How Dimensionality Reduction affects Sentiment Analysis NLP Tasks: An Experimental Study

L. Akritidis[1], P. Bozanis[1]

[1]School of Science and Technology, International Hellenic University

18th International Conference on Artificial Intelligence Applications and Innovations (AIAI)

Aldermar Knossos Royal 5★ Beach Resort, Crete, Greece

17 – 20 June, 2022

Dimensionality Reduction

- A well-known technique for limiting the feature space size and discovering latent meaningful variables in the input data.
- Particularly valuable when the raw data is sparse and its processing by machine learning algorithms becomes computationally very expensive.
- The typical text vector representations are in general very sparse and high dimensional.
- The traditional bag-of-words model (e.g. tf-idf, count vectors, etc.) leads to very long vector representations with a huge number of zeros.

Sentiment Analysis

- It refers to a collection of text classification methods that identify the polarity of the user opinions in blog posts, product reviews, user comments, tweets, etc.
- The opinion polarities may be binary (positive or negative), ternary (e.g., positive/negative/neutral), or fall into a specific range (for example, ratings within 1-5 or 1-10 scale).
- However, since text is naturally very sparse, training classification models is often intractable.
- Therefore, the importance of dimensionality reduction becomes crucial in such problems.

An experimental study

- Sentiment analysis is essentially a supervised classification problem that is usually solved by applying machine learning models on text input data.
- In this paper we studied the impact of dimensionality reduction in sentiment analysis classification tasks.
- Through extensive experimentation with traditional algorithms and benchmark datasets, we verify that dimensionality reduction:
  - improves the data preprocessing times and the model training durations.
  - sacrifices only small amounts of accuracy.

Datasets

- Four popular sentiment analysis datasets were used in this study.
- All are publicly available, and have been utilized extensively in the relevant literature for evaluating NLP algorithms.
- Here the classes represent the different opinion polarities.

