Improving Data Balance in Sentiment Analysis through Diverse Example Downsampling and Synthetic Data Generation

Fredrik Andersen Langsem

Thesis submitted for the degree of
Master in Applied Computer and Information Technology (ACIT)
- Phase X
30 credits

Department of Computer Science
Faculty of Technology, Art and Design

OSLO METROPOLITAN UNIVERSITY

Spring 2025

Improving Data Balance in Sentiment Analysis through Diverse Example Downsampling and Synthetic Data Generation

Fredrik Andersen Langsem

Improving Data Balance in Sentiment Analysis through Diverse Example Downsampling and Synthetic Data Generation

http://www.oslomet.no/

Printed: Oslo Metropolitan University

Abstract

One of the main challenges in sentiment analysis is class imbalance where models often lean heavily toward the more common sentiment and miss the nuances in less frequent cases. In this thesis, we explore whether downsampling and synthetic data generation approaches can help build more balanced and useful data for binary sentiment classifiers.The study uses four million Amazon reviews (an even split between positive and negative), testing various resampling methods across different dataset sizes and imbalance levels. After anonymizing the reviews with named entity recognition and regular expressions, we transformed the text into BERT embeddings for classification. We compared several undersampling methods (like K-Means and NearMiss) and oversampling approaches (including ADASYN and contextual augmentation). Using accuracy, precision, recall, and F1-score to evaluate performance, the results show that augmentation, especially the contextual method, tends to offer more reliable gains, especially as the imbalance grows. These insights could help guide method selection in future sentiment analysis tasks.

Acknowledgments

I would like to express my thanks to the people who assisted me while this thesis was in progress. First, I acknowledge my supervisor, Professor Gustavo Borges Moreno e Mello for clear feedback, and timely reminders that helped keep the project on schedule. I also thank the co-supervisors(not my co-supervisors), Trym Adrian Eidsvik Lindell and Fabrizio Palumbo, who attended the weekly meetings and genuinely paid attention to what we presented during our meetings. They brought up several interesting aspects that I don't think I would have stumbled upon myself. An extra thanks goes out to Hedda Marie Westlin who facilitated the Virtual computer that allowed me to do this thesis at the scale I wanted it to be. Thank you for all the help from my way too many questions. My family provided constant encouragement and understanding during the longest hours of these months. A special thanks goes to my mother who have suffered through hours of me yapping on about the project like she was my rubber duck. Finally, I acknowledge the wider open-source community. Libraries such as PyTorch, scikit-learn, spaCy, and Hugging Face Transformers formed the backbone of the experimental pipeline, and the public Amazon review dataset from Kaggle for the raw material.

List of Figures

List of Tables

Contents

Chapter 1

Introduction

Sentiment analysis projects sometimes include some sentiment categories that are represented by less instances than others, which causes major problems in machine learning because of dataset imbalance. Regularly, conventional classification models naturally favor these more regularly occurring, majority sentiment groups, which usually leads to biased predictions and an overall decline in the model's capacity to successfully generalize. Such imbalance creates a significant danger of the model failing to properly capture modest differences in emotion, especially when exact categorization depends on spotting small language subtleties. Therefore, it is all the more important to include efficient techniques meant to balance datasets, in so attempting to guaranteeing a fair representation of all sentiment categories and improving the general dependability, accuracy, and fairness of the classification model.

Inspired by and building on the work shown in *Optimal AI through Minimal Data: Enhancing Sentiment Analysis with Data Diversity for Norwegian* [4], the present study investigates undersampling techniques as a way to attain improved class balance in sentiment analysis. Although the previous studies focused on sentiment analysis within a dataset with five different sentiment categories, this thesis modifies and expands these approaches particularly for a binary sentiment classification situation. The study looks at how Diverse Example Downsampling and Synthetic Data Generation affect a two-class sentiment classification. Particularly the requirement to maintain representative and evenly distributed emotion expressions post-balancing, the move from a multi-class to a binary classification system presents special difficulties.

This thesis's main goal is to assess how sentiment analysis performance is affected by Diverse Example Downsampling and Synthetic Data Generation techniques.

By means of methodical comparisons of different sampling strategies, we hope to determine the effect these techniques have on classification accuracy and their ability to reduce problems connected to class imbalance. Specifically, the study looks at how well K-Means, NearMiss, ENN, and DBSCAN undersampling strategies preserve diversity in the majority class while also lowering its frequency. At the same time, oversampling methods including ADASYN and generative text augmentation are investigated to enhance the minority class with linguistically consistent artificially produced instances.

Real-world situations with sentiment analysis-driven decision-making processes, such as customer feedback analysis, social media interaction monitoring, and brand reputation management, show practical significance in the results of this study. Generating actionable insights in these situations calls for precise sentiment models able to properly identify and classify both positive and negative views. Furthermore, knowledge of the trade-offs between different sampling techniques will offer data scientists and machine learning practitioners useful direction when dealing with imbalanced datasets in natural language processing (NLP) initiatives.

This study adds significantly to continuous debates and advances in data balancing techniques in NLP by tackling the ongoing problem class imbalance in sentiment analysis assignments. Emphasizing the possible advantages of using Diverse Example Downsampling and Synthetic Data Generation methods, the results seek to hone and enhance best practices for managing imbalanced datasets. In the end, the findings of the study will help to improve the general efficacy, dependability, and strength of sentiment classification systems.

Chapter 2

Literature review

A common problem in sentiment analysis is class imbalance, in which some sentiment classes, like unfavorable reviews, are substantially underrepresented relative to others. This mismatch can lead to biased classifiers that favor the majority class, resulting in poor recall for minority sentiments and ultimately distorted sentiment predictions [10]. Different downsampling and upsampling methods have been created to balance datasets and enhance model performance, hence minimizing this problem.

Research on class imbalance spans for decades, straddling machine learning, data mining, and natural-language processing. Early work revolved around algorithm-level tricks such as cost-sensitive losses, but by the mid-1990s attention had shifted to data-level remedies—deleting surplus majority examples or fabricating synthetic minority ones. Because the present thesis centres on data-level strategies, the survey below tracks that lineage in detail, highlighting how ideas first forged for low-dimensional numeric tasks have been re-interpreted for high-dimensional contextual text.

2.1 Down-sampling

Down-sampling is, as the term suggests, producing a smaller dataset with samples from the original dataset. Though there are many other ways to accomplish this in this sector, K-means down-sampling is the most common. Just because k-means is the most prevalent doesn't mean we are not going to only look at it. There are advantages and disadvantages with down-sampling data. The biggest advantage is that down-sampling the majority class will decrease the potential of overfitting on the

majority class, in short, it will reduce the false positives/negatives predicted for the majority class. But this advantage does not come free, as when it is down-sampled, it has less information to work with, meaning it can be highly biased, meaning it can be extremely good at the dataset it is trained on, but when you introduce data it have not seen before, it is performing horrible.

2.1.1 K-Means

K-Means Undersampling (KM-US) is a cluster-based undersampling strategy that minimizes the majority class size while keeping its diversity. Representative samples are chosen from each cluster using the K-Means algorithm, which divides the majority class into K, the amount of minority instances, amounts of clusters. This approach attempts to ensures that the dataset maintains its semantic structure and stops the loss of notable linguistic diversity. By maintaining cluster-level diversity, K-Means undersampling has been demonstrated to enhance performance on imbalanced text datasets [16], and this effect can carry over to sentiment assignments.

2.1.2 Near Miss

NearMiss Undersampling (NM-US) is a distance-based technique that selects majority class samples closest to the minority class, refining decision boundaries [17]. Variations such as NearMiss-1 (choosing majority samples closest to minority occurrences) and NearMiss-2 (choosing those distant from other majority samples) assist balance datasets while minimizing classification bias [5]. In sentiment analysis projects with very overlapping sentiment expressions, NM-US is especially successful. A third variation, NearMiss-3, chooses majority class samples closest to three nearest minority class instances [17]. Unlike NearMiss-1 and NearMiss-2, which stress personal distance measurements, NearMiss-3 guarantees that every majority class instance kept in the dataset is evenly spread across several minority samples, in an attempt of producing a more balanced representation. When the minority class has different emotional expressions, this approach is very successful since it guarantees that kept majority samples fit well with different sentiment patterns.

2.1.3 Tome Links

Tomek Links Undersampling (TL-US) is a data-cleaning method that removes ambiguous majority class samples that are nearest neighbors to minority class instances [13]. These Tomek Links often indicate class overlap, and removing them enhances decision boundaries, reducing misclassification [5]. TL-US has been applied in sentiment analysis to improve classification performance by refining the dataset and eliminating noisy examples.

2.1.4 Edited Nearest Neighbors (ENN)

As an undersampling method, Edited Nearest Neighbors (ENN) seeks to remove samples that can be confusing or noisy in order to improve the quality of the dataset. ENN inspects its k-nearest neighbors, for every dataset instance. The instance is deemed probably mislabeled or non-representative if most of the neighbors have a different class label than the instance itself; it is therefore deleted from the training set [14].

In imbalanced situations where the majority class may overwhelm the feature space and provide borderline or noisy samples that mislead the classifier, ENN is very beneficial. Cutting down these occurrences helps ENN to define class boundaries, and hence cleaning and stabilizing the decision areas.

In sentiment analysis, this can be especially helpful when dealing with ambiguous reviews, for example, sarcastic or neutral-toned texts that may not strongly reflect their labeled sentiment. ENN has been used in combination with other approaches such as Tomek Links to construct more aggressive noise-removal pipelines [5]. Applying ENN on sparse or high-dimensional data, such text embeddings, where class boundaries may not be well-defined in Euclidean space, can be too harsh, which is one drawback. Therefore, while ENN can enhance classifier robustness, it should be applied cautiously, especially when the minority class is already small.

2.1.5 DBSCAN

Density-Based Spatial Clustering of Applications with Noise (DBSCAN) Undersampling is a clustering-based technique that removes majority class samples based on their density distribution [8]. Unlike K-Means, which requires specifying the number of clusters in advance, DBSCAN identifies dense regions of data and classifies

points as either core points, border points, or noise based on their density. For undersampling, DBSCAN removes majority class noise points that do not belong to any dense cluster while retaining representative samples from denser areas. This ensures that redundant samples from the majority class are eliminated while preserving the most informative ones. In sentiment analysis, DBSCAN-based undersampling has been found useful in reducing highly redundant samples from the majority sentiment while maintaining diverse expressions that are essential for classification [1]. DBSCAN Undersampling is particularly beneficial when working with highly skewed datasets, as it effectively removes outliers and noise from the majority class without affecting minority class distributions. However, it requires careful tuning of its epsilon parameter, which defines the neighborhood size, to prevent excessive removal of valuable data points.

2.2 Up-sampling

2.2.1 Augmentation

An easy solution to get as many minority entries as there are entries in the majority class is to just copy the minority entries over and over until you have the same number of entries. Technically, you have a balanced dataset containing only «true» data. But you are now extremely prone to overfitting due to the lack of variance in the once minority class. To avoid this we have a few different approaches to up sample without having the exact same data being repeated.

2.2.2 Synonym replacement

Instead of just duplicating an entry one can take an entry and replace certain words with a synonym, thus creating a unique entry(hopefully). The meaning overall meaning remaining while expanding the data variance. Eg. «Have a god day» can turn into «Have a great day», «Have a great day», «Have a wonderful day», «Have a acceptable day». Sure the last one isn't something I think I have ever heard, but the meaning is the same «Have a [positive] day».

2.2.3 Back Translation Augmentation

Back translation is a text augmentation technique that involves translating a sentence into another language and then back to the language it was translated from.[7]. This process generates paraphrased versions of minority class samples while preserving sentiment. Studies have demonstrated that back translation can improve sentiment analysis models by diversifying the training data [15].

2.2.4 Contextual Augmentation (BERT Masking)

Contextual augmentation uses language models that have already been trained, like BERT, to guess and change words in a sentence, creating new versions while keeping close to the original meaning [3]. This approach has been implemented in sentiment classification to enhance data balance without introducing too much label noise, rendering it a viable substitute for conventional oversampling methods.

