Sentiment Analysis in Swedish text, using decoder-only generative large language models for Nordic languages

William Schill

Luleå University of Technology
977 54 Luleå, Sweden
wilsch-9@student.ltu.se

György Kovács
EISLAB Machine Learning
Luleå University of Technology
977 54 Luleå, Sweden
gyorgy.kovacs@ltu.se

Abstract

The exponential growth of online communication has led to a surge in the expression of negative and hateful sentiments, posing challenges in fostering inclusive environments. To meet these challenges, research has targeted the analysis of sentiment in text, and the detection of hateful/offensive content, mostly focusing on high-resource languages, like English. Here, we explore sentiment analysis, as a stepping stone towards detecting hateful and offensive content in Swedish. For this, we use the Svensk ABSAbank-Imm 1.1 dataset.

Using this dataset, we examined two deep learning models, namely a hybrid network of Convolutional and Long-Short Term Memory layers (CNN-LSTM model), and a pre-trained GPT model (GPT-SW3). To account for the class-imbalance of the corpus (i.e. moderate and neutral samples highly outnumbering highly negative and positive ones), both models were trained using a combination of class-balancing and automated weighting of samples. We evaluated the resulting models using several metrics (including F_1-score and Krippendorff's alpha).

Results show GPT-SW3 outperforms CNN-LSTM in all metrics. Furthermore, our experiments show that balancing the training data is beneficial for only certain evaluation metrics, and when using CNN-LSTM. Lastly, we find that in terms of Krippendorff's Alpha the relatively new GPT-SW3 model is outperformed by older, BERT- and RoBERTa-based models.

1 Introduction

The rise of digital platforms has transformed how individuals express their emotions and opinions, often leading to the proliferation of negative sentiments, including hate speech. Understanding these negative sentiments is important in many sensitive subjects, including that of immigration, where negative expressions can contribute to discrimination and societal division.

Despite the growing interest in Natural Language Processing (NLP) in general, and sentiment analysis in particular, there have been moderate efforts focusing on languages with limited resources (Birjali et al., 2021) (such as Swedish). Here, we aim to address this gap by examining the effectiveness of sentiment analysis techniques on Swedish text. During this work, our primary objective is to assess advanced NLP models in identifying negative sentiment directed at immigrants using the Svensk ABSAbank-Imm 1.1 dataset.

1.1 Related work

Several key studies have laid the foundation for sentiment analysis in Swedish. Yantseva and Kucher (2022) explored stance classification in social media texts, highlighting how right-wing Twitter users in Sweden often use neutral language strategically. Their work emphasizes the need to consider context and metadata in sentiment analysis. Similarly, Åkerlund (2020) examined political discourse on Twitter, showing that influential users propagate specific narratives that can influence public sentiment, stressing the importance of contextual factors. Baglini et al. (2021) advanced a lexicon-based approach, adapting the VADER tool for Scandinavian languages, revealing cross-linguistic sentiment differences. This lexicon-based method was further applied by Hammarlin et al. (2023) to analyze sentiments around COVID-19 vaccinations in Swedish social media, showcasing its broad applicability. While Sundström (2018) examined specifically the sentiment in reviews of various domains of products and services, and how the performance of different models transfer across the different domains.

Regarding model selection, we follow earlier works where the hybrid CNN-LSTM approach was shown to be successful (Wei et al., 2017; Kovács et al., 2022), while also building on the success of decoder-only models in classification (Liga and Robaldo, 2023).

Label	Partition			Overall
	Train	Validation	Test	
Very Positive	333	41	66	440
Positive	850	158	139	1147
Neutral	1800	214	228	2242
Negative	736	72	42	850
Very Negative	141	1	9	151

Table 1: Distribution of class labels in the ABSAbank-Imm 1.1 dataset

2 Data

The dataset used here (Svensk ABSAbank-Imm 1.1) is a subset of the larger Swedish ABSAbank corpus, with a focus on immigration-related text. For this, sources include editorials and opinion pieces from major Swedish newspapers (Svenska Dagbladet, Aftonbladet), and posts from Flashback (a popular Swedish forum). The dataset contains 4,872 paragraphs (a breakdown of which is summarized in Table 1), totaling approximately 199,000 words. Each paragraph is manually labeled with a sentiment score on a scale of 1 (very negative) to 5 (very positive) regarding immigration in Sweden (Berdicevskis, Aleksandrs et al., 2024). As can be seen in Table 1, these sentiment scores are represented to highly different degrees in the data.

