Machine learning method utilization for Lithuanian company clustering*

Eimantas Zaranka[1,2,*] and Dovilė Kuizinienė[1,2]

[1] *Vytautas Magnus University, Universiteto str. 10 53361 Akademija, Lithuania*
[2] *Centre of Applied Research and Development, Universiteto str. 10 53361 Akademija, Lithuania*

Abstract

Effective understanding and interpretability of the economic landscape of a country requires an identification of all business activities. When companies declare their activities, they are legally required to declare only their main activity, often omitting secondary operations. This limitation leads to an incomplete representation of the true economic realities. Lithuania utilized the NACE based economic activities classification system, which consists of six digits. The first four digits denote broad industry categories, while the remaining two digits provide a finer national-level classification. This paper aims to apply clustering methods for enterprise description in order to identify the economic activities of Lithuanian enterprises. The dataset used in this paper consists of 28 350 enterprise descriptions, sourced from the publicly available business directory "Rekvizitai.lt." The initial experiments showed that satisfactory clustering results can be achieved using LaBSE and Word2Vec embeddings in conjunction with K-means, Gaussian Mixture Model, Agglomerative clustering and Mean Shift. These results suggest that clustering techniques can be used on enterprise descriptions to capture patterns and segment them. To further refine the clustering results a hyperparameter search is planned to optimize model performance.

Keywords

Clustering, NACE, Enterprise Activities

1. Introduction

Identification of enterprise activities code is a complex task that serves a crucial task in real world applications. Activity codes are used in various political and economic organizations' daily tasks in order to understand tendencies, tax strategies, developing various social and economic programs, understanding possible partnerships and competitors and in general to understand distribution of companies in various industry sectors.

NACE classification system was established in 1970s, this system serves as a backbone for individual countries classification system [1]. The activities notation consists of six symbols, where the first 4 symbols must match between European Union members and is part of NACE, where final two numbers a designated for individual countries needs.

The objective of this research is to apply clustering algorithms in order to segment enterprises by their activity codes utilizing enterprise descriptions. To achieve the objective, the following tasks are set: create embeddings utilizing *BERT*, *LaBSE* and *Word2Vec*, complete feature compression and selection utilizing *PCA*, and *UMAP*, train hard clustering and soft clustering models, compare the results and provide insights on clustering algorithms applicability for enterprise segmentation.

IVUS2025: Information Society and University Studies 2024, May 15, Kaunas, Lithuania
[1*] Corresponding author.
eimantas.zaranka@vdu.lt (E. Zaranka); dovile.kuiziniene@vdu.lt (D. Kuizinienė)
0000-0002-8133-0698 (D. Kuizinienė);
© 2025 Copyright for this paper by its authors. Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).

CEUR-WS.org/Vol-4213/paper21.pdf

CEUR Workshop Proceedings ceur-ws.org ISSN 1613-0073
published 2026-06-12

2. Literature Review

Literature review in this section covers two main ideas behind text clustering: embeddings creation and application of clustering algorithms for said purpose.

Researchers [2] conducted experiments to investigate how different *LLMs* textual embeddings influence the clustering accuracy of text datasets. Authors observed a positive influence of *LLMs* embeddings at capturing an underlying structures in languages and improvements in cluster purity. The research showed that *BERT* has the most significant improvements compared to other embedding strategies and provides the smallest overhead. On the other hand, the research showed that utilization of the *LLMs* embeddings requires careful consideration in text preprocess step in order to avoid the loss of essential information.

Article [3] objective is to segment industry sectors based on *NACE* groups. Authors formulated a couple of hypotheses: businesses can be segmented based on their innovation and similar activities; enterprises that introduce new or improved products are very similar. The dataset consisted of Polish central statistical department data that are in range between 2016 – 2018. The suggested approach is utilization of cosine angle between observations and *Ward* algorithms to evaluate the results Rand Index is used. Authors noticed that best segmentation results are achieved in three clusters: new products, new business processes and combination of both.