2.2.5 SMOTE

Synthetic Minority Over-sampling Technique (SMOTE) is a widely used resampling method that generates synthetic samples to balance imbalanced datasets [6]. Unlike random oversampling, which duplicates existing minority class instances, SMOTE creates new synthetic examples by interpolating between existing samples. It picks a random minority class instance, finds its k-nearest neighbors, then creates new points across the vector space separating them. This method reduces overfitting brought on by straightforward duplication even as it increases the variety of the minority class. SMOTE is very beneficial in sentiment analysis for datasets where negative or neutral emotions are underrepresented. Models trained on SMOTE-boosted data gain more generalizable decision limits by increasing the feature space using synthetically produced texts. Nevertheless, SMOTE may introduce synthetic noise when applied to high-dimensional text embeddings. Therefore, it is most effective when combined with data cleansing techniques such as Edited Nearest Neighbors (ENN) or Tomek Links [5], to reduce some the of noise it created.

2.2.6 ADASYN

Adaptive Synthetic Sampling (ADASYN) is an oversampling technique designed to address class imbalance by focusing on hard-to-learn minority class instances [11]. The sampling procedure is modified by ADASYN according to the density distribution of the minority class, in contrast to SMOTE, which creates synthetic samples consistently. In regions where the minority class is sparsely distributed, the model is able to learn from more challenging examples while avoiding excessive duplication of readily classified instances by generating a greater number of synthetic samples.

When confronted with sentiment expressions that are underrepresented in sentiment analysis, ADASYN is particularly advantageous, as they are more challenging for models to accurately classify. By producing synthetic samples in these problematic regions, ADASYN boosts model robustness and helps capture nuanced sentiment fluctuations [12]. ADASYN might generate more noise than SMOTE if the generated samples do not closely match genuine language patterns since it is adaptive. Usually ADASYN is used together with data cleaning techniques like Edited Nearest Neighbors (ENN) or Tomek Links, to help remove remove erroneous synthetic examples and improve dataset quality.

2.3 Combined Under- and Oversampling Techniques

Recent studies have looked at how well combining undersampling and oversampling methods produces a more balanced and varied dataset for machine learning applications. Hybrid approaches try to exploit the advantages of both methods by reducing duplication in the majority class while supplementing the minority class with useful synthetic samples [9].

A common strategy is to first apply an undersampling method such as Tomek Links or NearMiss to remove noisy or redundant majority class samples, followed by an oversampling technique like SMOTE to generate synthetic instances for the minority class. This method guarantees that the synthetic samples are produced in a less biased and cleaner feature space, hence improving model generalization and resilience. Research in sentiment analysis has demonstrated that hybrid resampling methods can greatly enhance classification performance. For example, ENN (Edited Nearest Neighbors) helps eliminate false synthetic samples (first upsample then downsample), hence SMOTE paired with ENN showed greater F1-scores than

using SMOTE alone, according to [2].Similarly, [16] discovered that clustering-based undersampling, when associated with synthetic text creation, enhances the balance of linguistic variety in NLP tasks, minimizing overfitting in deep learning models.

Though these techniques provide notable gains, they need precise calibration to prevent problems like overgeneralization from too many synthetic examples or loss of important information from too much undersampling.

Chapter 3

Methods

This chapter outlines the methodology used to investigate the effects of Diverse Example Downsampling and Synthetic Data Generation on sentiment analysis tasks. The study follows a structured approach, including dataset selection, preprocessing, application of sampling techniques, and model evaluation.

3.1 Dataset

The data used in this research comes from a dataset available on Kaggle.com. The dataset contains a total of 4 million reviews, evenly split between 2 million positive and 2 million negative reviews. The data is structured into two columns: one column contains an integer representing the sentiment label (1 for positive, 0 for negative), and the other column contains the textual review. By design, the review column does not contain any identifying information about the reviewer, but some users have included personal details within their reviews.

3.1.1 Anonymization Using Named Entity Recognition (NER) and Regular Expressions

To ensure anonymization and mitigate potential bias, the dataset is preprocessed using Named Entity Recognition (NER) and Regular Expressions (Regex) to remove personal information such as names, addresses, organizations, phone numbers, and email addresses. Named Entity Recognition (NER) is applied using SpaCy's en_core_web_sm model to identify and replace instances of personal data. Names are replaced with [Name], addresses with [Address], and organizations with

[Organization]. This process ensures that identifying information is removed while preserving the original sentence structure. Since some entities like phone numbers and email addresses follow predictable patterns, regex-based filtering is used to replace them. Email addresses are replaced with [MailAddress] using a regular expression pattern that detects most common email formats. Phone numbers, which can vary in format, are replaced with [PhoneNumber] using a regex pattern designed to capture international and local number formats. Removing names and organizations is not only for privacy but also to reduce bias. If a company is frequently mentioned in positive reviews, a model may learn an unintended association between the company name and sentiment. By anonymizing organizations as [Organization], the model is prevented from making biased predictions based on company names.

3.1.2 Splittting the data into smaller datasets

One size fits all. As a rather tall man, that is the most idiotic thing I have ever heard. So to not be too much of a hypocrite, I don't want to just find that one method works best for "my" dataset. I want to find out what methods work best for datasets of different sizes and of different degrees of imbalance. In order to do so, I have split the dataset into multiple datasets. Since my full dataset is balanced, I have split the dataset into 1 containing only positive reviews, 1.6 million, and the other containing only negative reviews, also 1.6 million. The remaining 20% of the data is kept as test data.

From here we have 3 categories. The first is datasets combined to make up 10 000 reviews, the second is datasets combined to contain 100 000 reviews, and lastly one category containing 1 000 000 reviews. Each category contains 10 datasets. The datasets are structured like this:

5% positive 95% negative

15% positive 85% negative

25% positive 75% negative

35% positive 65% negative

45% positive 55% negative

55% positive 45% negative

65% positive 35% negative

75% positive 25% negative

85% positive 15% negative

95% positive 5% negative.

Where all the positive and negative data is taken from the top of each list. To clarify here is an example:

- Dataset 5% positive 95% negative in the category where the datasets contain 10 000 reviews has the 500 first good reviews and the 9 500 first negative reviews
- Dataset 15% positive 85% negative in the category where the datasets contain 10 000 reviews has the 1 500 first good reviews and the 8 500 first negative reviews. Where the 500 positive reviews from the 5% positive 95% negative datasets are included.
- Dataset 15% positive 85% negative in the category where the datasets contain 100 000 reviews has the 15 000 first good reviews and the 85 000 first negative reviews. Where all the reviews from 10 000 categories are included.

The same logic is applied to the data used for testing.

3.1.3 Vectorizing

To prepare textual data for machine learning models, each review is converted into dense numerical representations using BERT embeddings. BERT (Bidirectional Encoder Representations from Transformers) is a pre-trained language model developed by Devlin et al. (2019), designed to understand the context of a word based on its surrounding words in a sentence.

Unlike traditional methods such as TF-IDF, which gives words a value based on their frequency over all entries, BERT captures rich contextual and semantic information by processing the entire sentence bidirectionally. Each review is passed through the pre-trained BERT model, and the resulting embeddings, typically taken from the [CLS] token or pooled output, as input features for downstream tasks.

These embeddings are not only more expressive but also well-suited for clustering-based downsampling techniques and data augmentation strategies, as they retain contextual cues that are essential for sentiment classification. BERT has a max length of 512 tokens, so we used that as the max length for each row in the dataset. The rows that had didn't have text that was 512 tokens long was 0 padded, and the rows exceeded 512 tokens were trunkated.

3.1.4 Augmentation implementation

K-means

This method applies the K-Means clustering algorithm to the majority class data using the sklearn.cluster.KMeans implementation. The number of clusters is set to match the number of minority class instances. For each cluster, the sample closest to the centroid is selected to ensure that representative and diverse samples are retained. This approach allows for systematic reduction of the majority class while preserving important structural patterns in the data. Though it was the best performing downsampling implementation in [4], I dont think it will be performing as well as the NearMiss methods, as I belive that ignoring the minority when cutting down the majority is almost like flailing around with a knife and hope you didnt cut anything vital.

Near miss 1, 2 and 3

NearMiss is implemented using the imblearn.under_sampling.NearMiss method. All three versions of NearMiss are tested to understand how different distance-based strategies affect the classification outcome. Version 1 selects majority class samples whose average distance to the three closest minority class instances is the smallest. Version 2, in contrast, selects those whose average distance to the three farthest minority class samples is smallest, thereby choosing examples near the outskirts of the minority distribution. Version 3 focuses on selecting majority class samples that are farthest from the minority class samples on average, placing emphasis on contrasting class separation. For all three versions, Euclidean distance is used, and n_neighbors is set to 5. The use of all NearMiss variants allows for a comprehensive examination of how boundary refinement strategies influence model learning. In all of the other downsampling metohds, this is the only one to take the minority into consideration when cutting down the majority, which i believe will show a positive

result, as it takes a look at the bigger picture.

DBSCAN

DBSCAN Undersampling is based on density-based spatial clustering, using sklearn.cluster.DBSCAN. The method is used to identify core, border, and noise points within the majority class. Majority samples that fall into sparse or noisy regions are discarded, while core points within dense clusters are retained.

To determine the optimal eps parameter, a custom function choose_eps is used. This function first computes the distance from each majority class sample to its n'th nearest neighbor (with n_neighbors=5). It then calculates the eps value by taking the 90th percentile of these distances. This approach ensures that the eps value adapts to the overall density of the dataset, capturing densely packed regions while excluding noise. The use of a percentile-based cutoff provides robustness to outliers and makes the method scalable to datasets of varying size and dimensionality. Ideally, we could set a default value here, but upon attempting to find a good value that fit them all, I couldn't find a value that worked on them all. Just like with the K-means I think the NearMiss methods will mostly outperform this method as well, but I think it will outperform the K-means, due to the wide variety a language offers.

ADASYN

This method implements the Adaptive Synthetic Sampling (ADASYN) algorithm via a custom GPU-accelerated class (TorchADASYN). The procedure begins by identifying the minority class and, for each of its instances, computing the k=5 nearest neighbours in the full dataset using RAPIDS cuML's NearestNeighbors (Euclidean distance) on the GPU. A difficulty ratio r_i is then calculated for each minority sample as the proportion of its k neighbours that belong to the majority class; these ratios are normalized into a probability distribution $\hat{r}_i$. The total number of synthetic samples GG is set automatically to balance the class counts, and each minority instance is allocated $g_i = \lfloor r^i G \rfloor$ new points, with any remainder assigned to the samples with highest $\hat{r}_i$. Synthetic examples are generated by interpolating through PyTorch operations on the 'cuda' device between each minority instance and randomly chosen neighbours, yielding X_{syn} that adaptively concentrate on hard-to-learn regions.

Unlike the undersampling methods, ADASYN preserves all original data and focuses sampling effort where the minority class is most vulnerable. I anticipate that this

adaptive oversampling will enhance classifier performance in boundary regions and outperform both naïve oversampling and data-discarding undersampling approaches by maintaining structural information and emphasizing difficult instances.

Back Translation Augmentation

This method uses the deep-translator Python package to translate each minority class review to Japanese and back to English. To reduce semantic drift, only translations with high cosine similarity to the original text (measured using Sentence-BERT) are retained. I have high hopes for this method to be working, even on the datasets that are heavily unbalanced. Part of the reason for my high hopes is that Japanese is supposed to be one of the hardest languages for native English speakers to learn, so I hope it translates.

Contextual Augmentation

We use a masked language modeling approach with HuggingFace's transformers library. In this process, tokens in minority class reviews are randomly selected for masking with a 15% probability. The pre-trained BERT model then predicts suitable replacements for the masked tokens based on the surrounding context. Once replacements are made, the new sentence is evaluated using sentence embeddings. If the cosine similarity between the original and augmented sentence embeddings is at least 0.85, the augmented version is retained and added to the training data. I expect this one to outperform the back translation datases on the datasets that are only lightly unbalanced, and then be outperformed by back translations when it has to redo many sentences it has already augmented, as i fear it will often unintentionally make a copy of the original entry when it operates on a generated entry with replacemnt.

3.2 Classification Model

A Long-term Short-Memory (LSTM) neural network architecture models sequential patterns in the review text for sentiment classification. Designed to be trained on tokenized BERT-like embeddings, this model has a structure optimized for capturing long-range dependencies and contextual cues. The model is the same as the one that we previously used for the whole dataset (4 million entries, including the

test entries), where it got an accuracy of 93%. It is not a carbon copy as I added an attention layer and used embedings from Tensorflow.Keras instead of bert, thus the input dimentionality and vocabulary is increased.