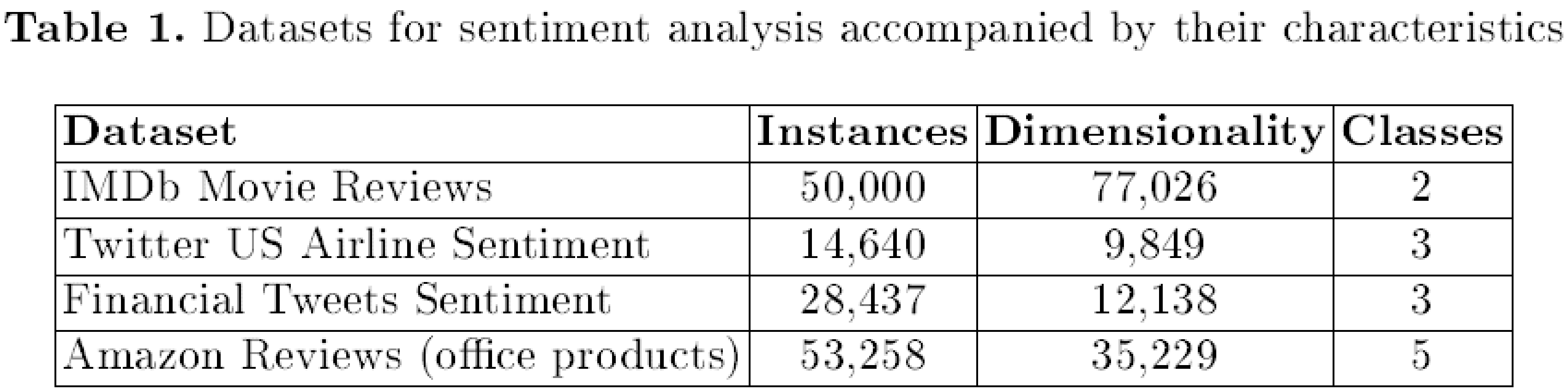


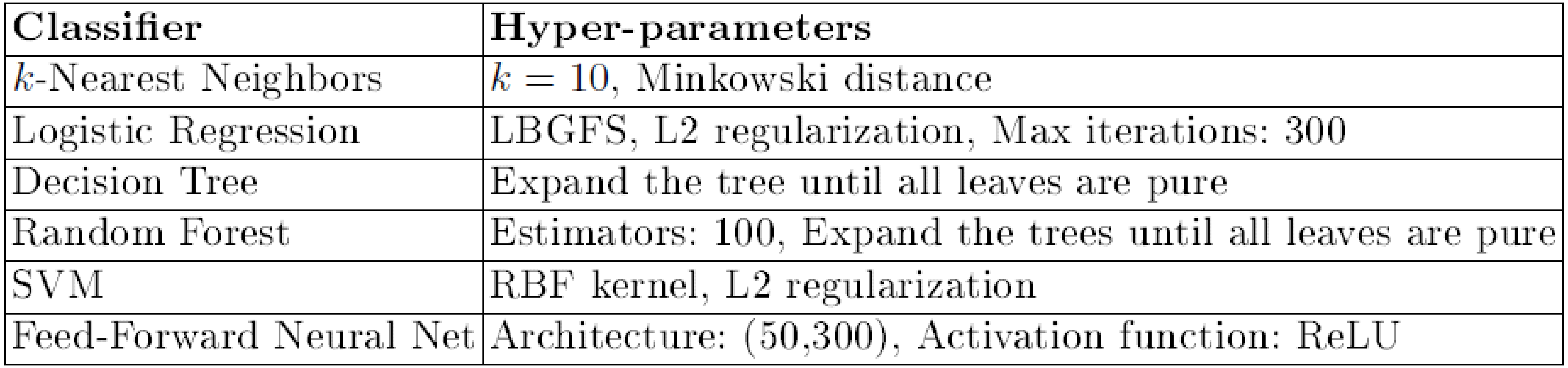
Table 1. Datasets for sentiment analysis accompanied by their characteristics

Dataset	**Instances**	**Dimensionality**	**Classes**
IMDb Movie Reviews	50,000	77,026	2
Twitter US Airline Sentiment	14,640	9,849	3
Financial Tweets Sentiment	28,437	12,138	3
Amazon Reviews (office products)	53,258	35,229	5

Classifiers

- Six classifiers were employed to perform sentiment analysis.
  - Linear models (logistic regression)
  - Non-linear models (ANNs, SVMs with RBF kernel).
  - Tree models (Decision Trees, Random Forests, etc.).
- Deep classifiers (RNNs, LSTMs,…) are currently being examined.

Table 2. Classifiers and hyper-parameters



Classifier	**Hyper-parameters**
k-Nearest Neighbors	$k = 10$, Minkowski distance
Logistic Regression	LBGFS, L2 regularization, Max iterations: 300
Decision Tree	Expand the tree until all leaves are pure
Random Forest	Estimators: 100, Expand the trees until all leaves are pure
SVM	RBF kernel, L2 regularization
Feed-Forward Neural Net	Architecture: (50,300), Activation function: ReLU

Text preprocessing

- The input raw text was converted to lowercase.
- A word-level tokenizer converted each document into a bag of words.
- The stop words were removed.
- The WordNet lemmatizer was subsequently employed to convert each word to its meaningful base form.
- The collection was split to training and test sets by applying a constant ratio of 70%/30% and stratification by class.
- The two sets were individually vectorized by applying the well-known tf-idf transformation.

Truncated Singular Value Decomposition (TSVD)

- TSVD is a variant of Principal Component Analysis (PCA).
- Recall that PCA identifies the principal components by maximizing the variance of the projected data.
  - PCA is not feasible to sparse matrices.
- In contrast, TSVD does not center the data before computing the singular value decomposition.
  - It operates efficiently on sparse matrices.
- TSVD is often known as Latent Semantic Analysis (LSA).

