Using machine translation to improve sentiment classification of Polish restaurant reviews

Tomasz Starakiewicz
Faculty of Economic Sciences
University of Warsaw
tomasz.starakiewicz@student.uw.edu.pl

Dustin Pacholleck
Faculty of Economic Sciences
University of Warsaw
d.pacholleck@student.uw.edu.pl

Abstract **Online reviews are an important gauge of service quality in hospitality industry. Due to the heterogeneity and scale of available data, modern sentiment analysis using these reviews commonly rely on machine learning. We hypothesize that these approaches can be hampered by typos and spelling errors. We investigate whether sentiment classification task can be improved by taking advantage of machine translation automatically provided by common platforms such as Google Maps, which as a side effect removes errors. We analyze a representative sample of Polish reviews of all restaurants in Poland available on Google Maps as of October 2022. We find that review text is best used for sentiment analysis in its original, untranslated form. We also find that lemmatization and stop word removal degrade classifier performance. We hypothesize that typos and spelling errors are semantically important in sentiment analysis.**

I. Introduction

Sentiment analysis is a popular field in natural language processing, which has gained significant attention due to its applications in fields such as social media analysis, market research, and customer feedback analysis. However, sentiment analysis of reviews in languages other than English remains a challenging task, particularly for languages such as Polish which has complex grammar and syntax.

On top of that, online reviews, which are the main subject for sentiment analysis, are known to be rife with spelling errors – based on different studies, humans are expected to misspell from 13% to 26% of words [1][2]. Due to the noise introduced by typing errors, performance of sentiment classifiers is expected to suffer.

While misspellings can carry semantic information on their own, we hypothesize that this is outweighed by the performance degradation due to failures of lemmatization and other preprocessing techniques, due typical pipelines being trained on properly formed text.

In this paper we investigate whether this effect can be mitigated by taking advantage of automated machine translation provided by many services such as Google Maps, which are typically scraped for sentiment analysis. We hypothesize that by using the translated text, instead of the original, the sentiment classification accuracy can be improved.

II. Literature review

Sentiment analysis is the most popular subset of machine learning literature focused on hospitality industry [3]. It is used to analyze consumers' attitude towards product or services [4] in a scalable, automated manner [5]. One of the sub-tasks usually included in the analysis is sentiment classification, where the text is classified into positive/negative or more granular buckets. Methods used in classification range from Naïve Bayes classifiers [6], Support Vector Machines [7], Logistic Regression [9], various Decision Trees-based [6] approaches to deep-learning [10].

Compared to expansive literature on sentiment analysis, the impact of spelling errors on classification accuracy is poorly studied. Some studies have found that pre-processing removing typos and spelling errors had no significant impact on classification task [11], while most recent study found that pre-processing *decreases* the

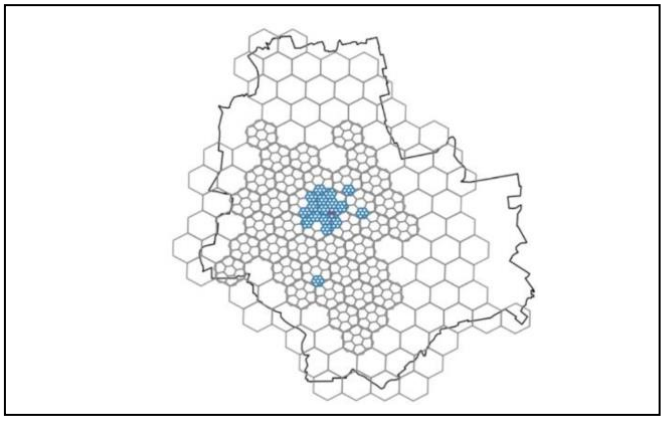
classification performance for deep-learning based approaches, while *increasing* for lexicon based approaches [12]. Using automated machine translation to take advantage of both spelling correction and richer NLP ecosystem, has not been studied yet to our knowledge.