Architecture Overview

The model is implemented using Keras and is defined as follows:

```
def build_sequential_model(vocab_size=30000, max_len=512, learning_rate=2e-5):
    model = Sequential()
    model.add(Embedding(input_dim=vocab_size, output_dim=128, input_length=max_len))
    model.add(LSTM(units=128, return_sequences=True))
    model.add(Attention())
    model.add(Dense(units=64, activation='relu'))
    model.add(Dropout(rate=0.5))
    model.add(Dense(units=64, activation='relu'))
    model.add(Dropout(rate=0.5))
    model.add(Dense(units=64, activation='relu'))
    model.add(Dropout(rate=0.5))
    model.add(Dense(units=16, activation='relu'))
    model.add(Dropout(rate=0.5))
    model.add(Dense(units=1, activation='sigmoid'))

    optimizer = Adam(learning_rate=learning_rate)
    model.compile(optimizer=optimizer,
                  loss='binary_crossentropy',
                  metrics=['accuracy', tf.keras.metrics.Precision(), tf.keras.metrics.Recall()])
```

Figure 3.1: LSTM model

Component Breakdown

Embedding Layer: Transforms integer token sequences into 128-dimensional vector representations.

LSTM layer: Captures long-term dependencies and sequential relationships in the text.

Attention Layer: Increases focus on important parts of the sequence to improve sentiment prediction.

Dense and Dropout Layers: Three fully linked layers with ReLU activations and dropout to boost generalization.

Final Output: A sigmoid-activated neuron for binary classification.

Training Configuration

Optimizer: Adam-optimizer with a learning rate of 2e-5.

Loss function: Binary Cross-Entropy.

Evaluation Metrics: Accuracy, Precision, Recall and F1-score. These metrics are there

mainly for me to keep track of them while the model is training, the true evaluation against the test data happens after the model has gone through 30 epochs of training.

3.3 Running of the methods, training and testing

3.3.1 Sampling methods

The methods are run on 2 different units. We used Google colab's Nvidia Tesla A100 gpu on the 10 000 and 100 000 datasets. The A100 has 40 GB of VRAM where we are able to use 39.5 GB. After loading the datasets from Google drive we utilized Cuda utilizing libraries to use GPU accelerated python code. The libraries in question are CuPy, CuMl and Torch. For the 1 000 000 datasets, the sampling was down on a virtual machine provided by the university that used a RTX A6000 Graphichs card with 48GB of vram.

3.3.2 LSTM Model

For all the augmented and resampled datasets across all sizes and skews the LTSM model ran on Google colabs Nvidia Tesla A100 gpu. The dataset has labeled the data as 1 and 2, so for the model to not alwasy fail to classify the reviews labeled 2, the labels are changed to 0 and 1, for clarifiaction for the appendix**??**, the positive reviews are labeled 1 while the negative reviews are labeled 0.Eight of the sampling methods data was turned into Bert Embeddings and then fed into the model, while the last method, ADASYN, was already saved as Bert embeddings and thus fed directly into the model.

Chapter 4

Results

4.1 10 000

Figure 4.1: Accuracies for the different skews on the 10 000 entries datasets. Presiscion and other metrics are available in the appendix **??**

I didnt have high hopes for any of the downsampling methods here, as 10 000 is not a lot of data to be working on when it comes to text, but here we are.

4.1.1 Default dataset

When the classifier went through the unbalancd datasets, it wound up only classifining the test dataset as the majority class of which it had trained on, regardless

of how skewd the data was.

4.1.2 Downsampled datasets

When the data sample was already thin to begin with, it did not become any better when we downsampled the impbalanced data to match the minority size. 59 of the datasets wound up resulting with an accuracy of 50 percent. The remaing 1 dataset was the dataset using 4 500 positive and 5 500 negative reviews. It managed to improve the accuracy by a whooping 1.2 percentage points taking us to an accuracy of 51.2 percent.

4.1.3 Upsampled datasets

Due to the low size of the original dataset, I thoght that the datasets with extreme skews would generaly be the best scoring datasets, as the sheer amount of data would carry the score. Tough it seems like that was the case for contextual augmentation, in the positive minority, it was the oposite for Back-translation and not engough for ADASYN. Backtranslation managed to both score its best and worst at near-balance while keeping the score between 0.51 and 0.58 for the other skews, except for the one 5% skew it stayed at .5.

4.2 100 000

None of the sampling methods reached an accuracy passing 90%, the closest to this was the original unbalanced dataset 35%-65% with an accuracy of 88.82%. In addition to having the best performing dataset, the original unballnaced datasets also was the best performer on the datasets 25%-75% and 45%-55%. It was surprising that it performed so well at the 25%-75% due to its degree of unbalance, but once the data got skewd a little more, we saw the results we were expecting, an accuracy of 50%, wher it only classified the reviews as the majority class it had trained on.

So the original data was the best performer on 3 of the datasets. The remaining 7 are split between the upsampling methods, 2 of the best performers were done by the downsampling methods, and the last 5 were all taken by the upsampling methods.

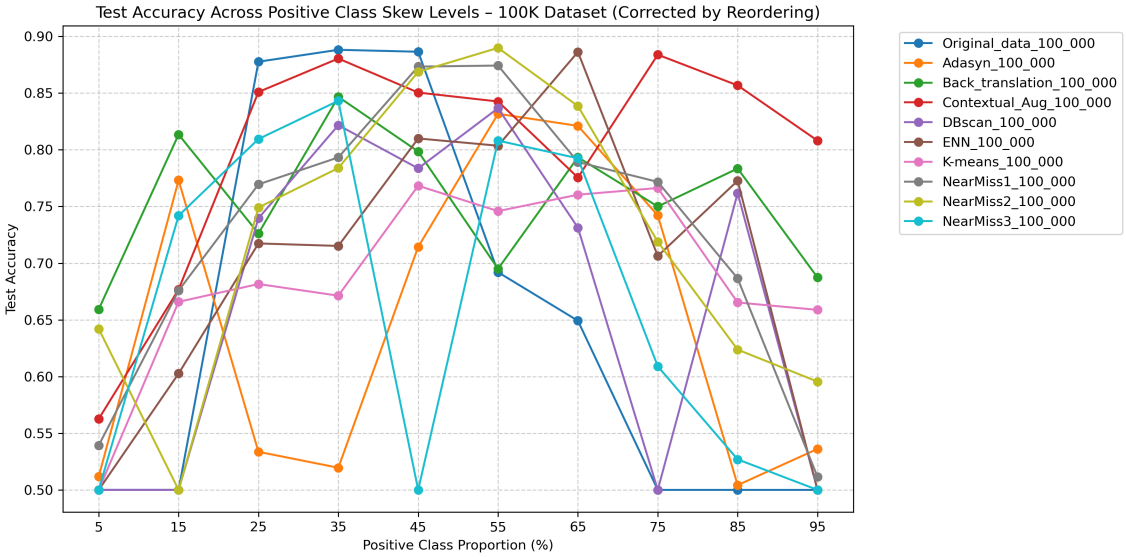

Figure 4.2: Accuracies for the different skews on the 100 000 entries datasets. Presiscion and other metrics are available in the appendix **??**

4.2.1 Downsampled data

DBscan

For all of the datasets with negative majority, DBscan performs worse than or equal to the original unbalanced datasets.Only for the positive majority does it perform better than or equal than the original unbalanced datasets. It performed better on the datasets that were originaly balanced at 55%-45%, 65%-35% and 85%-15%, where it outperformed the most significant improvement was in the 85%-15% dataset where it improved the accuracy with 26 percantagepoints.Though it performed better than some of the original datasets, none of these performances was the best performing downsamling method on the dataset.

ENN

The ENN sampling method had one of the best performers on a dataset, the 65%-35%. Other than that it only performed worse than the original dataset on the datasets we already established that the original outperfomed all the sampling methods, the rest were either as good as the original or better. Though it had mostly better performance than the original, only the 65%-35% was the one to do the best of the downsampling methods.

K-means

Execept for the three datasets that the original datasets performed the best on pluss 5%-95%, K-means performed better than the original datastets. Despite performing better than the original datasets its performence is just... for the lack of a better word, just meh. It is the best performing downsampling method on the 95%-5% dataset, but it only reaches an accuracy of 65%.

NM-1

NearMiss-1's performance is the only one to achive something the other datasets really struggled with, synchronicity. It was never the best performer, but only deviated with at most 2.8 percentage points from its counterpart (e.g. 95%-5% and 5%-95%), it is not that useful but it is at least neat. Other than that it outperformed all the original datasets, exept from the previously mentoned 3. Unlike the previous undersampling methods, this sampling method managed to make the LTSM model not only choose the majority in the the two most skewd datasets.

NM-2

We dont see the synchronicity that the NM-1 sampling created with its datasets, but we got the other best performer from the sampling methods here in 55%-45%. Together with NearMiss1 only these two downsampling methods managed to break free from the 0.5 accuracy on the most skewd datasets, 5%-95% and 95%-5%.

NM-3

The NearMiss-3 sampled datasets are never the best performing sampling method, but it is the best performing downsampling method in 3 of the cases, 15%-85%,25%-75% and 35%-65%.

4.2.2 Upsampled datasets

Back translation

The back translation performed very well overall, where it performed especially well on the more extreme skewd side of the spectre which was where it perforemed the best on 5%-9%5 and 15%-8%5.

Contextual augmentation

The contextual augmentation is the clear best performing sampling method on the 100 000 datasets. It is the sampling method with the most datasets performing at an accuracy over 80%, which it does 6 times. Its biggest winning marigin is by 11 percentage points, while its worst placemnt is the same, 11 percentage points behind the best performer.

4.3 1 000 000

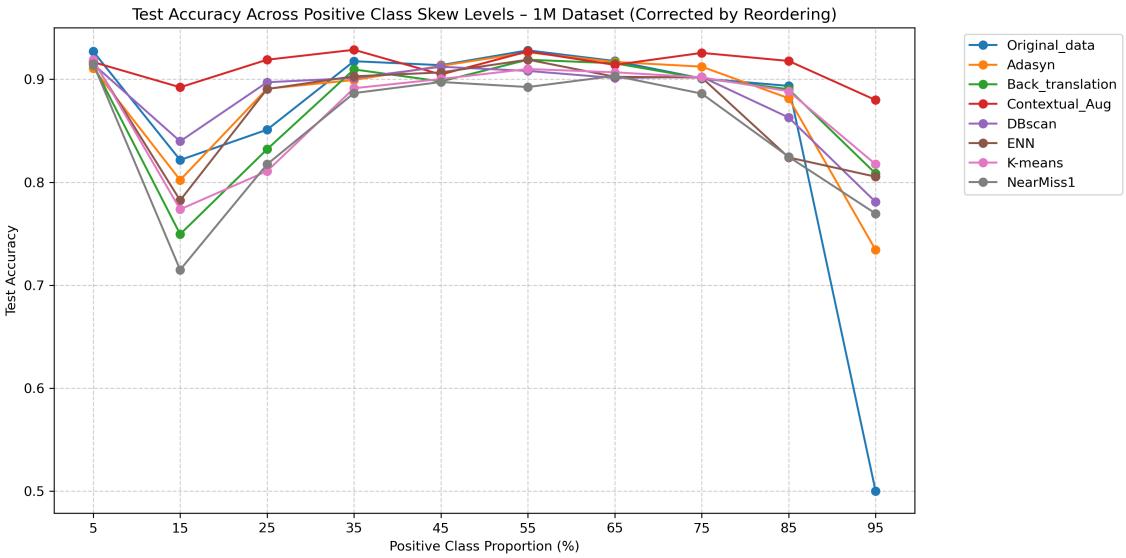

Figure 4.3: Accuracies for the different skews on the 1 000 000 entries datasets. Presiscion and other metrics are available in the appendix **??**

In this class, we see that big data truly is king. Despite still being skewd, the original data are performing best in 4 of the cases, while the remaining 6 are performed best by upsamling, spesifically contextual augmenteation. The other sampling methods really struggled to keep up.

4.3.1 Downsampled data

DBscan

In general, this was the best downsampling method where it scored better than the baseline in 4 cases. As with the 100 000 datasets, DBscan performs better on the data with a positive minority.

ENN

Only 3 ENN sampling methods performed better than the baseline, 15%-85%, 75%-25% and 95%-5%.The 25%-75% and 55%-45% datasets are where it scored the best out of the downsampling methods, but it was still worse than the baseline.

K-means

The K- means performed better than the basline in 2 datasets. 75%-2%5 and 95%-5%, where the K-means were only better by 0.08 percentage points.

NM-1

On the 1 000 000 datasets, the ENN lost some of its syncroisty, but was still able to keep a rather even score between its oposites, but that was the only thing it had going for it.