Results (IMDb dataset)

- SVM, LogReg and ANNs are the most accurate classifiers.
  - A x10 reduction **does not** affect the accuracy significantly.
    - Exception: Random Forests and kNN.
  - x100/x1000 reductions → greater loss.
- The running times are surprising.
  - A x10 reduction **decelerates** training.
    - Neural nets are the only exception.
  - x100 reduction → slight acceleration.
  - x1000 reduction → great acceleration.

Dataset	Instances	Dimensionality	Classes
IMDb Movie Reviews	50,000	77,026	2

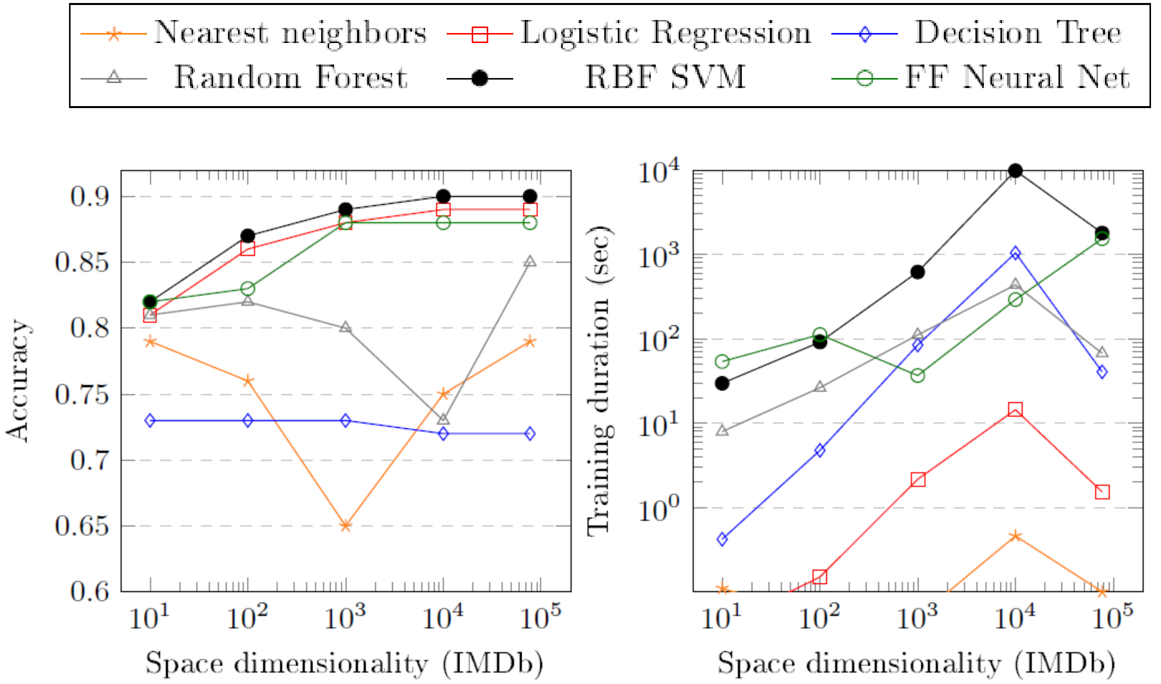


Fig. 1. Accuracy values (left) and training durations (right) of the six classifiers of Table 2 for the IMDb (top) and the Twitter US Airline (bottom) datasets. The horizontal axes are plotted in logarithmic scale, and represent various input spaces of different dimensionalities. In all diagrams, the rightmost markers denote the performance of the algorithms in the original feature spaces; namely, without dimensionality reduction.

Results (Twitter US Airline dataset)

- SVM and Logistic Regression are the most accurate classifiers.
  - A x10 reduction **does not** affect their accuracy significantly.
    - More Exceptions: R. Forests, kNN, D. Trees
  - x100/x1000 reductions → greater loss.
- The running times change similarly.
  - A x10 reduction **decelerates** training.
    - Neural nets are the only exception.
  - x100 reduction → slight acceleration.
  - x1000 reduction → great acceleration.