Research [4] conducted a statistical industry classification utilizing *K-Means* and multilevel clustering algorithms. Authors introduced mathematical clustering improvement in order to increase the accuracy of clustering. The suggested clustering improvements consist of top-down approach, bottom-up approach and relaxation. It is observed that combining all three methods the results have increased. According to the authors, the improvement of clustering algorithms has many nuances and even the smallest change in algorithms makes a big difference between various experimentation results.

Authors [5] conducted an industry sector classification where data was separated into 4 different segments. This research allowed authors to interpret different *NACE* levels based on selected companies. The used dataset consisted of companies belonging to groups 10-33 excluding group 19. During the research it was noticed that even after successful segmentation it is impossible to differentiate between business differences that belong to the same *NACE* group in order to complete even further segmentation.

The review showed that LLMs embeddings are outperforming simple text vectorization due to ability of capturing inner text structures. The use of the embeddings shows a positive improvement in cluster purity. Regarding clustering, it is noticed that clustering algorithms is the appropriate choice.

3. Methodology

In this section, all methods used in embeddings creation, cluster model training and evaluation methods are described.

Word2Vec is a word embeddings technique that is based on neural networks. Method captures semantic relationships between words, representing them as dense vectors in a continuous space. The two architectures can be utilized: Skip-gram and Continuous Bag of Words (CBOW). The learned word vectors preserve similarities that can be applied in various NLP tasks [6].

BERT is an open source, transformer based deep learning model designed for natural language understanding. The transformer architecture allows model to understand the context of long sequences based on the context that words belong to [7].

LaBSE is an improved *BERT* model that uses a dual encoder approach in order to get a more precise and contextual vector representation of a text. The sentence representations are created utilizing different languages that later are mapped into a shared embedding space [8].

PCA is a dimensionality reduction technique that transforms high-dimensional data into a lower-dimensional space while preserving patterns in a data. It does this by standardizing the initial values,

calculating principal components from a covariance matrix and fixing the new components utilizing main component axes [9].

UMAP is a dimensionality reduction technique that is based on graphs and places data in a lower dimensional space. The algorithm works in two steps. First, a high dimensional graph representation of the data is obtained using a fuzzy simplicial complex and then by optimizing the graph into lower-dimensional graphs in order to keep the structure as close as possible to the original [10].

K-Means is an unsupervised learning algorithm that groups data without the labels into different clusters by first initializing cluster centers at random, then assigning observations to the cluster utilizing Euclidean distance by formula

$$d(e_1, e_2)=\sqrt{\sum_{i=1}^{n} e_{2i}\text{-}e_{1i}}\,, \tag{1}$$

where e_1 – observation, e_2 – cluster center, n – observation feature number.

Then check if there were changes in the clusters and if so, recalculating the new cluster centers and recalculating the assignments [11].

GMM is a probabilistic clustering algorithm that assumes data is generated from multiple Gaussian distributions. This method assigns probabilities, allowing points to belong to multiple clusters. It is widely used in density estimation, anomaly detection, and speaker identification, particularly when clusters have different shapes and variances [12].

Agglomerative clustering is a hierarchical clustering method that iteratively merges the closest data points or clusters based on a linkage. The method does not require specifying the number of clusters beforehand and produces a dendrogram, which helps in selecting an optimal cut-off point. Most commonly the Euclidean distance is used in order to calculate the distance between two clusters [13].

BIRCH is a clustering algorithm mostly used for large datasets. First the algorithm prepares a summary of a dataset that is later used in the clustering process. The algorithm is only suitable for numerical values that can be represented in a Euclidean space, for that reason the categorical features are not suitable [14].

Mean Shift is a density-based clustering algorithm that iteratively shifts data points toward the densest regions in feature space. Method does not require the number of clusters to be specified and can discover arbitrarily shaped clusters. Mean Shift can be computationally expensive for high-dimensional data [15].

The Silhouette Score is a metric used to evaluate clustering quality by measuring how similar each data point is to its own cluster compared to other clusters. It ranges from -1 to 1, where a higher value indicates well-separated and cohesive clusters [16]. The Silhouette score is defined as:

$$S=\frac{b\text{-}a}{max\ (a,b)}, \tag{2}$$

where S – Silhouette score, a – average intra-cluster distance, b – average inter-cluster distance.

