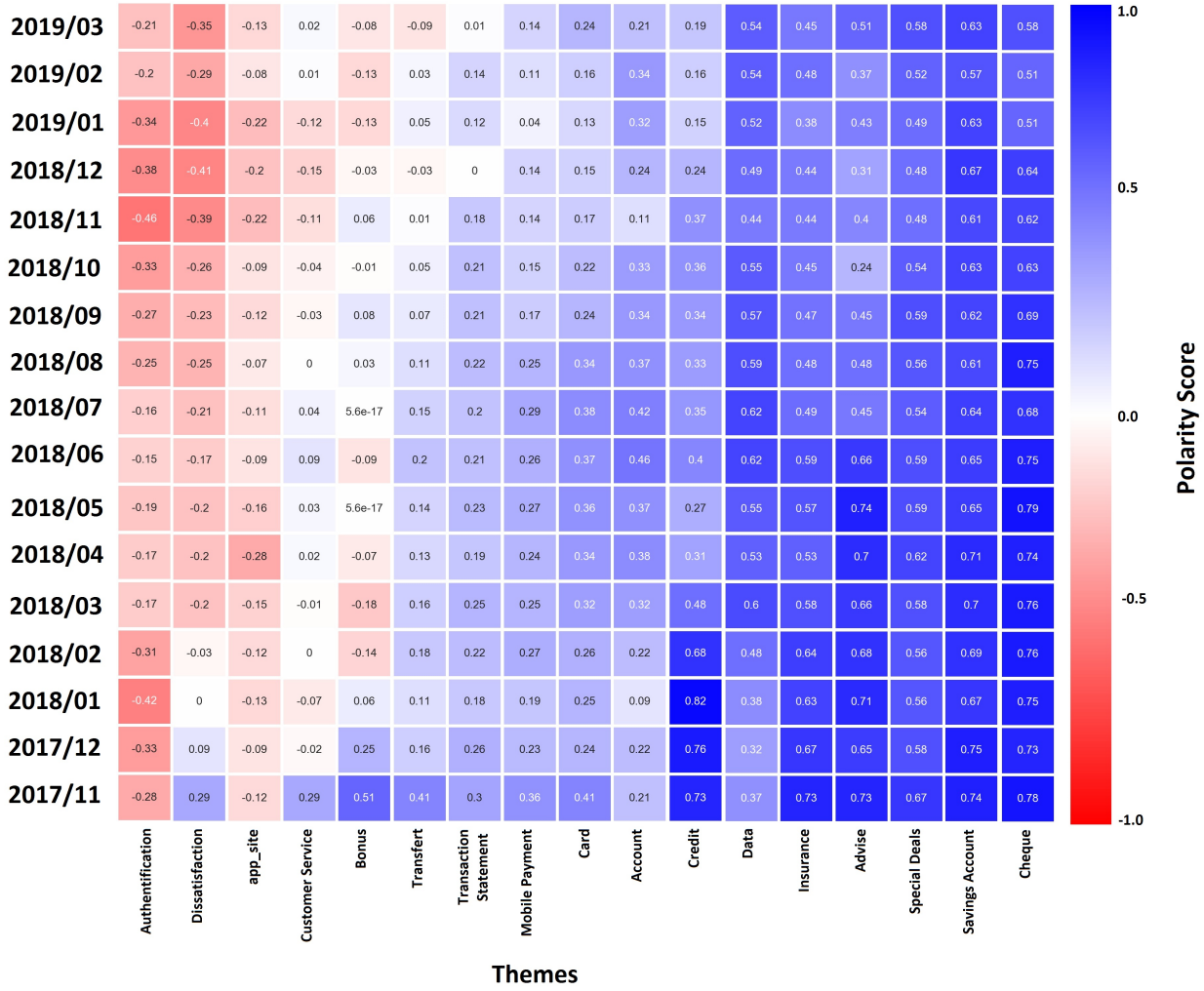

Figure 4: Heatmap of polarity