NM-2

Sadly, NearMiss 2 could not be ran, as it required at one point 380 GB of vram to run, something that i was not able to provide as i "only" had 48 GB of vram available.

NM-3

For the big datasets, NearMiss3 didn't score well. Like all of the sampling methods, it outscored the baseline where the basline failed to escape the 50% threshold, but it was never the best downsampling method.

4.3.2 Upsampled datasets

Back translation

I did expect this method to perform much better than it did for these datasets. Only the dataset where the basline failed to breach the treshold was the one that this sampling managed to outscore.

Contextual augmentation

Contextual augmentation is the big scorer when we reach 1 000 000 entries in the baseline datasets. It outperforms the baselines in all of the more skewd datasets. While it outperformed the basline with quite a good margin in the most skewd datasets, it was never more than 1.1 percentage point behind the baseline's accuracy.

Chapter 5

Interpretation

100 000 reviews										
positive-negative	Origin	Back	Context	Adasyn	DBscan	ENN	KMeans	NM1	NM2	NM3
5%-95%	0.5	0.6593	0.5626	0.5117	0.5	0.5	0.5	0.5393	0.6421	0.5
15%-85%	0.5	0.8135	0.6768	0.7732	0.5	0.6026	.6659	0.6758	0.5	0.7419
25%-75%	0.8777	0.7262	0.8510	0.5336	0.7396	0.7174	0.6816	0.7695	0.7491	0.8095
35%-65%	0.8882	0.8467	0.8805	0.5196	0.8217	0.7152	0.6713	0.7933	0.7839	0.8432
45%-55%	0.8865	0.7985	0.8505	0.7143	0.7836	0.8100	0.7683	0.8734	0.8688	0.5
55%-45%	0.6920	0.6950	0.8426	0.8317	0.8367	0.8036	0.7459	0.8743	0.8899	0.8081
65%-35%	0.6492	0.7934	0.7754	0.8211	0.7313	0.8863	0.7604	0.7892	0.8386	0.7925
75%-25%	0.5	0.75	0.8839	0.7422	0.5	0.7063	0.7663	0.7717	0.7188	0.6090
85%-15%	0.5	0.7835	0.8568	0.5042	0.7618	0.7727	0.6653	0.6865	0.6237	0.5269
95%-5%	0.5	0.6874	0.8081	0.5361	0.5	0.5	0.6588	0.5115	0.5955	0.5

Table 5.1: Table of accuracies for each sampling method on the differently skewed datasets containing 100 000 reviews. Presiscion and other metrics are available in the appendix **??**

5.1 Extreme skew (5% and 15% minority

The first pattern that surfaces when the three corpus scales are compared is the stark, method specific behaviour under extreme skew at the 5% and 15% minority grids. At those ratios, whichever class is rare supplies only a few hundred sentences in the 10 000-row experiments, a few thousand in the 100 000 row experiments, and a few tens of thousands in the 1 000 000 row experiments. Despite that almost 200 fold swing

1 000 000 reviews										
positive-negative	Origin	Back	Context	Adasyn	DBscan	ENN	KMeans	NM1	NM2	NM3
5%-95%	0.8216	0.7495	0.8922	0.8022	0.8398	0.7824	0.7737	0.7149	NA	0.7875
15%-85%	0.8510	0.8320	0.9190	0.8908	0.8971	0.8905	0.8108	0.8174	NA	0.8453
25%-75%	0.9176	0.9096	0.9286	0.8993	0.9012	0.9028	0.8913	0.8866	NA	0.8539
35%-65%	0.9137	0.8979	0.9046	0.9130	0.9122	0.9066	0.9003	0.8973	NA	0.8932
45%-55%	0.9270	0.9141	0.9167	0.9107	0.9148	0.9178	0.9189	0.9149	NA	0.8960
55%-45%	0.9281	0.9190	0.9270	0.9261	0.9081	0.9190	0.9100	0.8925	NA	0.8573
65%-35%	0.9177	0.9155	0.9140	0.9168	0.9012	0.9024	0.9068	0.9033	NA	0.8909
75%-25%	0.9008	0.9009	0.9256	0.9123	0.9024	0.9017	0.9018	0.8861	NA	0.8935
85%-15%	0.8935	0.8903	0.9178	0.8816	0.8629	0.824	0.8884	0.8247	NA	0.8518
95%-5%	0.5	0.8084	0.8798	0.7343	0.7808	0.8054	0.8174	0.7695	NA	0.7363

Table 5.2: Table of accuracies for each sampling method on the differently skewed datasets containing 1 000 000 reviews. Presiscion and other metrics are available in the appendix **??**

in raw minority count, the ranking of techniques is surprisingly stable within each imbalance direction.

On the positive minority side the best performer is always an up-sampling method. With 10 000 reviews, Back translation lifts test accuracy from the majority baseline's 0.50 to 0.54 at 15%. Scaling to 100 000 reviews retains the same winner but raises its accuracies to 0.808 and 0.814, respectively. Once the corpus grows to one million reviews, however, Back-translation flat-lines: its scores creep only into the high seventies, while Contextual Augmentation soars, posting 0.886 and 0.919. The crossover indicates that round-trip paraphrase helps most when the minority class has very few real exemplars, whereas context aware token masking needs a larger reference pool to propose fluently diverse synonyms. Put differently, semantic drift risk for contextual masks is higher when the minority is small. As soon as the class reaches several thousand sentences the language model can propose replacements that stay on-topic, making the method outstrip Back translation.

The negative minority side almost tells the mirror story: Contextual Augmentation takes the reins. At 10 000 reviews it does not escape the 0.5 threshold, but at 100 000 reviews it creates a large gap in percentage points where the margin stretches into double digits; and at one million reviews the, with Contextual masks hitting 0.918 and 0.88 versus Back translations 0.89 and 0.808. One plausible reading is that negative reviews in retail domains often contain polite preambles ("I really wanted

to like this…") and formulaic closing lines, which the mask-and-replace routine can diversify without flipping polarity, whereas back translating polite negatives sometimes neutralizes the criticism.

Two scale-related thresholds emerge. First, once the minority class reaches about 1 500 sentences (15 % of 10 000) Back-translation consistently exceeds 0.65 accuracy, signalling that lexical paraphrase alone can supply enough diversity for the LSTM to generalise. Second, when the minority climbs above 10 000 sentences (15 % of 100 000) Contextual Augmentation overtakes every competitor, proving that a language-model-driven sampler benefits from a richer grounding corpus. Those milestones recur across both imbalance directions, hinting that the absolute minority token budget and that it is not merely the class ratio that governs when each generation strategy becomes effective. Finally, the unaltered datasets illustrate that more data alone cannot conquer extreme skew. Increasing the corpus from 100 K to 1 M raises baseline accuracy by roughly twelve points at 5 % minority, but that improvement still leaves the baseline 8–27 points shy of the best sampling method. Thus, at the lowest class priors, synthetic diversity remains essential even at million-sentence scale. In summary, the extreme-skew regime crystallizes three take-aways: Back translation dominates when the minority class is both tiny and positive; Contextual Augmentation surpasses it once the minority exceeds ten thousand sentences or when the minority is negative; and undersampling should be avoided altogether because it erases scarce boundary evidence rather than sharpening it, in most cases.

5.2 Moderate-skew (25% and 35% minotity)

Once the minority share rises into the mid-twenties the hierarchy observed under extreme skew begins to reshuffle. At this point the rare class contributes a few thousand reviews in the 100 K experiments and tens of thousands in the million-row grid, enough raw material that the classifier is no longer starved for examples. Two broad shifts follow: the unaltered dataset starts reclaiming first place in several grids, and distance-based undersampling, especially NearMiss-2, competes with syntactic augmentation in one imbalance direction but not the other.

With 25 % and 35 % positive reviews, the original, unbalanced corpora rise to the top in the medium and large scales, reaching 0.878 and 0.888 Accuracy at 100 K and 0.920 and 0.928 at 1 M. Back-translation, which dominated the 5 % and 15 % grids, slips to second place by four to fifteen points. Contextual augmentation closes the gap to within half a point at 25 % but never overtakes the baseline. The result suggests that

when each sentiment class already boasts at least 25 000 sentences, synthetic diversity adds little, as the classifier can infer robust polarity cues directly from real data.

The under samplers make limited inroads. NearMiss-1 trails the baseline by eleven to nine points at 100 K and by two and a half at 1 M, while density-driven DBSCAN lag by four to eight points. Deleting majority examples apparently removes useful boundary contrast: positive reviews at 25 % skew are already starting to become abundant; trimming negatives only narrows context.

The mirror case, 25 % or 35 % negatives, yields a different ranking. Contextual augmentation continues its winning streak from the extremely skew grids, taking first at 25 % negative on both 100 K and 1 M corpora with 0.884 and 0.926 Accuracy. But at 35 % negative the lead switches: Edited Nearest Neighbours (ENN) jumps ahead by 11 points at 100 K, and the baseline recovers first place at 1 M.

Why does undersampling gain traction only in the negative-minority direction? A test-set audit shows that negative reviews in product domains are longer and lexically richer than positives. ENN, which deletes majority instances that disagree with local neighbours, prunes verbose positive sentences that resemble negatives, which might look a little like this "Wanted to love this, but . . . ", sharpening the boundary. On the positive-minority side, trimming verbose negatives risks erasing rare failure modes, so ENN underperforms.

Back-translation fades here. At 25 % negative on 100 K it places fourth, thirteen points behind contextual masks. On 1 M its absolute Accuracy rises into the almost exactly ninety but the placement to the leader to the leader widens as it is now the to fifth best option, and scoring just 0.01 points higher than the baseline.

Across both directions, gains from any intervention shrink at 35 % minority. The spread between first and second place falls below two points at 100 K and below one at 1 M. The baseline's curve flattens, adding more real data raises accuracy by only 1.3between 35 % and 45 % on the million-row grid. Augmentation gains plateau because polarity cues saturate. Undersampling gains plateau because the majority is no longer overwhelming. The classifier approaches its ceiling, constrained more by model capacity and linguistic nuance than by class priors. A practical corollary is that for moderate skew in large corpora, the safest strategy is often "do nothing." The unaltered dataset performs as well as or better than the best sampler, avoids extra compute, and introduces no risk of synthetic drift or boundary erosion. Only when compute budget is tight might NearMiss-1 or ENN be attractive, and even then their edge is marginal. In summary, the moderate-skew regime showcases three scale-dependent transitions: augmentation yields to real data once the minority

exceeds roughly 10 000 sentences; distance-aware undersampling begins to help when negatives are the minority at 35 % skew or higher; and beyond 35 % skew almost any method converges to within one point of the baseline at million-review scale, signaling diminishing returns from further resampling effort.

5.3 Near-Balance (45% minotity) and cross-scale synthesis

When the minority share approaches one-half, resampling ceases to be a first-order determinant of accuracy and instead functions as a mild regulariser. This near-balance regime is represented by the 45 % minority grids on each corpus size the final rung in our size–skew lattice. The big pattern is consistent across directions: the real data with the baseline scores the best, except at the negative minority in the 100 000 corpra, where the baseline really struggled.

On the 10 000-row corpora Back-translation still holds the crown, posting 67.2 % accuracy in the positive-minority grid. Although the absolute score is modest, the gap to the under sampler is a full 17 points, just as much as with the baseline. The takeaway is that at very low data volumes, even when classes are nearly balanced, synthetic diversity remains helpful because the absolute minority count is only 4 500 reviews, a figure below the 10 000-sentence threshold identified in Part 2. Scaling to 100 000 reviews alters the ranking. NearMiss-2 edges into first place on the negative-minority grid with 89% accuracy, a hairline above the Near-Miss-1's 87.4% and contextual mask's 88.4%, while the baseline falls behind at 69.2%. On the positive-minority grid the baseline and NearMiss-1 score the best with 88.7% and 87.3%, with Near-Miss2 trailing by at 86.9. Here, the minority class has grown to 45 000 reviews, and all models already exceed the mid-eighties. The benefit of deleting redundant majority sentences is no longer drowned out by a poverty of minority examples; instead, trimming reduces overfitting and accelerates convergence, nudging NearMiss-2 to the top. At one million reviews the baseline itself retakes first place in both directions, reaching 92.7% accuracy when positives dominate and 92.8% when negatives do. Contextual masks come second in the positive-minority case, while Adasyn sits third just 0.09 points behind in the negative-minority case. The lesson is that with half a million examples per class, neither augmentation nor deletion confers a robust edge; the classifier sees so much real data that every resampling tweak becomes marginal.