3 Methods

Here, we examine two distinct architectures (CNN-LSTM and a fine-tuned GPT-SW3) chosen to represent an earlier deep learning approach, and novel, decoder-only transformers. We have placed both models in similar pipelines consisting of the following steps: 1) Pre-processing, 2) Re-sampling, 3) Model training.

3.1 Pre-processing

The following steps were used for both models:

1. Remove special characters (e.g. special symbols, punctuation marks)
2. Tokenization
3. Stopword removal
4. Stemming and lemmatization

For the CNN-LSTM model, to allow the efficient processing of batches, additional steps were necessary, such as padding sequences to ensure uniform input length.

3.2 Re-sampling

As shown in Section 2, the class distribution of ABSAbank is skewed. This imbalance can bias the model toward the majority class. To address this, various resampling methods (including undersampling, oversampling, and data augmentation) were examined. And as a baseline, we also trained without resampling.

The best resampling technique was selected for final model training and meta-parameter optimization based on the evaluation of performance on minority class predictions. Undersampling reduced training data and led to poorer overall performance, while oversampling caused overfitting, yielding only modest improvements. The hybrid approach, combining undersampling, oversampling, and data augmentation, produced the best results.

3.3 Deep Learning Models

CNN-LSTM: An investigation was conducted to explore a wide range of meta-parameter combinations on the training set. For this, we used a random search approach with meta-parameters drawn from the pre-defined ranges (see, Table 2). For each select parameter configuration, we trained five independent models, and chose the combination to use in later experiments based on various evaluation metrics measured on the validation set.

Description	Values
Size of batch during training	32, 64
Dimension for token embedding	64, 128, 256
Number of filters in 1st CNN layer	16, 32, 64
Number of filters in 2nd CNN layer	0, 16, 32, 64
Number of cells in the LSTM layer	64, 128, 256
Number of LSTM layers	1, 2, 3
Dropout rate	0.3, 0.4, 0.5
Learning rate	0.0005, 0.001, 0.002

Table 2: Potential meta-parameters of the CNN-LSTM architecture

GPT-SW3: We also fine-tuned the 1.3 billion parameter GPT-SW3 (Ekgren et al., 2023) for the task. To mitigate the risk of overfitting, a layer-freezing strategy was experimented with, where only certain layers were allowed to be updated during fine-tuning. By freezing certain layers, the model could retain previously learned representations while allowing for fine-tuning of the upper layers to adapt to the specific characteristics of the dataset. The optimization of meta-parameters also included learning rate, and batch size.

Experiment	Batch Size	EmbedDim	FilterNo1	FilterNo2	LSTMSize	LSTMLayers	Dropout	LR	Loss ↓	Accuracy ↑	F_1-Score ↑	MSE ↓	κ ↑
1	32	64	32	16	256	1	0.4	0.0005	1.455	0.303	0.309	1.403	0.12
2	16	256	64	32	128	3	0.3	0.0020	1.442	0.322	0.329	1.392	0.15
3	64	128	32	64	64	2	0.4	0.0005	1.498	0.279	0.294	1.418	0.13
4	64	128	16	0	64	2	0.5	0.0010	1.289	**0.353**	**0.362**	**1.284**	**0.21**
5	32	256	64	32	128	1	0.3	0.0020	1.351	0.331	0.342	1.322	0.18
6	32	64	16	0	256	3	0.4	0.0010	1.323	0.324	0.331	1.299	0.16
7	16	128	64	16	256	1	0.4	0.0010	1.399	0.314	0.321	1.352	0.16
8	64	256	32	0	128	2	0.5	0.0010	**1.287**	0.339	0.348	1.285	0.20
9	64	16	32	64	64	1	0.5	0.0005	1.511	0.271	0.282	1.454	0.11
10	16	128	16	32	128	2	0.3	0.0020	1.378	0.318	0.327	1.341	0.14