4. Dataset

4.1. Dataset Description

Dataset that is used in this research consists of Lithuanian enterprise descriptions from business directory "Rekvizitai.lt", companies reference code and declared activity code from “Sodra”. The original dataset is consisted of 240 192 observations from which 147 125 observations were nonempty and only 31 835 had an enterprise description that consisted of more than just a keywords, check Table 1 for quantitative statistics.

Table 1
Quantity of original dataset

Observation count	Non-Empty description counts	Description counts
240 192	147 125	31 835

4.2. Dataset Preparation

Data preparation is conducted in seven steps, note that the preparation steps order is important to correctly prepare the dataset:

1. Web links and emails removal.
2. Punctuation marks removal.
3. Letters decapitalization.
4. Numeric symbol removal.
5. Name of the enterprise removal.
6. Additional spaces removal.
7. Removal of stop words.

Due to the complexity of a dataset, the stop words removal is not a straightforward task. Initial cleaning is done by utilizing *SpaCy* library and a prepared stop words dictionary. It is noticed that this approach is not fully effective and in order to complete full stop words removal based on domain knowledge manual dictionary creation was conducted.

4.3. Data Analysis

In this section, the data analysis, which is conducted on cleaned dataset, is presented. The analysis consists of statistical and textual methods in order to better understand the underlaying tendencies of the dataset.

The analysis of the enterprise description lengths distribution, see Figure 1 (a), shows that there are several outliers in the company descriptions, or rather in the length of the descriptions, which can have a negative impact on the creation of embeddings. Figure 1 (b) shows the distribution of description lengths after removal of outliers. The plot without outliers shows that the longest description consists of ~80 non stop words, while the smallest description is three words. The median description length is 14 words, the third percentile is ~37 words and the first percentile is ~7 words.

The word cloud plot shows 100 most common words in enterprise descriptions, see Figure 2. The most frequently used words in the study text were *didmeninė* (wholesale), *mažmeninė* (retail), *prekyba* (trade), *krovinių* (freight), *žemės ūkis* (agriculture) and *gamyba* (manufacturing). Based on the results, it is assumed that the dataset consists mainly of companies involved in sales, agriculture and freight transport. Other activities such as renting, medicine, consultancy, systems development and renting can also be observed among the major terms.

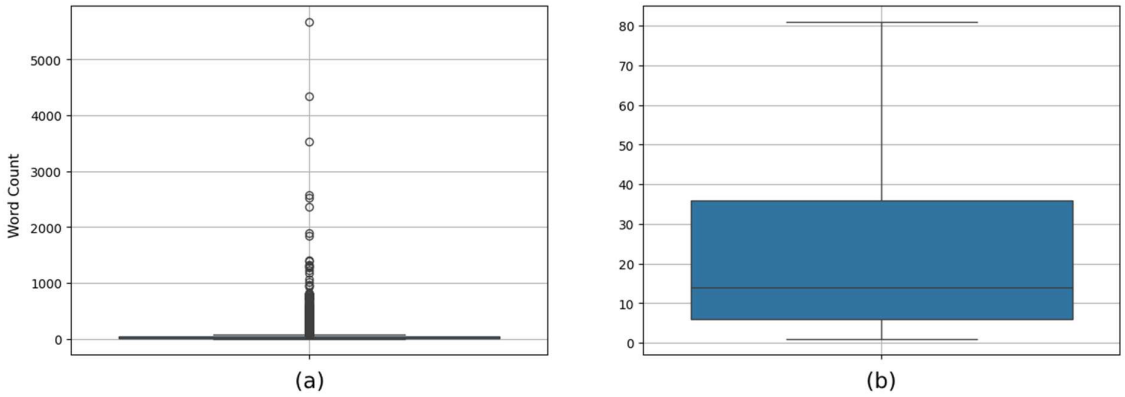

Figure 1: Enterprise description lengths boxplot, where (a) values with outlier, (b) values without outliers

Figure 2: Enterprise descriptions word cloud

The bigram analysis, check Figure 3, shows the same results as in the word cloud, where the most common terms are retail, wholesale, freight and agriculture. The results show that the major clusters of economic activity will be related to retail and wholesale trade, freight transport and agriculture.