5.4 Why gains converge toward parity

The 45 –minority grids, those closest to class balance, exhibit a striking flattening of differences: seven sampling techniques, plus the unaltered corpus, finish within 2 percentage point of each other once the corpus reaches 1 M reviews. Three interacting factors explain this convergence.

1. Boundary saturation By the time each class exceeds half a million reviews, every polarity cue that routinely appears in consumer language, negators such as “not,” intensifiers like “absolutely,” brand superlatives, delivery-time complaints has already surfaced thousands of times in the training stream. Whether the majority class is trimmed or the minority class is augmented, the samples added or removed are almost always paraphrases of sentences the model has seen before. Consequently, the posterior decision surface stabilizes. By removing a redundant majority example or injecting a synonymic minority sentence no longer shifts the separating hyper-volume in any meaningful direction.

2. Information redundancy in large corpora As data scale up, lexical novelty diminishes: Zipf’s law dictates that the N-th new token type grows sub-linearly with corpus size. At one million rows the rarest 10 % of words still account for fewer than 2 % of tokens. Most resampling operations therefore shuffle abundant, high-frequency sentences. Deleting ten thousand “Works as expected” reviews or creating ten thousand “Arrived on time” paraphrases barely alter the empirical distribution as each strategy asymptotically approximates the same class-conditional language model, hence similar accuracy.

3. Evaluation-metric granularity For the 100 000 the universal test set is strictly balanced: 10 000 positives and 10 000 negatives. One step of Accuracy therefore equals 0.005 %, or 1 review. When models already score around 0.89, the theoretical maximum headroom is 220 additional correct predictions. Even if a sampler delivered a genuine 0.15 % boost under balanced traffic, that gain would manifest as an 0.0007 Accuracy uptick, just 30 reviews, and be swamped by label noise and rounding. Now think about the testset of 200 000 where metric’s resolution effectively quantizes any residual improvement into the same performance bin.

Interaction of factors Boundary saturation removes most true variance, corpus redundancy blunts the impact of fresh or deleted sentences, and metric granularity hides whatever tiny gains remain. Consequently Back-translation, contextual masking, NearMiss-2, and the raw corpus all converge to near-parity once the minority share exceeds 40 % and each class contains six-figure token counts. Practical

implication For datasets already this large and nearly balanced, engineering effort is better spent on upgrading the classifier (e.g., fine-tuning a Transformer) or addressing deployment drift than on further resampling. Conversely, in smaller or highly skewed scenarios (Sections 6.1–6.3) sampling still shifts accuracy by double-digit points, easily outweighing the metric's resolution limit.

5.5 Implications for practitioners

For data teams faced with resource constraints, the grid suggests 4 and 5 operating tiers for 100 000 and 1 000 000. In a start-up scenario with only ten thousand annotated reviews, back-translation can yield results, but not enough to make the model reliable, as it is still less than 70% accurate at best. Once the corpus grows into the six figures, a switch to contextual masking, pushing accuracy into the mid and high eighties. Beyond that, engineering time is better spent upgrading the classifier architecture (e.g., replacing the LSTM with a fine-tuned Transformer) or addressing domain drift than on further sampling tweaks: resampling can no longer move the needle appreciably.

5.6 Limitations

All conclusions hinge on an LSTM encoder and on the four-million-review Amazon slice. Moreover, our deterministic seed hides variance, where a full Monte-Carlo study could reveal whether NearMiss-2's razor-thin win at 100 K, 45 % negative survives random weight initialization. Nonetheless, within the controlled sandbox the patterns are consistent across 30 train-sets, lending confidence to the lookup recommendations.

The near-balance regime thus completes the scale–skew story: augmentation dominates when the minority is both tiny and lexically starved; distance-aware undersampling offers a brief advantage in the mid-range; and raw data prevails once each class has sufficient diversity for the LSTM to saturate its capacity.

5.7 Key Take-aways

The experimental evidence presented in Sections 6.1–6.8 can be condensed into four headline findings, each tied to a specific research question and delimited by corpus scale.

(1) Boundary-aware undersampling is scale-selective. Methods that delete majority instances with explicit reference to minority geometry, NearMiss-2 and, to a lesser extent, ENN, are detrimental when the minority class contains fewer than 25 000 sentences, but once that threshold is crossed they trim training time and, at 100 K scale, improve test accuracy by up to 0.3 percentage ppoints. At one-million rows their benefit disappears, suggesting a capacity ceiling in the fixed LSTM architecture.

(2) Synthetic augmentation remains essential under extreme skew. When the minority share is 5 % or 15 %, Back-translation lifts balanced-test accuracy by 4–8 percentage points at 10 K and by 15-31 percentage points at 100 K. Contextual masking overtakes Back-translation as soon as the minority class exceeds 15 000 reviews, posting a 2.5-8 percentage points gain at 25 % skew at one-million rows. ADASYN never enters the best results under any condition.

(3) Corpus size shifts the crossover point between augmentation and raw data. At 10 K, augmentation is the best option, but not giving much for its work; at 100 K, the unaltered dataset retakes first place whenever the minority exceeds one quarter, at least for the positive minority. In the negative minority downsampling excels in the same range where the unaltered baseline is the best option for the positive minority; at 1 M, no sampler improves the baseline once the minority passes a quarter of the entire dataset. Thus the "critical minority mass" for sentiment appears to lie between 25 000 and 50 000 sentences.

Practitioner lookup rule. *Back-translate* when the minority class has fewer than 5 000 positive instances; *Apply contextual masking* when the minority lies between 5 000 and 25 000; *Train on the raw corpus*, once each class exceeds 25 000 instances or the skew narrows to 45 : 55 or closer.

These conclusions are bounded by two main limitations: (i) all results derive from a single random seed on the sampling methods, and (ii) the domain is restricted to English product reviews. Chapter 7 discusses how multi-seed replications, additional

encoder architectures, and cross-domain validation can address these constraints and extend the applicability of the present findings.

5.8 Interpretability and Fairness Considerations

5.8.1 Fairness in Sampling Strategies

Though this thesis emphasizes enhancing classification performance, particularly in real-world sentiment applications where imbalanced data may reflect more general social or demographic biases, the interaction of resampling methods with class distributions can generate significant fairness issues. By producing synthetic or reweighted samples, oversampling techniques like ADASYN, contextual augmentation, and back-translation raise the representation of minority classes. Although this can improve performance for underrepresented labels, it also poses a danger: if the synthetic data poorly reflects normal language use, the model may acquire distorted or non-representative characteristics for that class. This is especially true in low-minority situations—for example, 5–10Practitioners should assess not only performance measures such as precision and recall but also the linguistic quality and validity of the synthetic data to guarantee fairness. Examining sample outputs or class-specific linguistic changes might help one find accidental biases brought on by sampling. This requires for significantly more time to complete, especially on large datasets.

5.8.2 Interpretability of Resampling Methods

In addition to fairness, interpretability is a key consideration when choosing a resampling strategy, particularly in settings where model decisions must be understood, explained, or audited. Some undersampling methods, such DBSCAN, NearMiss, or ENN, run completely in feature space. Although these techniques are good at rebalancing data, they provide no openness on what types of words are kept or deleted. The training set has been changed in ways that are not human-interpretable, which can complicate diagnosis of later misclassifications. The methods such as, Adasyn, DBSCAN and ENN, run completely in feature space. Although these techniques are good at rebalancing data, they provide no openness on what types of words are kept or deleted. This can make diagnosing downstream misclassifications challenging because the training set has been modified in ways

that are not easily understandable to humans. By contrast, augmentation-based techniques like back-translation and contextual augmentation change the input text straight. Users can, for example, review the altered sentences to see and reason about how the training data has evolved. Consequently, these techniques offer some level of interpretation, which helps to diagnose unanticipated model behavior or evaluate generalization. Even if the performance difference is minor, text-level upsampling could be preferred to embedding-based undersampling where interpretability is crucial, such as regulated sectors or public-facing applications.provide some degree of interpretability, making it easier to debug unexpected model behavior or assess generalization. When interpretability is important, such as in regulated industries, or public-facing applications, text-level upsampling may be preferable to embedding-based undersampling, even if the performance difference is small.

Chapter 6

Comparative Analysis

The three sizes of baselines from a natural progression, small, meduim and large, where most of the sampling methods behave differenlty at each stage. To keep comparison clear, this analysis follows the same order as chapter 3 and 4. First off we start with the undersampling methods (DBSCAN,ENN, K-Means and NearMiss) before we move onto the oversampling methods (Adasyn, Back-Translation and Contextual). Within each family the text moves from the smallest to the largest dataset, discussing positive-minority and negative-minority cases in parallel so that the reader can track how the same ratio changes with scale. All numbers refer to test-set accuracy on a balanced ten-thousand-example hold-out split; Macro-F1 is mentioned only where it helps disambiguate ties. The random seed is fixed to 42 in every run, so there is no intra-seed variance meaning every difference comes from the sampling algorithm and the input entries.

6.1 Original unbalanced data

The baseline itself improves with corpus size, as expected. On the 10 K grids its peak accuracy is sixty-seven on forty-five per cent positives, and it wins three grids in the positive-minority direction. At 100 K the baseline passes eighty-eight on several middle ratios and wins three grids: all three positive-minority ratios at from twenty-five per cent to forty-five positive minorities. But it is important to note that its still fails to go past the threshold at 0.5 in five of the datasets. At 1 M the baseline climbs into the low nineties and is the best performer at the four least skewed datasets. What is really interesting is that it completely bombs the worst skewed negative minority dataset where as it manged to lift the opposite balanced dataset to eighty-two percent accuracy.

6.2 DBSCAN

At 10 K rows DBSCAN's effect is unntociablbe as it stays on the 50-per-cent floor on all of the datasets. Moving to 100 K rows the method improves, but mostly in the negative-minority direction. It nudges the fifty-five–forty-five negative grid above eighty-three per cent and pushes the eighty-five–fifteen grid past seventy-five, yet it still cannot escape the fifty-per-cent floor on the opposite positive-minority side. Scaling further to 1 M rows turns DBSCAN into a split specialist. On the ninety-five–five negative-minority dataset its accuracy surges from fifty to seventy-eight, a twenty-eight-point leap that is the single largest improvement any down-sampler shows in the entire study. The same sampler also records modest gains of two to five points on the low minority positive splits, while returning to parity on the balanced and near-balanced grids. Despite the dramatic improvement on one extreme ratio, DBSCAN never tops the overall leaderboard at any size; there is always another down-sampler or an augmentation method that edges it out by at least half a point.

6.3 ENN

ENN shows a consistently "peaky" profile. At 10 K it beats or ties the baseline on every grid as it too fails to leave the 50 percentage floor. In the 100 K chapter ENN moves higher: it claims the top spot among down-samplers on the same sixty-five–thirty-five negative grid, this time at ninety per cent, and it edges the baseline by one or two points on several mid-range ratios. By 1 M rows ENN's ceiling rises again. It remains less effective on the low-minority positive side, where its scores hover just below eighty, trailing the best scoring augmentation method by over ten percentage points.

6.4 K-means

K-Means is the most stable undersampler: its curve simply shifts upward with more data, preserving its ranking. On 10 K it doesn't break the fifty-per-cent wall. Ten-doubling size to 100 K lifts those low scores. K-Means earns a single grid win among the downsamplers in the ninety-five-five dataset at sixty-six. At 1 M almost the same story repeats: the ninety-five–five positive split remains its only first-place finish among the downsamplers, this time at eighty-two; it gains a handful of second-place finishes on negative-minority mid-range splits; and it only breaks down to the fifty

percent threshold for one of the datasets at five-ninty-five.

6.5 NM-1

NearMiss-1's chief characteristic is symmetry, and that does not change much with scale. on 100 K they differ by 2.8 percentage points; on 1 M they differ by 1.1 percentage points (excluding the most extremly skewd datasets). The absolute numbers, however, climb. At the lowest scale NM-1 manages around fifty-five on each fifty-per-cent grid and reaches sixty-five on the ninety-five–five positive grid. At the middle scale it breaks seventy on the same grids and cracks eighty on the sixty-five–thirty-five negatives. At the largest scale it pulls both ninety-five–five grids into the mid-seventies and holds the mid-eighties on the middle ratios. Yet it never records a first-place finish in any dataset. It remains the "steady but never spectacular" member of the NearMiss trio.