Table 3: Results from the CNN-LSTM experiments (results reported are average scores of five independently trained models) with various meta-parameter combinations on the validation set. In this table (and all subsequent tables), columns where higher values signify better performance are marked with ↑, while columns where lower values signify better performance are marked with ↓

4 Experiments and Results

In this section, we discuss the results of our experiments, evaluating the performance of the fine-tuned GPT-SW3 model and the CNN-LSTM for sentiment analysis on Swedish text. First, in Section 4.1 we list and briefly discuss the various evaluation metrics used for evaluating the models. Then, in sections 4.2 and 4.3, respectively we share the results of our experiments using CNN-LSTM and GPT-SW3 models. Lastly, in Section 4.4 we compare our results to those found in the literature for the same dataset.

4.1 Evaluation metrics

We used five metrics for performance analysis, corresponding to different perspectives from which the task at hand can be approached.

For example, one can consider sentiment analysis as a classification task. Corresponding to this, we examined measures primarily used for classification. Namely, **accuracy** and to also take into account the imbalance present in the dataset, F_1**-score**. One could also look at the prediction of sentiment labels in the ABSAbank-IMM 1.1 dataset as the prediction of values from 1 (very negative), to 5 (very positive), transforming the task into one of regression. Because of this, we also examined a common metric used for regression, the **Mean Squared Error (MSE)**.

Lastly, following common practice in the literature (Provoost et al., 2019), we examine **Cohen's Weighted kappa** (Cohen, 1968; Ben-David, 2008) (later referenced as **Weighted** κ or κ, for brevity), and **Krippendorff's Alpha** (Krippendorff, 2018; Saura et al., 2019). Two metrics that assess the agreement between model predictions and human judgments, ensuring consistency in sentiment classification.

4.2 CNN-LSTM

Results of our experiments conducted for the examination meta-parameters are shown in Table 3. As Table 3 shows, the model in Experiment 4 attained the best performance according to all but one metric, where it was competitive with the more complex model from Experiment 8. Based on these results (and on model complexity), we selected Experiment 4 for further analysis.

The CNN-LSTM model achieved an **accuracy** of 0.329 and an $\mathbf{F_1}$**-Score** of 0.329 (see, Table 4), indicating a moderate ability to classify sentiment labels correctly. The **MSE** for this model was recorded at 1.298, reflecting the average squared difference between predicted and actual sentiment scores. Additionally, the **Weighted** κ was measured at 0.321, suggesting that the model's predictions were somewhat distant from the true sentiment classes. The **Krippendorff's Alpha** value of 0.255 indicates a low level of agreement between the model's predictions and human judgments, highlighting potential areas for improvement in capturing sentiment nuances.

When trained without using re-sampling, the CNN-LSTM model shows improved performance if measured according to **Accuracy** and **MSE** - two metrics that do not take into account the class imbalance. For the other measures, however, the performance of the CNN-LSTM model considerably decreased without re-sampling. The decreased κ and α suggesting a further decrease in agreement with human sentiment assessment.