III. Data

A. Sourcing

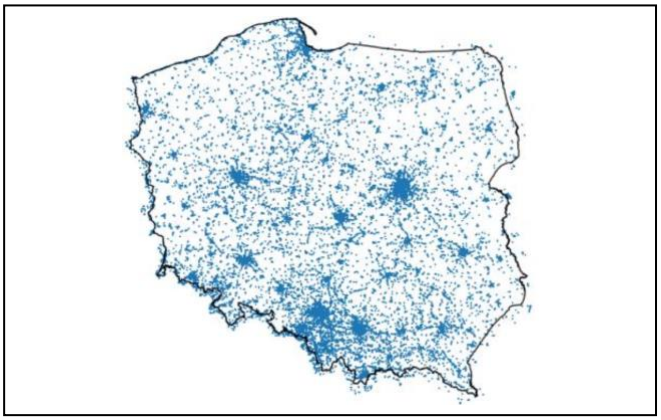
The main data source we use in the paper is Google Maps. We accessed the Google Places API in October 2022 and performed an exhaustive search of all restaurants in Poland. The API was limited to returning a maximum of 60 restaurants, paged 20 results at a time, around a single set of coordinates submitted in the API call. To overcome this limitation, we first split the territory of Poland into hexagons using the H3 library created by Uber [13]. Then we called the API at the center of each hexagon. Given that the results were sorted according to the center from the submitted coordinates, if any of the restaurants returned was outside of a circle circumscribed around the hexagon, the hexagon in question was considered exhausted. If the hexagon was not exhausted, we proceeded to recursively split it into 7 smaller hexagons and call the API for each of the resulting hexagon centers. An exemplary result of the procedure, limited to Warsaw is shown on Fig. 1.

Fig. 1. Exhausted hexagons in Warsaw



The resulting universe contained 41,817 restaurants, shown on Fig. 2.

Fig. 2. Locations of restaurants identified on Google Maps



For each of the restaurants, we accessed the review section at the Google Maps page and scraped it using Selenium. As a result, we were able to collect 6.1M of non-empty text reviews, which represent all reviews of all Polish restaurants present on Google Maps as of October 2022. We collected both the reviews in their original languages, as well as their English translation powered by Google Neural Machine Translation, which is based on transformer architecture.

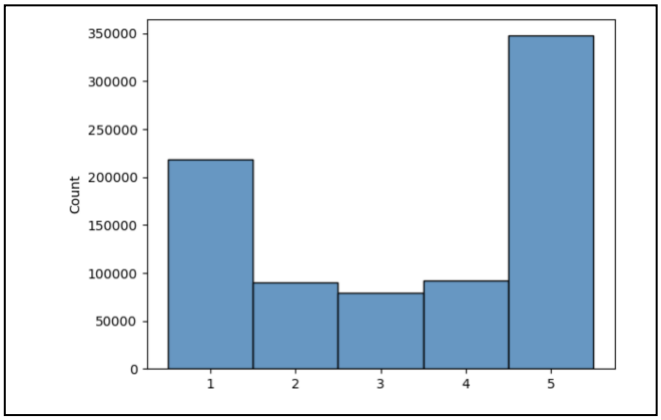
For our study, we selected only reviews longer than 150 characters, which reduced the number of reviews to 827,341. We used Compact Language Detector (*cld3*) created by Google to identify non-Polish reviews. Cld3 is a pre-trained neural network model which uses uni-, bi- and trigrams fed into an embedding layer, then single hidden layer and finally into a softmax layer [14]. We included only reviews classified as "Polish" with probability above 0.5. This reduced the number of reviews to 792,971.

B. Sampling

Each of the text reviews is accompanied by a rating on one to five scale. As expected, the distribution of ratings was highly imbalanced, as shown on Fig. 3, with just two classes responsible for 73% of all the ratings. Interestingly, the ratings also displayed significant polarization, since these

two classes corresponded to the extreme ends of the rating sclae. The reviews were downsampled, with 12,500 reviews drawn randomly from each class, ensuring balanced dataset with equal representation of each rating. This resulted in the final dataset of 62,500 reviews, with 50,000 earmarked for training dataset and 12,500 for the test dataset.

Fig. 3. Distribution of ratings paired with reviews of at least 150 character legnth



C. *Preprocessing*

Both original and translated reviews were processed using pre-trained pipelines provided along with SpaCy library. SpaCy is industrial-grade NLP library offering solutions to process text at scale, using optimized methods for each language task [15]. We have chosen the library to better represent real-life use cases, since the library is widely used by practitioners in commercial applications.