An analysis of the distribution of activities, see Figure 4, shows that the largest number of observations is under code G. This code is assigned to enterprises engaged in wholesale and retail trade and repair of motor vehicles and motorcycles. This result is also visible in the analysis of word clouds and bigrams. Immediately after this code, the most frequent activities in the dataset are activities C and M. Activity C is responsible for manufacturing, while activity M is responsible for professional, scientific and technical activities. The least frequent activities are B, D, O and Z, where B is mining and quarrying, D is electricity, gas, steam and air conditioning supply, and O is public administration and defense, compulsory social security. Meanwhile, Z denotes all the remaining enterprises which do not have a separate activity assigned to them.

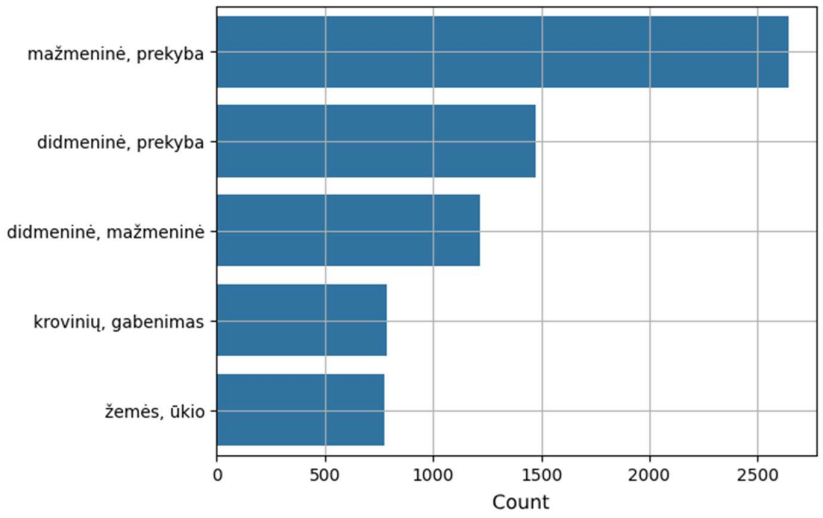

Figure 3: Bigram distribution of enterprise descriptions

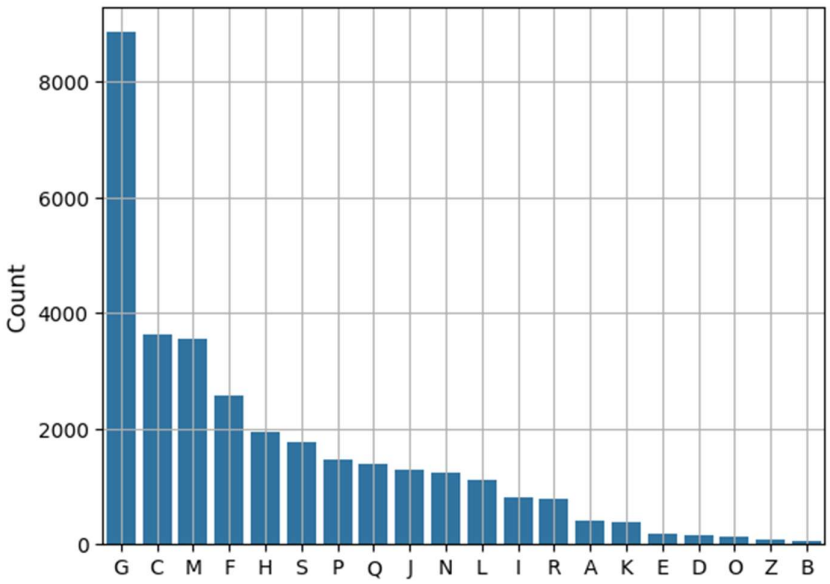

Figure 4: Dataset distribution based on first NACE level

5. Experiments

The experiments are conducted using 5 clustering methods. For hard clustering the *K-Means*, *Gaussian mixture model* and agglomerative clustering, meanwhile for soft clustering mean shift and BIRCH are chosen. Each hard clustering model is trained with 20, 82, 595 and 769 clusters. These values are chosen because they represent 4 different levels of *NACE* codes that are represented in dataset.