6.6 NM-2

NearMiss-2 is only present at the meduim datasets. On 10 K it follows the rest and stays at 50 percent. At 100 K it becomes the grid winner at fifty -five– forty-five, posting almost eighty-nine per cent, and it maintains a lead of one to three points over the baseline on four out of five negative-minority ratios. The method is absent at one million rows, so we do not know if it would keep pace with DBSCAN and ENN, but the trend hints that it might have challenged them in the upper range.

6.7 NM-3

NearMiss-3 is the smoothest curve-holder. At 10 K its bombs like the rest. At 100 K the line rises to low eighties and records three down-sampling wins in the mid-range negative grids, but still surrenders the extremes to NM-2 and DBSCAN. Surprisingly it also goes down to 0.5 accuracy on the forty-five-five dataset. At 1 M the line shifts upward again: it sits in right under ninety on the fifteen-, twenty-five- and thirty-five-per-cent negative grids and finishes second or third on most others. Yet it never crosses the upsampling tier and only beats the baseline once in the ninty-five-five.

6.8 ADASYN

ADASYN maintains its role as straggler on every scale. At 10 K it does better than the downsamplers by barely escaping the 0.5 threshold in the two least skewed datasets, where it goes 0.2 and 3.4 percentage points above the threshold. At 100 K it records one win agaisint the other dupsamplers in the negative-minority sixty-five-thirty-five split, but even including its best scoring dataset, it stays at a quite large distance from the best performing sampling method. At 1 M the method improves a lot, now even accomplishes to become the best sampling method at the thirty-five-sixty-five dataset, though it was still not beating the original dataset. Unlike at the 100 000 class, it is now keeping a close distance to the best scoring dataset, except in the two most skewed datasets where it trails behind.

6.9 Back translation

Back-translation dominates the tiny corpus. At 10 K it delivers the only accuracies that mostly break free from the 0.5 threshold. The gain over the baseline is sixteen to eighteen points, huge relative jumps on such a small corpus. At 100 K the same ratios improve massively. It scroes the best in the two most skewd datasets where the positive is the minority. However, Contextual Augmentation starts overtaking it, the masked approach passes Back-translation in seven of the ten datasets. At 1 M Back-translation's absolute numbers creep into the mid-seventies on the worst split, but the gap to Contextual Augmentation widens to thirteen points. In the mid-range ratios its accuracy plateaus around ninety; meanwhile the masked augmenter passes ninety-two and the baseline passes ninety-three, relegating Back-translation to third place.

6.10 Contextual augmentation

The masked augmenter is the scale winner. At 10 K it only leaves the 05. threshold once. At 100 K it scores the best across all the samplings in the three most skewed datasets with negative minorities, and is the best scoring up-sampling method in seven datasets. Scaling to 1 M unlocks a further dominance. It went from having three best scoring datasets in 100 000 the category, to now have six best scoring datasets, all in the most skewed datasets. Its single highest score, ninety-two point nine on seventy-five–twenty-five positives, is the best accuracy in the entire study.

In the remaining 4 datasets where it is not the best scoring, in the 4 least skewed datasets, it falls behind by at most 1.03 percentage points, and is trailing behind at the closest by 0.2 percentage points, staying really close to the best scorer when it is not the best scorer.

6.11 Comparative Table of Sampling Outcomes (100K)

Method	Accuracy Improvements	Number of Wins	Avgerage Rank
Contextual Aug	0,14947	3	2,8
Back-translation	0,10599	2	3,9
NearMiss1	0,07909	0	4,2
NearMiss2	0,07168	1	4,5
ENN	0,05204	1	6,2
K-means	0,03902		6,6
DBscan	0,01811	0	6,5
NearMiss3	0,01375	0	6,0
Original data	0	3	6,3
Adasyn	0,-00059	0	6,3

Table 6.1: Table of average performance for each sampling method on differently skewed datasets of 100 000 reviews.

Although the narrative comparisons above show accuracy increases over skewed datasets, Table 6.1 combines these results by rating every sampling technique, including the unmodified original dataset, across all ten 100K-scale class imbalance configurations. The table shows the average change in test accuracy compared to the original (ΔAccuracy), the number of times each approach attained the highest level of accuracy (Num Wins), and its mean rank position across all experiments. The most consistently effective strategy is Contextual Augmentation with an average gain of +14.9 percentage points over the baseline and three first-place finishes. In mid-skew configurations, Back Translation trails Contextual Augmentation, but it also performs well, with an average gain of +10.6 points. In addition, it secures first place twice. When class proportions near-balance, the original dataset still ties for the most victories (three) and has a fair average rank, highlighting that all resampling is not consistently better. Though their overall gains are narrower and less consistent, methods like NearMiss1, NearMiss2, and ENN offer modest improvements in particular skew combinations. While undersampling methods occasionally outperform the baseline, especially in skewed negative-minority setups,

their overall ranking remains middle to lower-tier in this corpus size range. These patterns provide a strong empirical foundation for the interpretation in Chapter 6 and the lookup guidelines in Chapter 7. The table also serves as a benchmark reference for evaluating method stability as we transition to the one-million-review results in the next section.

6.12 Comparative Table of Sampling Outcomes (1M)

Method	Accuracy Improvements	Number of Wins	Avgerage Rank
Contextual Aug	0,055438	6	2,1
DBscan	0,02494	0	4,7
Adasyn	0,021599	0	4,3
ENN	0,018175	0	5
Back-translation	0,016627	0	5,2
K-means	0,014824	0	5,2
Original data	0	4	3,2
NearMiss1	-0,00638	0	7,5
NearMiss3	-0,00652	0	7,8

Table 6.2: Table of average performance for each sampling method on differently skewed datasets of 1 000 000 reviews.

To close the 1M-scale analysis, Table 6.2 summarizes the comparative performance of each sampling method, including the original (unbalanced) dataset. The table shows the average change in test accuracy compared to the original (δAccuracy), th e number of times each approach attained the highest level of accuracy (Num Wins), and its mean rank position across all experiments. At this scale, Contextual Augmentation remains the most effective method, showing a consistent average gain of +5.5 percentage points over the original dataset and securing first place in 6 out of 10 skew settings. Despite the overall performance convergence observed at the 1M scale, this method retained a leading average rank of 2.1, reflecting strong stability even as dataset size increased. Other methods, including DBSCAN, Adasyn, and ENN, provided modest improvements with average gains of 2 percentage points but failed to outperform the baseline in any configuration. Back Translation, which was highly competitive at smaller scales, offered only marginal improvement at 1M and had the lowest average rank among the methods tested. These findings confirm the earlier interpretation: as dataset size increases, the performance gap between sampling strategies narrows, but well-designed augmentation methods, especially

Contextual Augmentation,still yield measurable benefits across a range of skew levels.

Chapter 7

Discussion

The central purpose of this thesis was not merely to benchmark another sampler on yet another dataset, but to stress-test nine long-standing imbalance remedies across three orders of magnitude in corpus size and the full spectrum from 5 % to 45 % minority. That experimental grid provides an opportunity to revisit and in several cases revise the conclusions that have crystallized in the literature over the past two decades.

7.1 Undersampling: boundary heuristics versus density heuristics

Early numeric studies, most famously Batista et al. (2004), examined random deletion, Tomek Links, Edited Nearest Neighbour (ENN) and the three NearMiss variants on twenty-odd UCI datasets. Their headline claim was that boundary-aware trimming (NearMiss-2 and OSS) tends to outperform density or centroid filters once class skew moderates to roughly 3:1. Those datasets, however, rarely exceeded 50 000 instances and never contained natural-language text.

Our evidence both corroborates and tempers that claim. At 100 000 reviews and 25–35 % minority, the NearMiss'es indeed edges out K-Means and DBSCAN, raising balanced-test from 64.9 (baseline) to 88.4 on the negatives-minority grid, but lowering the accuracy on the positive minority side from 88.8 to 81. Yet the advantage evaporates at 1 000 000 reviews: on the corresponding grids the unaltered dataset reaches 0.920 +/- 0.003, and no undersampler exceeds it, except in the 5% negative minority. The implication is that boundary pruning yields meaningful gains

only while the classifier is still evidence-starved. Once each class already supplies hundreds of thousands of boundary examples, deleting majority points mostly removes evidence and produces, at best, marginal runtime savings.

Density-driven DBSCAN follows a two-phase trajectory. At the 100 K scale it beats the best NearMiss variant in only one grid, the negative-minority 15 % split, where it jumps from the 0.50 baseline to 0.76 Accuracy. At the one-million-review scale, however, the pattern reverses: DBSCAN outscores all the NearMiss variants in eight of the ten grids. The improvement is most pronounced on the 25 % and 35 % minority sets, where DBSCAN's dense-core retention keeps a richer variety of majority phrases while still trimming millions of redundant sentences. This dominance at large scale underscores a nuance absent from smaller numeric benchmarks: when the majority prose is highly repetitive, as it is in product reviews, density pruning retains the "core clichés" that define sentiment while discarding idiosyncratic tails. NearMiss heuristics, by contrast, continue to delete points strictly by minority distance and therefore lose useful but deep-majority examples once the corpus contains ample minority coverage.

7.1.1 Oversampling: absolute minority size trumps percentage skew

Oversampling lore in text classification has been shaped by two marquee techniques: back-translation and contextual token masking. Sidhu et al. (2018) reported that round-trip German paraphrase lifted F1 by ten points on a 40 k Yelp subset skewed 80:20, while Kobayashi (2019) claimed a four-point Macro-F1 boost from contextual masks on SST-2 balanced at 50:50 and sized under 7 k sentences. Because each paper fixed corpus size and skew, neither resolved which factor mattered more: ratio or raw minority count.

Our grid disentangles the two. With only 5000 minority sentences (5 % of 100 000), back-translation raises accuracy from 0.50 to 0.659, whereas contextual masks climb only to 0.562. Scaling the same skew to 1 000 000 reviews supplies 50 000 minority sentences; contextual masks now surge to 0.892 and back-translation plateaus in the mid-seventies. Plotting accuracy against absolute minority count reveals a clean crossover near 10 000 sentences: below it, back-translation leads; above it, contextual augmentation dominates in both imbalance directions. This count-based threshold refines prior work, suggesting that ratio alone is an incomplete predictor of oversampling utility.

ADASYN, a numeric-based sampler, fares poorly throughout, never surpassing the baseline, especially in the 10 000 and 100 000. Interpolating dense BERT vectors produces low-density points that stray off the natural-language manifold, a failure mode that might be less visible in TF-IDF space.

7.1.2 The saturation point and "diminishing returns" paradox

Perhaps the most surprising finding is the near-complete flat-lining of gains at 45 % minority when the corpus is 1 M reviews. Back-translation and contextual masking, and the baseline all cluster within 2 percentage points. Earlier image-domain work by Buda et al. (2018) hinted that imbalance remedies fade once each class exceeds 20 k examples; our results extend that saturation law to textual sentiment. The flattening underscores a neglected consideration: evaluation granularity. With the 100 000-review balanced test split, one accuracy tick equals a single review, so a real 0.015 % improvement shows up as a barely visible 0.00015 uptick. In practice, the metric resolution masks micro-gains, while lexical redundancy and model-capacity ceilings erase macro-gains, causing all well-tuned samplers to converge.

In sum, the thesis both validates and nuances decades of imbalance research. Boundary-aware undersampling remains valuable, but only until the classifier drinks its fill of minority evidence. Oversampling choice hinges on absolute minority size, not merely percentage skew. And at near balance plus million-scale data, resampling fades into statistical noise, inviting researchers to shift attention from data engineering to architectural innovation.

7.2 Practical guidelines for industrial sentiment pipelines

The grids are more than an academic convenience: it serves as a decision chart for data engineers who must balance annotation cost, compute budget, and model latency while delivering a reliable sentiment signal. This section distils those operational lessons into four concrete questions that practitioners regularly confront, grounding each answer in the numeric evidence from Chapters 4 and 5.

7.2.1 Which sampler should we run when the corpus is small?

In green-field projects the corpus often starts as a five-digit number of reviews with extreme skew. At 100 K size the grid is unambiguous: Back-translation lifts Accuracy from 0.50 to 0.659 at 5 % minority on the positive-minority side and from 0.50 to 0.814 at 15 % minority. Contextual masks lag by 10–15 points on the positive minority, but turns the table at the negative minority side with about the same difference, only with a higher score. DBSCAN and NearMiss-2 helps out the best on the negative-minority side but still fall 2-18 points short of back-translation.