Model	Resampling	Accuracy ↑	F_1-Score ↑	MSE ↓	κ ↑	α ↑
CNN-LSTM	Yes	0.329	**0.329**	1.298	**0.321**	**0.255**
CNN-LSTM	No	**0.424**	0.297	**1.059**	0.268	0.215

Table 4: Test scores attained using CNN-LSTM on the test set (results reported are the average of 5 independently trained models)

Last N Layers Trained	Learning Rate	Loss ↓	Accuracy ↑	F_1-Score ↑	MSE ↓	κ ↑
All	0.000020	1.920	0.312	0.321	1.593	0.236
1	0.000010	1.354	0.514	0.530	0.921	0.381
2	0.000010	**1.339**	**0.529**	**0.541**	**0.881**	**0.397**
4	0.000020	1.410	0.482	0.494	1.054	0.341
6	0.000005	1.466	0.418	0.437	1.228	0.296

Table 5: Training experiments for fine-tuning the GPT-SW3 model (results reported are the average of five independently trained models on the validation set)

4.3 GPT-SW3

Results of our experiments for the optimization of the GPT model's meta-parameters are shown in Table 5. As can be seen in the table, the configuration where the last 2 layers are trained attained consistently the best performance over all metrics. Thus we evaluated the performance of GPT-SW3 on the test set using this configuration.

Results attained by GPT-SW3 are listed in Table 6. As can be seen, the Fine-Tuned GPT-SW3 achieved an **accuracy** of 0.448 and an **F_1-Score** of 0.452, indicating a strong capability in correctly classifying sentiment labels. The **MSE** for this model was 0.985, reflecting a relatively low average squared difference between predicted and actual sentiment scores. Additionally, the **Weighted κ** was measured at 0.462, suggesting that while the model performed well overall, there were still some misclassifications affecting its predictions. The **Krippendorff's Alpha** value of 0.379 indicates a moderate level of agreement between the model's predictions and human judgments, highlighting its effectiveness in capturing sentiment nuances.

Resampling	Accuracy ↑	F_1-Score ↑	MSE ↓	κ ↑	α ↑
Yes	0.448	0.452	0.985	0.462	0.379
No	**0.508**	**0.501**	**0.836**	**0.481**	**0.424**

Table 6: Test scores attained using fined-tuned GPT-SW3 on the test set (results reported are the average of 5 independently fine-tuned models)

4.4 Benchmarking

The Fine-Tuned GPT-SW3 and CNN-LSTM models are benchmarked against a range of models on the same data[1], using **Krippendorff's Alpha** (see, Table 7). Table 7 shows that the Fine-Tuned GPT-SW3 model outperforms the CNN-LSTM. However, it is important to note that the Fine-Tuned GPT-SW3 model requires markedly more energy due to its larger size and complexity. One can also see that despite being one of the newest models, the GPT-SW3 performs relatively poorly compared to other large transformers. This discrepancy raises questions regarding the factors contributing to the underperformance of GPT.

[1]we consider models that distinguish five levels of sentiment, thus excluding the work of for example, Hägglöf (2023)

Model	α
KB/bert-base-swedish-cased	0.529
xlm-roberta-large	0.516
KBLab/megatron-bert-large-swedish-cased-165k	0.508
AI-Nordics/bert-large-swedish-cased	0.480
KBLab/megatron-bert-base-swedish-cased-600k	0.449
KBLab/bert-base-swedish-cased-new	0.428
Fine-Tuned GPT-SW3 (No Resampling)	0.424
NbAiLab/nb-bert-base	0.390
Fine-Tuned GPT-SW3	0.379
xlm-roberta-base	0.366
SVM	0.286
CNN-LSTM	0.255
CNN-LSTM (No Resampling)	0.215
Decision Tree	0.117
Random	0.008
Random Forest	0.005
MaxFreq/Avg	-0.052

Table 7: Comparison of the examined and SotA models (Språkbanken Text, 2024) on the test set, using Krippendorff's Alpha (α)

5 Conclusions and Future Work

Between GPT-SW3, and CNN-LSTM, the latter achieves lower scores on all metrics. It is still important, however, to take into account the increased energy consumption of GPT when making a decision. Regarding the relative low Krippendorff's alpha values attained by GPT-SW3, in comparison with other transformers, one potential explanation could be overfitting. Another possibility is variations in the training process, (e.g. dataset composition, meta-parameter tuning), which may have influenced the performance of GPT-SW3. Addressing these issues requires further investigation. Alternative training methodologies, regularization techniques, or architectural adjustments may help mitigate overfitting and improve generalization. Moreover, refining the training process to better align with the model's architecture and objectives could lead to enhanced performance.