5.1. Hard Clustering

Analyzing the results of the hard clustering, see Table 2, *UMAP* is the most appropriate feature extraction method. The most accurate clustering is performed using the K-Means algorithm with 82 clusters, which achieves a silhouette estimate of 0.3764. Silhouette scores between 0.2 and 0.3 indicate that the clusters are mediocre, i.e. the clusters are not well-defined, but, equally, not random either.

Table 2
Best hard clustering results

Embeddings	Feature Selection	Clustering Algorithm	No. of clusters	Silhouette Score
LaBSE	UMAP	K-Means	82	0.3764
Word2Vec	UMAP	K-Means	769	0.3406
BERT	UMAP	K-Means	82	0.3263
LaBSE	UMAP	GMM	20	0.3351
Word2Vec	UMAP	GMM	769	0.2870
BERT	UMAP	GMM	20	0.2381
LaBSE	UMAP	AGG	82	0.3751
Word2Vec	UMAP	AGG	769	0.3224
BERT	UMAP	AGG	769	0.2847

where GMM – Gaussian Mixture Model, AGG – agglomerative clustering.

5.2. Soft Clustering

Experiments with soft clustering, see Table 3, show that the best results are obtained using the *Word2Vec* embedding method with principal component analysis feature selection and the mean shift clustering method. The resulting Silhouette score is 0.46, indicating that the clusters are sufficiently well separated, but still overlap strongly. Additionally, it is observed that more compact clusters are received using *UMAP* feature selection.

Table 3
Best soft clustering results

Embeddings	Feature Selection	Clustering Algorithm	Silhouette Score
Word2Vec	PCA	Mean Shift	0.4614
Word2Vec	UMAP	Mean Shift	0.4472
LaBSE	UMAP	BIRCH	0.3495
Word2Vec	UMAP	BIRCH	0.3461

5.3. Hyperparameter Search

To further improve clustering results, the parameter search is conducted. This allows to find the most optimized set of values, that manages to separate the cluster the best. For this, the *Python* library *Optuna* was selected and studies with desired parameters were created, see Table 4.

Table 4
Parameter search space

	K-Means	Gaussian Mixture	Agglomerative	Mean Shift	BIRCH
Params	Init, n_init, max_iter, tol, algorithm	Covariance_type, tol, reg_covar, init_params	Metric, linkage	Bandwidth, bin_seeding, min_bin_freq, Max_iter	Threshold, branching_factor

The following values are found to be the best regarding maximizing Silhouette score:

1. K-Means
   n_init – 13, max_iter – 971, tol – $1.925*e^{-5}$, algorithm - elkan
1. Gaussian Mixture Model
   covariance_type – diag, reg_covar – 0.0005, init_params – kmeans, tol – $1.211*e^{-5}$
2. Agglomerative clustering
   metric – cosine, linkage - average
3. Mean Shift
   bandwidth – 2.1249, bin_seeding – False, min_bin_freq – 9, max_iter - 249
4. BIRCH
   threshold – 1.7317, branching_factor - 84

With the best new parameters found, new clustering models were trained, and the results can be seen in Table 5. In all cases except for agglomerative clustering, clustering results improved. It was found that for the dataset that is used in this research default parameter agglomerative clustering is still better than the optimized ones. Regarding other models, the result shows that the clusters are well medium to well defined.