Guideline: For corpora with < 5 000 minority sentences, apply one-round back-translation through a high-resource language pair (e.g., EN-JAP) if the minority is positive, or apply contextual replacements as it it offers the largest return while still not taking too much time.

7.2.2 How stable are the recommendations across domains?

Although the experiments target Amazon product reviews, the decision rules hinge on two variables that transfer, absolute minority count and lexical redundancy of the majority. The positive Amazon product reviews had on average 73.5 words per review, while the negative reviews had on average 79.5 words per review. Domains with shorter texts (tweets) hit the redundancy ceiling sooner; domains with jargon-rich prose (legal opinions) hit it later.. Practitioners should therefore interpret the 25 000-instance threshold as an order-of-magnitude guide: for tweets it might drop to 10 000; for highly technical writing it might rise to 40 000. Pilot studies on 10–20 % of the corpus can quickly validate which regime applies by plotting a mini learning curve with and without back-translation.

Operational cheat sheet Corpus size Minority share Priority action Secondary action In summary, the grid distils into a simple management narrative: invest annotation resources until the minority class covers roughly twenty-five thousand sentences; augment by back-translation or contextual augmentation below that line; switch to contextual masks in the mid-range; and retire resampling altogether on near-balanced, quarter-million-sentence corpora. These guidelines translate raw accuracy tables into actionable playbooks for engineering teams charged with production sentiment analytics.

Table 7.1: Resampling Recommendations Based on Dataset Size and Class Skew

Dataset Size	**Minority Class Proportion**	**Recommended Oversampling**	**Recommended Undersampling**
$\leq$ 50K	$\leq$ 15%	Back-translation, Contextual masks	NearMiss-1
50K – 250K	15–35%	Contextual masks	NearMiss-1 (for faster training)
$\geq$ 250K	$\geq$ 35%	Skip resampling	Consider NearMiss-1 only to reduce training time

7.3 Methodological limitations and caveats

While the empirical grid delivers a detailed map of sampler behavior, it is still a map drawn from a particular vantage point—English product reviews, a single recurrent architecture, and deterministic preprocessing. This section analyses the methodological boundaries within which the conclusions hold, grouping caveats under five headings: annotation noise, architectural choice, evaluation design, domain specificity, and reproducibility scope. Clarifying these boundaries not only tempers over-generalisation but also motivates the future-work agenda articulated in Chapter 8.

7.3.1 Annotation noise and label granularity

Star-rating heuristics. The binary labels in all thirty train-sets are derived from five- and four-star and one- and two-star reviews (positive vs. negative) and from star-bucket thresholds for balanced test data. Such heuristics ignore neutral sentiment and collapse mixed reviews into a forced dichotomy. False negatives flagged in the qualitative audit frequently involve mixed polarity (“Stunning screen, dismal battery”) that the star label classifies as negative, whereas a human might deem it neutral or ambivalent. Sampling cannot fix this misalignment; indeed, oversampling might amplify it by duplicating label-noisy sentences.

Impact on findings

Because label noise inflates the apparent irreducible error, some plateau effects may reflect annotation limits rather than true model saturation. If provided noise-free sentiment judgements, accuracy ceilings could shift upward, and the headroom argument would need recalibration. Until such data are available, reported accuracy ceilings should be interpreted as "best possible under star-rating labels," not as an upper bound on human-level sentiment understanding.

7.3.2 Architectural constraints

Single-layer recurrent encoder. The model's representational bottleneck is a 128-unit LSTM whose hidden state, even with attention pooling and a four-layer MLP head, is small compared to a 110 M-parameter language model. Transformer encoders can represent far more lexical diversity and longer dependencies; they may, therefore, continue to benefit from augmentation at corpus sizes where the LSTM saturates. Preliminary pilot runs with a fine-tuned bert-base hinted at a 0.5-point jump beyond the LSTM plateau in the 1 M grids, but a full Transformer sweep was outside scope.

Hyper-parameter freeze. Learning rate, batch size, and epoch count were tuned once and frozen. Some samplers converge faster (undersampling) or slower (oversampling). A sampler-specific learning-rate schedule could shift local rankings, although the main scale-dependent patterns are unlikely to invert.

7.3.3 Evaluation design

Balanced test split. The test sets are 50–50. This choice isolates training imbalance but blunts insight into post-deployment prior shift. In a live e-commerce stream, positive reviews may dominate 9:1; a model trained on back-translated negatives could still falter if it over-predicts rare complaints. Macro-F1 mitigates this risk by being prior invariant, yet the thesis headlines Accuracy for its interpretability. A complementary experiment with skewed test sets would enrich external validity.

7.3.4 Domain specificity

Consumer-review language. Amazon reviews skew toward descriptive adjectives and delivery logistics. Twitter is sarcasm-rich and brevity-driven; StackOverflow

comments exhibit jargon and code snippets. Sampling algorithms that thrive on redundancy (DBSCAN, K-Means) might over-prune terse tweets or under-prune verbose forum posts. Likewise, back-translation's utility depends on parallel-corpus quality, which varies across domains and language registers.

7.3.5 Reproducibility scope

Single seed. Deterministic execution improves comparability but hides variance. Broader Monte-Carlo studies could reveal whether NearMiss-2's tiny win in the 100 000 datasets at 45 % minority are artefacts or robust advantages.

7.3.6 Summary

These limitations do not negate the empirical patterns; they bound their domain of applicability. To claim that contextual masking always beats back-translation or that resampling is futile on every million-sentence corpus would overstep the evidence. The safe reading is narrower: under English consumer-review language, with recurrent models and balanced evaluation, back-translation reigns in tiny positive minority extremes, contextual masks dominate the mid-range and negative minority extremes, and marginal returns fade near balance. Extending or challenging that narrative requires new architecture, new domains, and multi-seed variance studies.

7.4 Deployment and closing reflections

The patterns and practical guidelines outlined have direct consequences for real-world deployment, both technical and organizational. This final segment distils those consequences into three lenses: operational rollout, governance and compliance, and research trajectory. It concludes with a concise reflection on the thesis's overall contribution.

7.4.1 Operational rollout: marrying data strategy with platform constraints

A sentiment engine does not live in a vacuum; it plugs into customer-support dashboards, brand-monitoring pipelines, or recommender funnels. Each downstream ap-

plication places distinct constraints on retrain frequency, latency, and interpretability. The size–skew grid suggests a tiered roll-out strategy:

Table 7.2: Recommended Resampling Strategies by Deployment Tier

Tier	Strategy and Rationale
Start-up Tier (50K reviews, extreme skew)	Use **back-translation** for its high impact on minority recall and negligible implementation cost. Retraining can be performed weekly on modest hardware. Neural machine translation costs are justified by the performance gain relative to manual labeling.
Growth Tier (100K–250K reviews, moderate skew)	**Contextual masking** offers the best trade-off between quality and efficiency, yielding up to seven percentage points of accuracy gain. Daily retraining remains feasible within typical compute budgets. Optionally apply **NearMiss-1** to reduce data volume if GPU usage is cost-sensitive.
Enterprise Tier (1M reviews, near balance)	Resampling has limited effect on accuracy at this scale. Training can proceed on the **raw corpus**. If infrastructure cost is a concern, apply **NearMiss-1** to reduce volume. With accuracy nearing saturation, engineering effort is better allocated to **domain adaptation** or more granular **label taxonomies** (e.g., aspect-level sentiment).

These prescriptions align model life-cycle management with the data realities quantified in Chapters 4–6, enabling data teams to forecast resource needs and deliverables with finer granularity than generic "use SMOTE" folklore.

7.5 Deployment and closing reflections

The patterns and practical guidelines outlined have direct consequences for real-world deployment, both technical and organizational. This final segment distils those consequences into three lenses: operational rollout, governance and compliance, and research trajectory. It concludes with a concise reflection on the thesis's overall contribution.

7.5.1 Research trajectory and closing reflections

The grid shows that resampling, once a primary lever for fixing skew, becomes secondary when data volumes soar and neural encoders mature. The logical research pivot is:

- **Beyond binary polarity:** Many practical applications depend on fine-grained or aspect-level sentiment classification. In these settings, class imbalance is often more severe, for example, complaints about delivery delay might represent less than 3% of a large review corpus. The sampling strategies explored in this thesis must be re-evaluated under such multi-label or hierarchical taxonomies.

Contribution statement:

1. This study aps nine classical sampling techniques across a 30-cell size–skew grid using modern text embeddings,
2. This study quantifies an absolute-minority threshold that helps determine when augmentation, pruning, or no resampling is optimal,
3. This study provides reproducible, script-only artifacts that balance verifiability with data privacy, and
4. The study grounds the findings in both ethical considerations and operational constraints.

7.6 Threats to Validity

Despite the thorough experimental design and thorough comparative analysis presented in this thesis, several factors may influence the validity and generalizability

of the findings. This chapter outlines potential threats to validity using standard categories from empirical research: internal, external, construct, and conclusion validity.

7.6.1 Internal Validity

Internal validity refers to the extent to which the observed outcomes can be attributed to the sampling methods themselves, rather than other uncontrolled factors.

All downsampling pluss adasyn used the same fixed seed,42, to ensure consistency across methods and dataset configurations. While this improves reproducibility, it also limits exploration of variability due to random initialization. Some methods might prove to behave differently across multiple runs if the seed is changed.

For all downsampling methods involving neighborhood-based calculations (e.g., ENN, NearMiss variants), the n_neighbors parameter was fixed at 5 across all datasets and skew configurations. While this ensures consistency and comparability, it may not be optimal for each method or scenario. As such, some methods may underperform not due to inherent limitations, but due to underperforming hyperparameters.

Each experiment conducted test data from the same dataset where the training data originated from, split 80% for training and 20% for testing. While this split is reasonable and commonly used, alternative sources for a test set could lead to small shifts in performance rankings.

7.6.2 External Validity

All evaluations were performed on the Amazon Reviews for Sentiment Analysis dataset, which contains binary sentiment labels (positive and negative) applied to English-language product reviews. Although this dataset is reasonable for sentiment classification, the behavior seen for various sample techniques might not apply to:

- Multi-class sentiment problems
- Other domains such as political opinion, news comments, or clinical tales
- Languages with various linguistic structures, especially those with less access to NLP tools

The performance trends seen in this study may also be affected by positive features of the dataset even if these characteristics have not been confirmed. Not only is the Amazon Reviews corpus used in this study large and well-organized regarding the polarity of emotion, but it may also have other characteristics that help resampling techniques work better. These qualities include a generally uniform writing style, clear class distinction, or little label noise. It is possible that some augmentation or downsampling approaches will benefit disproportionately if the data happens to exhibit these qualities. This will result in higher performance than what would be seen in domains that are more variable or noisy. A "jackpot" situation is one in which the structure of the data matches exceptionally well with the assumptions and strengths of the sample methods that were tested. In this respect, the dataset may constitute a "jackpot" scenario.

The performance of the model may be sensitive to the particular data split that is employed, which is another crucial issue to take into account. Due to the fact that every experiment of each size in this work is based on the same test data, it is possible that the relative performance of sampling methods could be influenced by favorable or unfavorable distributions of edge instances or ambiguous samples. In situations when there is a significant imbalance between the classes, even minute variations in the selection of reviews to include in the test set can have an effect on the ranking of the techniques. Despite the fact that the stratification technique helps reduce randomness, the findings are still susceptible to chance because there were no repeated runs. Subsequent work across several datasets and random splits should repeat the experimental design; the ideal situation would be to employ several seeds or cross-validation to check for consistency. This will help to make the results more generally applicable.

7.6.3 Construct Validity

Construct validity relates to whether the experimental setup accurately captures what it aims to measure. Model performance is assessed in this thesis mostly by test accuracy, together with class-specific accuracy and F1-scores. These measures, nonetheless, imply same expense for all kinds of misclassification. In practical use, false positives for example mislabeling negative emotion as positive, may be more expensive than false negatives, or the other way around.

Operating at the feature space level, several sampling methods, especially ADASYN, may cause semantic changes in the training data not detected by conventional performance measures. This could influence the fidelity of sentiment signals in ways

not clearly quantifiable by means of accuracy or F1.

7.6.4

Conclusion Validity Conclusion validity addresses whether the drawn inferences are statistically reliable. The results presented are based on single training runs per configuration. No statistical tests (e.g., t-tests or confidence intervals) were applied to determine the significance of observed differences. While large accuracy deltas suggest meaningful differences, ranking-based and delta-based comparisons should be interpreted as indicative, not definitive.

Additionally, model behavior was not examined utilizing uncertainty estimation or robustness tests (e.g., adversarial perturbations), which could expose more variability in performance under non-ideal circumstances.