Two pipelines used for preprocessing included a mix of neural network and rule-based components. The reviews were tokenized, then converted into vector representations with 300 dimensions, tagged for parts-of-speech, parsed, and finally fed into lemmatization component. For each of the languages we prepared three versions of tokenized reviews: raw (with punctuation removed), lemmatized, and lemmatized with stop-words removed.

IV. MODELING

To predict the sentiment, we have selected three approaches commonly used in sentiment analysis: Naïve Bayes (NB) classifier, and eXtreme Gradient Boosting (XGBoost) classifier and deep neural network. NB classifier serves as a baseline, due to its popularity as a text classification tool thanks to its speed combined with reasonable effectiveness. We chose XGBoost as a comparator, as it allows for modelling complex, non-linear interactions between features, which is a natural fit for text classification task. Deep learning approach uses a pre-trained language model, BERT, which has become a modern baseline for many NLP tasks [16].

A. *Naïve Bayes classifier*

The NB-classifier is a probabilistic classifier, based on Bayes theorem, known for its simplicity and speed. It relies on two important simplifying assumptions:

- Order of the words doesn't matter (Bag of Words assumption),
- Conditional probabilities of the words occurring in each document class are independent.

For NB classification, we converted tokenized review text into a matrix of token counts, with dictionary based on training data and tokens occurring in fewer than 50 reviews ignored. We considered two candidate models: Multinomial NB classifier and Binary Multinomial NB classifier. To train the latter, we clipped the token frequencies at document level to 1. This is based on a common assumption in sentiment analysis, that word occurrence is more important that word frequency. The standard Multinomial NB classifier outperformed the Binary NB however, so we present only the results of the former.

B. *XGBoost classifier*

XGBoost is a popular system for training ensembles of gradient boosted decision trees, used for ranking, classification and regression tasks [17].

To create features for the model, we converted the tokenized text reviews into a term frequency-inverse document frequency matrix. We ignored tokens occurring in fewer than 200 reviews to reduce computational power requirements. We trained the model using the softmax objective, minimizing the cross-entropy loss function. We performed hyperparameter tuning using Bayesian optimization. We used stratified 3-fold cross-validation and $\varepsilon = 0.3$, 200 boosting rounds and early stopping after 5 rounds without improvement on multinomial log loss. We sequentially tuned depth of boosting trees, L1 regularization (alpha) and feature sampling proportion, with 20 trials for each hyperparameter. Macro F1 score was used as optimization criterion. We arrived at the maximum depth of 8, alpha of 0.2 and no random sampling of features. Each model took ca. 1 minute to train on Nvidia Tesla P100 GPU.

To determine the number of boosting rounds for the final model, we cross-validated the model with tuned hyperparameters again, with $\varepsilon = 0.1$ and 1000 boosting rounds and 10 early stopping rounds. We arrived at 605 boosting rounds for the final model.

C. *Neural Network*

Neural networks are computational systems loosely inspired by the human brain, able to learn complex functions mapping inputs to outputs. We have used an established neural network architecture, Bidirectional Encoder Representations from Transformers (BERT), which is itself based on transformer architecture [18]. Transformer model is a neural network that learns context via self-attention mechanism, by tracking relationships in sequential data, such as sentences [19].

To train the model, we have used the base, cased BERT model which was pre-trained on a large corpus of raw English texts. For original reviews, we have also used HerBERT, which is a BERT-based language model trained on Polish corpus [20]. We have fine-tuned both models by adding a fully connected layer combined with a ReLU activation on top of the embeddings output by BERT, then adjusted the parameters to minimize the cross-entropy loss over multiple batches of reviews. We trained each of the models over 4 epochs, using batch size of 16 reviews, clipping the gradients at norm 1 after each batch. We used 5e-5 and 1e-5 learning rates for BERT and HerBERT respectively. We have frozen parameters for the first 8 out of 12 BERT layers to achieve 2x training speedup without material performance degradation, based on recommendation by Merchant et al. [21] Fine-tuning each model took approx. 45 minutes on Nvidia Tesla P100 GPU.

V. RESULTS

Results of the classification are presented in Table I and Table II. Classifier performance varies greatly across ratings, with both performing best on the extreme ratings. Best score for each preprocessing method is underlined.