Dataset	Instances	Dimensionality	Classes
Twitter US Airline Sentiment	14,640	9,849	3

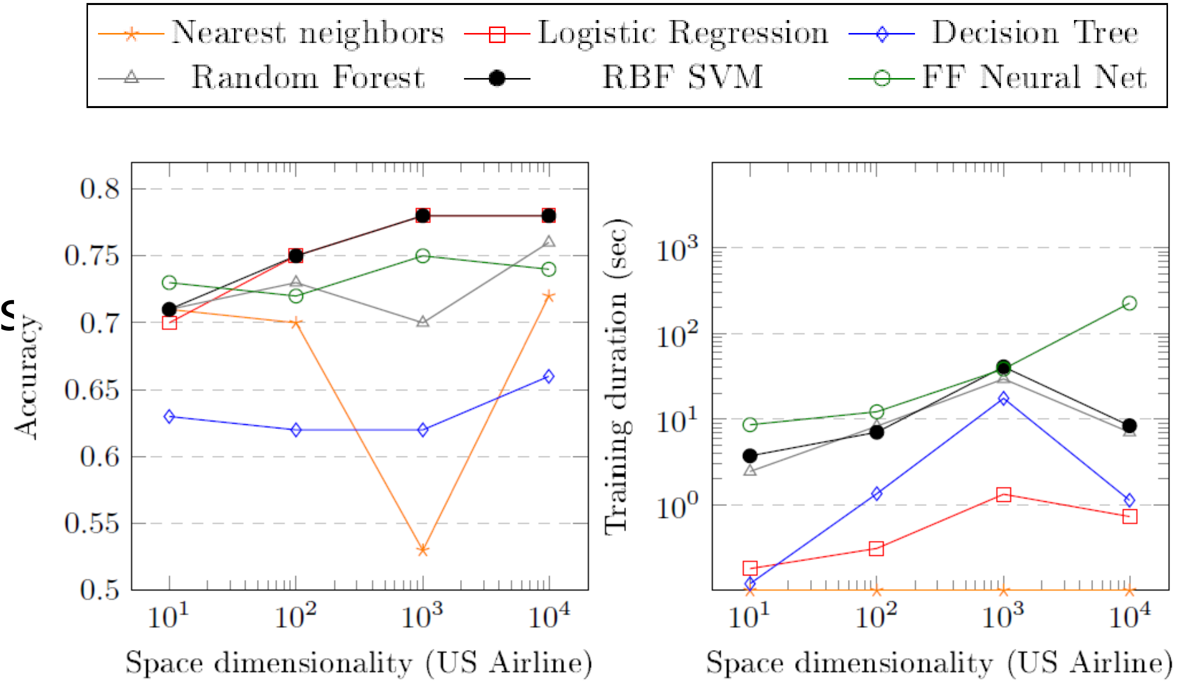


Fig. 1. Accuracy values (left) and training durations (right) of the six classifiers of Table 2 for the IMDb (top) and the Twitter US Airline (bottom) datasets. The horizontal axes are plotted in logarithmic scale, and represent various input spaces of different dimensionalities. In all diagrams, the rightmost markers denote the performance of the algorithms in the original feature spaces; namely, without dimensionality reduction.

Results (Financial Tweets Dataset)

- ANN and the two tree classifiers are the most accurate classifiers.
  - Here, a x10 reduction **affects** their accuracy significantly.
    - Exception: SVM
  - x100/x1000 reductions → greater loss.
- The running times change similarly.
  - A x10 reduction **decelerates** training.
    - Neural nets are the only exception.
  - x100 reduction → slight acceleration.
  - x1000 reduction → great acceleration.