Table 5
Clustering results after hyperparameter search

	K-Means	Gaussian Mixture	Agglomerative	Mean Shift	BIRCH
Silhouette Score	0.3749	0.3574	0.3112	0.4770	0.6783

During the initial experiments, the obtained Silhouette scores vary between 0.23 and 0.46, which shows that clusters have structure but are overlapping. When considering the nature of sparse textual data, which results in boundaries between clusters less distinct, the expected score values are in range of 0.2 and 0.5. After hyperparameter search the results improved, where Silhouette scores are between 0.31 and 0.67, showing that results are well in expected theoretical range suggesting that clustering algorithms captures the essence of the underlying structure.

6. Conclusions

The analysis of the dataset showed that on average the 14 non stop words consists in company description, which consists mostly of wholesale and retail, agriculture and freight transportation enterprises. It is very important when working with textual data to complete preprocessing steps well defined order to prevent issues related to data cleaning. The experiments showed that the most promising way to create embeddings for text clustering is using *LaBSE* and *Word2Vec* methods with *UMAP* feature selection. Clustering experiments showed that for hard clustering *K-Means* are the most suited method, meanwhile for soft clustering *BIRCH* showed the most promise.

Declaration on Generative AI

The authors have not employed any Generative AI tools.

References

[1] Ec.europa.eu, Glossary:Statistical classification of economic activities in the European Community (NACE). Url: https://ec.europa.eu/eurostat/statistics-explained/index.php?title=Glossary:Statistical_classification_of_economic_activities_in_the_European_Community_(NACE)

[2] A. Petukhova, J. P. Matos-Carvalho, and N. Fachada, "Text clustering with large language model embeddings," International Journal of Cognitive Computing in Engineering, 6, (2025): 100–108.

[3] E. Bielińska-Dusza, M. Hamerska, Innovative activity of Polish enterprises–a strategic aspect. The similarity of NACE divisions (2021).

[4] Z. Kakushadze and W. Yu, Statistical industry classification (2016).

[5] J. R. Albisua, J. I. I. López, and B. Kamp, Classification of industrial sectors based on their profiles of greenhouse gas emissions and policy implications, Journal of Industrial Engineering and Management, 16(2), (2023): 425–437.

[6] Swimm.io, Word2Vec Explained: How Does It Work?. URL: https://swimm.io/learn/large-language-models/what-is-word2vec-and-how-does-it-work.

[7] B. Lutkevich, What is the BERT language model?, 2024 URL: https://www.techtarget.com/searchenterpriseai/definition/BERT-language-model.

[8] F. Feng, Y. Yang, D. Cer, N. Arivazhagan, and W. Wang, Language-agnostic BERT sentence embedding, 2022. arXiv:2007.01852.

[9] BuiltIn.com, Principal Component Analysis (PCA) Explained, 2024. https://builtin.com/data-science/step-step-explanation-principal-component-analysis.

[10] PairCode.github.io, Understanding UMAP. URL: https://pair-code.github.io/understandingumap/.

[11] D. Raj, The math behind k-means clustering, 2022. https://medium.com/@draj0718/the-math-behind-k-means-clustering-4aa85532085e.

[12] TowardsDataScience.com, Gaussian mixture model clearly explained, 2023. https://towardsdatascience.com/gaussian-mixture-model-clearly-explained-115010f7d4cf.

[13] BuiltIn.com, Hierarchical clustering: Agglomerative + divisive clustering, 2024. URL: https://builtin.com/machine-learning/agglomerative-clustering.

[14] N. Cs21, Balanced iterative reducing and clustering using hierarchies (BIRCH), 2022. URL: https://medium.com/@noel.cs21/balanced-iterative-reducing-and-clustering-using-heirachies-birch-5680adffaa58.

[15] R. B. F. Tuzel, Mean shift clustering. URL: https://homepages.inf.ed.ac.uk/rbf/CVonline/LOCAL_COPIES/TUZEL1/MeanShift.pdf.

[16] S. Yadav, Silhouette coefficient explained with a practical example: Assessing cluster fit, 2023. URL: https://medium.com/@Suraj_Yadav/silhouette-coefficient-explained-with-a-practical-example-assessing-cluster-fit-c0bb3fdef719.