The NB-classifier performs consistently better on original Polish text across all preprocessing methods. Out of all methods, no preprocessing (except of punctuation removal) performs best, with performance degrading with each additional preprocessing step added. This performance degradation is also present for the English translation.

XGBoost classifier performs better on translated text, except for lemmatized text with stop words removed, though the overall performance is not better than NB-classifier. Similar to NB-classifier, XGBoost classifier displays performance degradation progressing with each preprocessing

step is present for both original and translated text versions. Notably, XGBoost doesn't perform better than NB-classifier in general.

HerBERT model fine-tuned on original review text outperformed BERT fine-tuned on translated reviews. Both models achieved higher accuracy than NB and XGBoost classifiers. Notably, like previous methods, both models performed substantially worse at classifying non-extreme sentiments.

TABLE I. NB-CLASSIFIER TEST RESULTS

Rating	**Test F1 scores for Naïve Bayes classifier**					
	Raw		***Lemmatized***		***Lemmatized with stop words removed***	
(O)riginal (T)ranslated	**O**	**T**	**O**	**T**	**O**	**T**
1	0.65	0.64	0.64	0.63	0.63	0.61
2	0.43	0.42	0.43	0.42	0.41	0.40
3	0.43	0.41	0.41	0.40	0.41	0.40
4	0.57	0.55	0.56	0.54	0.55	0.53
5	0.80	0.78	0.79	0.78	0.77	0.76
Macro avg.	**0.58**	**0.56**	**0.57**	**0.55**	**0.55**	**0.54**

TABLE II. XGBOOST-CLASSIFIER TEST RESULTS

Rating	**Test F1 scores for XGBoost classifier**					
	Raw		***Lemmatized***		***Lemmatized with stop words removed***	
(O)riginal (T)ranslated	**O**	**T**	**O**	**T**	**O**	**T**
1	0.68	0.69	0.68	0.69	0.67	0.66
2	0.44	0.44	0.44	0.43	0.42	0.41
3	0.45	0.44	0.46	0.44	0.43	0.39
4	0.53	0.54	0.53	0.55	0.53	0.52
5	0.77	0.78	0.77	0.77	0.75	0.75
Macro avg.	**0.57**	**0.58**	**0.57**	**0.58**	**0.56**	**0.55**

TABLE III. NEURAL NETWORK TEST RESULTS

Rating	**Test F1 scores for neural network classifiers**	
	Raw	
	Original / HerBERT	**Translated / BERT**
1	0.74	0.73
2	0.57	0.53
3	0.57	0.54
4	0.63	0.62
5	0.84	0.82
Macro avg.	**0.67**	**0.65**

VI. DISCUSSION

In case of NB-classifier, the differences between original/translated text are small, but consistent. Contrary to our hypothesis, machine translation doesn't improve the classification performance. In light of these results, we hypothesize that mistyped words and misspellings carry valuable information, e.g. occurrence of misspelled words might be indicative of heightened emotional state of the review author, which could in turn be induced by bad experience, that would later result in one-star rating. The other possibility is that the linguistic noise introduced by imperfect machine translation reduces the semantic content of the reviews to the point where it starts to degrade classification performance.

Overall performance of XGBoost is surprising, as we expected the model to improve on NB-classifier due to decision trees' ability to naturally model conditional probabilities of word occurrences. This turned out not to be the case and the significant computing power required to train these ensemble models, combined with slower inference time, do not seem to justify their use.

Nonetheless, the classifier trained on translated text performed better when no preprocessing was applied or only lemmatization, although the differences are minimal, on the verge of rounding error. We believe that these results should be viewed in the context of specific term frequency-inverse document frequency matrix used in feature engineering. As NB-classifier results have shown, the performance degrades when entropy of the training dataset is reduced (by lemmatization or stop words removal). In case of XGBoost, the TF-IDF matrix was created with higher minimum frequency threshold than token count matrix for NB-classifier, due to computing power restrictions. This significantly reduced the number of features used for training. We hypothesize that the classifier

for Polish text might rely more on these less frequently occurring words or misspelled versions, while anecdotally, machine translation tends to reduce word variety in the text. In this context, higher threshold might disproportionately affect the classifiers built on the original text.