Dataset	**Instances**	**Dimensionality**	**Classes**
Financial Tweets Sentiment	28,437	12,138	3

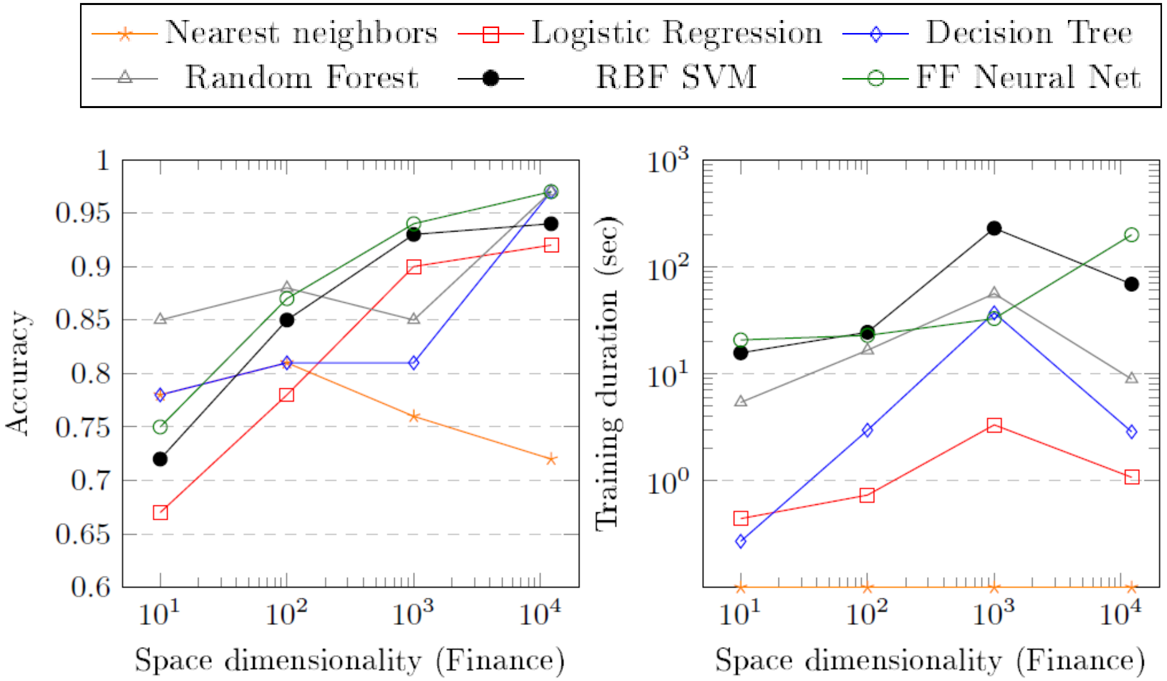
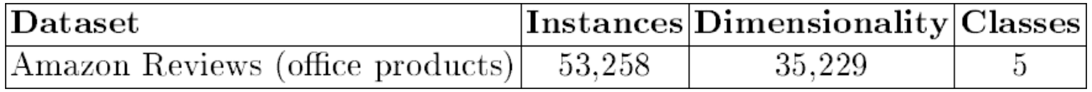


Fig. 2. Accuracy values (left) and training durations (right) of the six classifiers of Table 2 for Financial Tweets (top) and Amazon Product Reviews (bottom). The horizontal axes are plotted in logarithmic scale, and represent various input spaces of different dimensionalities. In all diagrams, the rightmost markers denote the performance of the algorithms in the original feature spaces; namely, without dimensionality reduction.

Results (Amazon Product Reviews)

- Logistic Regression and SVM are the most accurate classifiers.
  - Here, a x10 **and** a x100 reduction **does not** affect their accuracy.
    - Exception: Decision Trees
  - x1000 reductions → significant loss.
- The running times change similarly.
  - A x10 reduction **decelerates** training.
    - Neural nets are the only exception.
  - x100 reduction → slight acceleration.
  - x1000 reduction → great acceleration.