Through this work, the need for further advancements in fine-tuning multilingual models for sentiment analysis tasks in Swedish has been identified. Specifically, exploring ensemble learning approaches with diverse architectures or using multilingual models could better capture sentiment in text that combines Swedish with other languages.

References

Mathilda Åkerlund. 2020. The importance of influential users in (re)producing Swedish far-right discourse on twitter. *European Journal of Communication*, 35(6):613–628.

Rebekah Brita Baglini, Lasse Hansen, Kenneth Christian Enevoldsen, and Kristoffer Laigaard Nielbo. 2021. Multilingual sentiment normalization for scandinavian languages. *Scandinavian Studies in Language*, 12:50–64.

Arie Ben-David. 2008. Comparison of classification accuracy using cohen's weighted kappa. *Expert Systems with Applications*, 34(2):825–832.

Berdicevskis, Aleksandrs, Borin, Lars, Rouces, Jacobo, and Tahmasebi, Nina. 2024. Svensk absabank-imm 1.1. https://spraakbanken.gu.se/resurser/absabank-imm.

Marouane Birjali, Mohammed Kasri, and Abderrahim Beni-Hssane. 2021. A comprehensive survey on sentiment analysis: Approaches, challenges and trends. *Knowledge-Based Systems*, 226:107134.

Jacob Cohen. 1968. Weighted kappa: Nominal scale agreement provision for scaled disagreement or partial credit. *Psychological bulletin*, 70(4):213.

Ariel Ekgren, Amaru Cuba Gyllensten, Felix Stollenwerk, Joey Öhman, Tim Isbister, Evangelia Gogoulou, Fredrik Carlsson, Alice Heiman, Judit Casademont, and Magnus Sahlgren. 2023. Gpt-sw3: An autoregressive language model for the nordic languages. *Preprint*, arXiv:2305.12987.

Hillevi Hägglöf. 2023. The kblab blog: A robust, multi-label sentiment classifier for swedish. https://kb-labb.github.io/posts/2023-06-16-a-robust-multi-label-sentiment-classifier-for-swedish/.

Mia-Marie Hammarlin, Dimitrios Kokkinakis, and Lars Borin. 2023. Covid-19 vaccine hesitancy. *Journal of Digital Social Research*, 5:31–61.

György Kovács, Pedro Alonso, Rajkumar Saini, and Marcus Liwicki. 2022. Leveraging external resources for offensive content detection in social media. *AI Communications*, 35(2):87–109.

Klaus Krippendorff. 2018. *Content analysis: An introduction to its methodology*. Sage publications.

Davide Liga and Livio Robaldo. 2023. Fine-tuning gpt-3 for legal rule classification. *Computer Law & Security Review*, 51:105864.

Simon Provoost, Jeroen Ruwaard, Ward van Breda, and Tibor Bosse. 2019. Validating automated sentiment analysis of online cognitive behavioral therapy patient texts: An exploratory study. *Frontiers in Psychology*, 10.

José Saura, Ana Reyes-Menendez, and Pedro Palos-Sanchez. 2019. Are black friday deals worth it? mining twitter users' sentiment and behavior response. *Journal of Open Innovation: Technology, Market, and Complexity*, 5:58.

Språkbanken Text. 2024. Superlim 2. https://spraakbanken.gu.se/resurser/superlim.

Johan Sundström. 2018. Sentiment analysis of Swedish reviews and transfer learning using Convolutional Neural Networks. Master's thesis, Uppsala University.

Xiaocong Wei, Hongfei Lin, Liang Yang, and Yuhai Yu. 2017. A convolution-lstm-based deep neural network for cross-domain mooc forum post classification. *Information*, 8:92.

Victoria Yantseva and Kostiantyn Kucher. 2022. Stance classification of social media texts for under-resourced scenarios in social sciences. *Data*, 7(11):159.