Unsurprisingly, both neural network classifiers using pre-trained language models outperformed other approaches. BERT uses dictionary of 30K words and embeddings with 768 dimensions. Due to the deep, 12-layer architecture, first 4-8 layers are able to better capture the semantic load of each word, which in turn provide better foundation for classification. As far as original-translated text classification performance is concerned, it seems that even though most of the original tools are developed within English-speaking paradigm, translating Polish text into English does not improve performance.

VII. Summary

Overall, we find no evidence that machine translation significantly improves performance of two popular models used for sentiment analysis, when applied to Polish reviews of restaurants. The performance either degrades (NB-classifier case) or is negligibly better (XGBoost case), while still not better than best performing model using original, non-translated text. As expected, in the absence of extensive feature engineering, the deep-learning approach performs best, although retaining the original review text yields slightly better performance. We also find that common preprocessing techniques, lemmatization and stop words removal, degrade classifier performance in sentiment analysis of Polish reviews.

References

[1] Wang, P., Berry, M.W. and Yang, Y. (2003), Mining longitudinal web queries: Trends and patterns. J. Am. Soc. Inf. Sci., 54: 743-758

[2] Casey Whitelaw, Ben Hutchinson, Grace Y Chung, and Ged Ellis. (2009). Using the Web for Language Independent Spellchecking and Autocorrection. In Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing: 890–899

[3] Praphula Kumar Jain, Rajendra Pamula, Gautam Srivastava (2021), A systematic literature review on machine learning applications for consumer sentiment analysis using online reviews, Computer Science Review, 41

[4] Pang B., Lee L. (2009) Opinion mining and sentiment analysis. Computational Linguistics, 35 (2): 311 – 312

[5] Kumar, Akshi & Sebastian, Teeja. (2012). Sentiment Analysis: A Perspective on its Past, Present and Future. International Journal of Intelligent Systems and Applications

[6] Ramina Khorsand, Majid Rafiee, Vahid Kayvanfar (2020), Insights into TripAdvisor's online reviews: The case of Tehran's hotels, Tourism Management Perspectives, 34.

[7] Zhang, W., Kong, Sx., Zhu, Yc. et al. (2019). Sentiment classification and computing for online reviews by a hybrid SVM and LSA based approach. Cluster Comput 22 (Suppl 5): 12619–12632

[8]

[9] Xu, X., Liu, W., & Gursoy, D. (2019). The Impacts of Service Failure and Recovery Efforts on Airline Customers' Emotions and Satisfaction. Journal of Travel Research, 58(6), 1034–1051.

[10] Jiaqi Luo, Songshan (Sam) Huang & Renwu Wang (2021) A fine-grained sentiment analysis of online guest reviews of economy hotels in China, Journal of Hospitality Marketing & Management, 30:1, 71-95

[11] F. L. Dos Santos and M. Ladeira, (2014). The Role of Text Pre-processing in Opinion Mining on a Social Media Language Dataset, Brazilian Conference on Intelligent Systems: 50-54

[12] Kavanagh, James, Greenhow, Keith and Jordanous, Anna (2023) Assessing the Effects of Lemmatisation and Spell Checking on Sentiment Analysis of Online Reviews. In: 17th IEEE International Conference on SEMANTIC COMPUTING (ICSC)

[13] https://github.com/uber/h3

[14] https://github.com/google/cld3

[15] https://github.com/explosion/spaCy

[16] Anna Rogers, Olga Kovaleva, and Anna Rumshisky. 2020. A Primer in BERTology: What We Know About How BERT Works. Transactions of the Association for Computational Linguistics, 8:842–866.

[17] Tianqi Chen, & Carlos Guestrin (2016). XGBoost. In Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. ACM

[18] Devlin, J., Chang, M.W., Lee, K., & Toutanova, K.. (2018). BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding.

[19] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A., Kaiser, L., & Polosukhin, I.. (2017). Attention Is All You Need.

[20] Mroczkowski, R., Rybak, P., Wróblewska, A., & Gawlik, I.. (2021). HerBERT: Efficiently Pretrained Transformer-based Language Model for Polish.

[21] Merchant, A., Rahimtoroghi, E., Pavlick, E., & Tenney, I. (2020). What Happens To BERT Embeddings During Fine-tuning?. In Proceedings of the Third BlackboxNLP Workshop on Analyzing and Interpreting Neural Networks for NLP (pp. 33–44). Association for Computational Linguistics.