Dataset	Instances	Dimensionality	Classes
Amazon Reviews (office products)	53,258	35,229	5

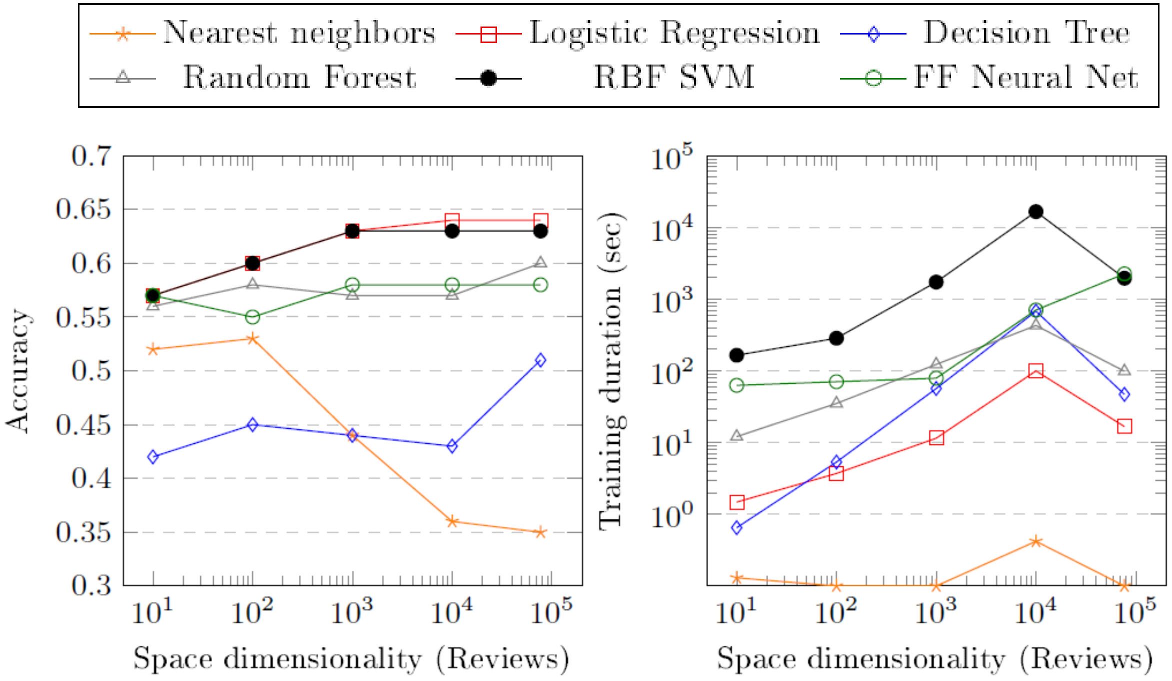


Fig. 2. Accuracy values (left) and training durations (right) of the six classifiers of Table 2 for Financial Tweets (top) and Amazon Product Reviews (bottom). The horizontal axes are plotted in logarithmic scale, and represent various input spaces of different dimensionalities. In all diagrams, the rightmost markers denote the performance of the algorithms in the original feature spaces; namely, without dimensionality reduction.

Interpretation and Conclusions (1)

- The experiments in all four datasets demonstrated that **conducting a limited reduction in the dimensionality of the input vector space is not beneficial.**
- **More specifically, reducing the dimensions by one order of magnitude renders the classifiers both slower and less effective.**
- The feed-forward ANNs are the only exception to this rule.

Interpretation and Conclusions (2)

- On the other hand, **reducing the dimensions of the vector space by two orders of magnitude has a small to moderate impact in both the model training durations and accuracies.**

Interpretation and Conclusions (3)

- The aggressive dimensionality reduction (e.g., input spaces of just 10 features, or reduced by three orders of magnitude) leads to:
  - significant, but not fatal accuracy losses.
  - substantially improved training times
    - especially for the deep ANN and the non-linear SVM classifiers.
- More specifically, a decrease by at least one order of magnitude in model training durations is achieved.

Future work

- There is still a lot of research and experiments to conduct.
- Additional dimensionality reduction techniques.
- Feature engineering methods (e.g. feature selection).
- Study of the behavior of several state-of-the-art deep-learning architectures:
  - Recurrent Neural Nets.
  - Long-Short Term Memory units.
  - Transformers
  - Many more!

Thank you for watching

Questions?