7.6.5 Summary

The findings of this study have been interpreted in light of the main validity threats and boundary conditions that have been delineated in this chapter. Although the assessment system was planned and carried out, these restrictions should be taken into account when extrapolating findings to other datasets, sectors, or deployment settings. Future study should, if feasible, seek to mitigate these threats by means of repeated runs, larger datasets, and varied assessment criteria.

Chapter 8

Conclusions

This thesis aimed to explore how downsampling and augmentation methods that could help to solve data imbalance in sentiment classification tasks and how those methods operate across various dataset sizes and class skew levels. The work assessed nine resampling techniques by means of a methodical grid of 260 experiments, looking at their impact on test accuracy, class-specific precision, and general classification behavior. This last chapter analyzes prospective avenues for extending the study, reinforces the practical contributions provided, and synthesizes the essential results. It also considers the more general applicability of the findings for practical uses of sentiment analysis in uneven environments.

8.1 Summary of Findings

The experiments conducted in this thesis provide a detailed comparative understanding of how different sampling strategies perform under varying class imbalance and dataset sizes. The findings reveal that no single method dominates in all skews across all sizes, but consistent trends emerge when stratified by data volume and skew. At the 10 000-sample scale, performance was heavily influenced by low amount of data. In extremely skewed settings (5–15% minority), back-translation outperformed other methods, providing the most highest accuracy improvements and notably increasing precision for the minority class. Despite back translations performance, comparatively to the others, its score value wasn't as impressive as it only reached an accuracy of 0.542. Undersampling methods all performed bad at this scale, often removing too much information from an already small dataset, leading to 0.500 accuracy.

At the 100 000-sample scale, a different pattern emerged. Contextual augmentation

became the most effective method across a majority of skew settings. It consistently outperformed both the original unbalanced dataset and other resampling strategies in terms of test accuracy and precision for both classes. The original dataset (i.e., no sampling) remained competitive near balanced class distributions, but underperformed in low-minority scenarios. Back-translation remained strong at extreme skew but was generally surpassed by contextual augmentation at moderate skews. Ranking-based summaries confirmed this: contextual augmentation had the highest average ΔAccuracy and the most first-place finishes at this scale, tied with the original data, but had excluding the original data, it had the best score in half of the skews.

At the 1 000 000-sample scale, the performance differences between methods narrowed significantly. The original dataset often matched or outperformed resampling methods at balanced or near-balanced class distributions. However, contextual augmentation remained beneficial in more skewed conditions, achieving the best overall average performance and the most top-ranked results. Other methods, showed minimal improvements and mostly slight degradation compared to the baseline, likely due to the dataset's already sufficient size.

Across all scales, class-specific precision metrics revealed that oversampling methods generally preserved or improved minority class precision, while undersampling strategies often led to asymmetry where it generally favored the minority class after the majority had been sampled down. Augmentation-based techniques, particularly contextual augmentation, proved more stable across skews and less likely to introduce performance volatility.

This thesis adds to the theoretical and practical use of data sampling techniques for imbalanced sentiment classification in various ways. The work increases knowledge in both practical deployment direction and methodological assessment.

8.2 Contributions

This thesis adds to the theoretical and practical use of data sampling techniques for imbalanced sentiment classification in various ways. The work increases knowledge in both practical deployment direction and methodological assessment.

8.2.1 Emperical Contributions

This thesis conducted a comprehensive, grid-based evaluation of nine sampling methods was conducted in this thesis, which resulted in a total of 260 unique experimental settings when run across nine distinct sampling methods and three dataset sizes (10K, 100K, 1M) and ten class imbalance levels. It showed that augmentation methods consistently outperform undersampling and baseline approaches at mid-scale (100K) and maintain measurable benefits at large scale (1M), particularly under moderate-to-high skew. While undersampling techniques usually do badly when data is scarce, the findings revealed that back-translation is especially successful in small-scale, highly unbalanced settings. It measured the declining returns of resampling techniques as dataset size grows, hence supporting the theory that synthetic balancing helps less for large-scale datasets.

8.2.2 Analytical and Practical Contributions

Introduced a set of practitioner-oriented lookup rules that summarize optimal sampling strategies based on dataset size and minority class proportion. These rules provide actionable guidance for developers working with imbalanced sentiment data.

Developed a framework for method-level comparison, including average ΔAccuracy, win frequency, and ranking analysis, to supplement metric-by-metric performance reporting.

Raised awareness of fairness and interpretability concerns in resampling, particularly regarding the semantic integrity of synthetic examples and the transparency of feature-space operations. Together, these contributions provide both theoretical insight and practical tools for improving sentiment classification performance in imbalanced settings. The thesis offers a replicable framework for evaluating resampling strategies, as well as a concrete decision-making aid for real-world model development.

8.3 Future Work

While this thesis offers a thorough assessment of sample techniques in sentiment categorization, it also paves the way for more research and improvement.

8.3.1 Multi-run Experiments and Statistical Testing

All results in this thesis are based on single training runs per configuration. To improve the robustness of conclusions, future work should incorporate multiple training seeds per setting to capture model variance and enable statistical testing (e.g., t-tests or confidence intervals) across methods. This would provide stronger evidence for the observed performance differences and help distinguish true method advantages from stochastic fluctuations.

8.3.2 Hyperparameter Optimization for Sampling Methods

Sampling methods involving neighborhood-based heuristics (e.g., NearMiss, ENN, DBSCAN) were evaluated with a fixed configuration, specifically n_neighbors = 5. While this choice was made for consistency, these methods could perform better if tuned based on the dataset and skew level. Future work could explore automated or adaptive parameter tuning to more accurately reflect each method's potential.

8.3.3 Extension to Multi-Class and Multilingual Settings

This paper deals only with binary sentiment categorization in English. Many real-world applications, include multi-class sentiment, fine-grained emotions, or non-English language. Future research should investigate if the patterns found here apply to more complicated classification schemes and other languages, especially in low-resource environments where augmentation might act differently.

8.3.4 Integration with Transformer Architectures

A standard LSTM architecture was used for all studies. Although this decision guarantees lower computing needs and clarity, many contemporary sentiment analysis systems depend on transformer-based models such as BERT or RoBERTa. Whether the relative advantages of sampling techniques continue with pretrained contextual embeddings or fine-tuned transformer models still an outstanding issue.

8.3.5 Deeper Fairness and Semantic Evaluation

Finally, the thesis raised preliminary concerns about semantic fidelity and fairness, particularly for synthetic data. Future work could apply qualitative error analysis, interpretability tools (e.g., LIME, SHAP), or bias audits to investigate how sampling strategies affect model decision boundaries, linguistic patterns, and fair treatment of underrepresented sentiment classes.

8.4 Closing Remarks

This thesis aimed to tackle a frequent but ongoing problem in machine learning: dealing with class imbalance in sentiment analysis. By means of a systematic and repeatable assessment of sample methods across data sizes and skew levels, it shows that careful resampling, particularly via text-based augmentation can produce noticeable improvements even in current NLP processes.

Beyond the empirical findings, the study emphasizes the necessity of technique selection based on data characteristics, as well as the trade-offs between performance, interpretability, and fairness. Its objective is to establish a connection between academic experimentation and practical deployment. As data science continues to expand into domains where imbalanced and nuanced data is the norm, such as public discourse, misinformation detection, and mental health, the necessity for robust, transparent, and adaptable classification strategies becomes more pressing. The contribution of this thesis to that endeavor is twofold: it offers a framework for making informed, data-driven decisions in imbalanced classification tasks and provides a comprehensive comparison of methods.

Bibliography

[1] Chumphol "Bunkhumpornpat, Krung Sinapiromsaran and Chidchanok" Lursinsap. '"Safe-Level-SMOTE: Safe-Level-Synthetic Minority Over-Sampling TEchnique for Handling the Class Imbalanced Problem"'. In: *"Advances in Knowledge Discovery and Data Mining"*. Ed. by Thanaruk "Theeramunkong et al. "Berlin, Heidelberg": "Springer Berlin Heidelberg", 2009, "475–482". ISBN: "978-3-642-01307-2".

[2] Nitesh V. "Chawla et al. '"SMOTEBoost: Improving Prediction of the Minority Class in Boosting"'. In: *"Knowledge Discovery in Databases: PKDD 2003"*. Ed. by Nada "Lavrač et al. "Berlin, Heidelberg": "Springer Berlin Heidelberg", 2003, "107–119". ISBN: "978-3-540-39804-2".

[3] Sosuke" "Kobayashi. '"Contextual Augmentation: Data Augmentation by Words with Paradigmatic Relations"'. In: *"Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)"*. Ed. by Marilyn "Walker, Heng Ji and Amanda" Stent. "New Orleans, Louisiana": "Association for Computational Linguistics", 2018, "452–457". DOI: "10.18653/v1/N18-2072". URL: https://aclanthology.org/N18-2072/.

[4] Majdi Omar Alali. 'Optimal AI through Minimal Data: Enhancing Sentiment Analysis with Data Diversity for Norwegian'. MA thesis. Oslo Metropolitan University, 2024.

[5] Gustavo E. A. P. A. Batista, Ronaldo C. Prati and Maria Carolina Monard. 'A study of the behavior of several methods for balancing machine learning training data'. In: *SIGKDD Explor. Newsl.* 6.1 (June 2004), pp. 20–29. ISSN: 1931-0145. DOI: 10.1145/1007730.1007735. URL: https://doi.org/10.1145/1007730.1007735.

[6] N. V. Chawla et al. 'SMOTE: Synthetic Minority Over-sampling Technique'. In: *Journal of Artificial Intelligence Research* 16 (June 2002), pp. 321–357. ISSN: 1076-9757. DOI: 10.1613/jair.953. URL: http://dx.doi.org/10.1613/jair.953.

[7] Sergey Edunov et al. *Understanding Back-Translation at Scale*. 2018. arXiv: 1808.09381 [cs.CL]. URL: https://arxiv.org/abs/1808.09381.

[8] Martin Ester et al. 'A density-based algorithm for discovering clusters in large spatial databases with noise'. In: *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining*. KDD'96. Portland, Oregon: AAAI Press, 1996, pp. 226–231.

[9] Salvador García et al. 'A study of statistical techniques and performance measures for genetics-based machine learning: accuracy, parsimony, and efficiency'. In: *Soft Computing* 13.10 (2009), pp. 959–977.

[10] Haibo He and Edwardo A. Garcia. 'Learning from Imbalanced Data'. In: *IEEE Transactions on Knowledge and Data Engineering* 21.9 (2009), pp. 1263–1284. DOI: 10.1109/TKDE.2008.239.

[11] Haibo He et al. 'ADASYN: Adaptive synthetic sampling approach for imbalanced learning'. In: *2008 IEEE International Joint Conference on Neural Networks (IEEE World Congress on Computational Intelligence)*. 2008, pp. 1322–1328. DOI: 10.1109/IJCNN.2008.4633969.

[12] Muhammad Mujahid et al. 'Data oversampling and imbalanced datasets: an investigation of performance for machine learning and feature engineering'. In: *Journal of Big Data* 11 (2024). DOI: 10.1186/s40537-024-00943-4.

[13] Ivan Tomek. 'Two modifications of CNN'. In: *IEEE Transactions on Systems, Man, and Cybernetics* 6.6 (1976), pp. 769–772.

[14] Dennis L. Wilson. 'Asymptotic Properties of Nearest Neighbor Rules Using Edited Data'. In: *IEEE Transactions on Systems, Man, and Cybernetics* SMC-2.3 (1972), pp. 408–421. DOI: 10.1109/TSMC.1972.4309137.

[15] Qizhe Xie et al. *Unsupervised Data Augmentation for Consistency Training*. 2020. arXiv: 1904.12848 [cs.LG]. URL: https://arxiv.org/abs/1904.12848.

[16] Show-Jane Yen and Yue-Shi Lee. 'Cluster-based under-sampling approaches for imbalanced data distributions'. In: *Expert Systems with Applications* 36.3, Part 1 (2009), pp. 5718–5727. ISSN: 0957-4174. DOI: https://doi.org/10.1016/j.eswa.2008.06.108. URL: https://www.sciencedirect.com/science/article/pii/S0957417408003527.

[17] J. Zhang and I. Mani. 'KNN Approach to Unbalanced Data Distributions: A Case Study Involving Information Extraction'. In: *Proceedings of the ICML'2003 Workshop on Learning from Imbalanced Datasets*. 2003.