Who are on Dutch Twitter?

Martijn Prikken and **Sijbren van Vaals** and **Sander Beyen**
m.prikken.1@student.rug.nl
s.j.van.vaals@student.rug.nl
s.beyen@student.rug.nl

Abstract

We aimed to increase our understanding of the behaviour and characteristics of different groups of Twitter users. To accomplish this goal, we used k-means clustering to analyse the biographies of Dutch Twitter users in order to identify commonalities within those clusters. We pre-processed a data set of 10 million Dutch tweets, tweeted in 2021. We applied a k-means algorithm to all of the pre-processed biographies, in order to group similar users. The quality of the model is evaluated through analyses of the plotted clusters and comparison with a baseline. We found that the clusters provide good indications of the type of users on Dutch Twitter. All code can be found on: https://github.com/sijbrenvv/MLP.

1 Introduction

Comprehending the behaviours and characteristics of Twitter users can provide valuable information for researchers, organisations, and businesses in order to determine how to engage with this demographic.

Hundreds of millions of Twitter users generate a gigantic amount of data every year, far too much for manual analysis. This goldmine of unstructured data can be studied through a wide variety of machine-learning techniques. Within the context of exploratory data analysis, cluster analysis is the unsupervised process of grouping data instances into relatively similar categories, without a prior understanding of the group's structure or class labels. (Han et al., 2012)

A review of clustering algorithms for Twitter data analysis was done by Alnajran (2017) and they concluded that the methods were powerful for pattern recognition as well as identification of user potentials and interests. Yet, they recommended that future studies should aim to increase the size of the data set to evaluate scalability and also perform more rigorous pre-processing to filter out all the noise.

In our study, we will focus on the most popular hard clustering algorithm according to Preeti Arora and Varshney (2016), which assigns discrete value labels of 0 and 1, called k-means.

Little research has been done on clustering Twitter users based on their biographies. Kohana et al. (2013) used a k-means implementation that calculates cosine similarity for clustering research documents. In their subsequent work, they adapted their algorithm to Twitter users. In this study, we will adopt a similar approach by using n-topics and k-means clustering to group Dutch Twitter users based on their biographies and analyse the topics of the resulting clusters. This will function as a kind of 'topic model', where we find a k amount of topics. This way, we aim to find commonalities within those clusters allowing us to answer the question 'What are the common characteristics of Dutch Twitter users, based on the information provided in their biographies?' The findings will contribute to the development of techniques to profile Twitter user behaviour.

2 Data

Our data set consists of 10 million Dutch tweets, tweeted between January 1st and December 31st, 2021. This data may contain Dutch tweets from countries other than the Netherlands, mainly Belgium. We will not account for this noise, since we expect this to be a small to negligible part of the data set, especially after the pre-processing. Furthermore, there will be users who use different languages in their biography. We will deal with this in our model. The data set also contains metadata such as user_id and in_reply_to. We extract the biographies from the data set.

3 Methodology

3.1 Pre-processing

We pre-processed the data as follows: First, we pre-processed all the biographies in the data set by removing duplicates using the user_id of the data. Second, we remove all rows that do not contain a biography. Next, we removed any special characters: emojis, URLs, hashtags, and punctuation. Then, we tokenised, lowercased, removed stop words, and lemmatised all of the words with nlp-wordnet. Finally, we remove all rows that do not contain a biography after the earlier pre-processing steps [1] The data set contained 92363 distinct biographies at first, this was reduced to 90898 biographies, after the pre-processing.

3.2 Baseline

We used a standard Countvectorizer as our baseline because the Countvectorizer is a basic, easy, and relatively powerful metric to represent textual data for machine learning models. Hence, providing us with a decent baseline. After pre-processing, we used it to get features from all of the tweets. We decided to keep the features that appear in more than 0.001% of the tweets. Afterward, we used the elbow method to find the k-value. However, we found no clear elbow curve (see figure 9). The silhouette scores were also low. We improved these scores by decomposing the features with SVD to 100 components. Unfortunately, this yielded an unclear elbow curve as well (see figure 10). Therefore, we used the silhouette method to find the best k-value for our k-means. The silhouette scores without decomposition were extremely low, with an average score below 0 for most n-clusters (see figure 4). With SVD decomposition, the scores increased substantially (see figure 5). The average silhouette score is the highest for five clusters, therefore we use a k-value of 5 for our baseline. The size, minimal, maximum, and average silhouette score per cluster is visualised in figure 6.

3.3 Our model

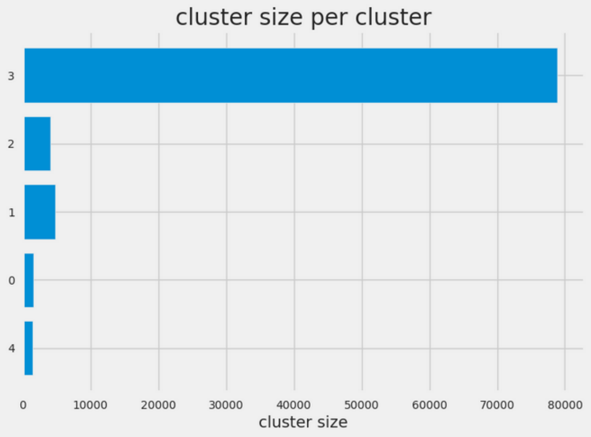
We find that a lot of tweets contain different languages. Therefore, we decided to use a pre-trained multilingual model (Reimers and Gurevych, 2020) to embed our sentences as features. We first convert our embeddings keeping 90% variance using PCA (principal component analysis) to speed up the process. We found the best k-value is 21 using the Yellowbrick cluster visualiser, although the elbow curve in figure 11 keeps declining afterward. There are data points and clusters overlapping, which we think is due to the nature of the data. Since it is very likely that users do not fall into only one category, but multiple. We also considered hierarchical clustering but decided not to do this, because of the size of our data set and features. For both the baseline and our model, we use the default parameters for k-means from scikit-learn, our elbow methods and silhouette scores are calculated using the default parameters of MiniBatchKmeans. (Pedregosa et al., 2011).

[1] In practice, this means biographies that contain only links.

4 Results and evaluation

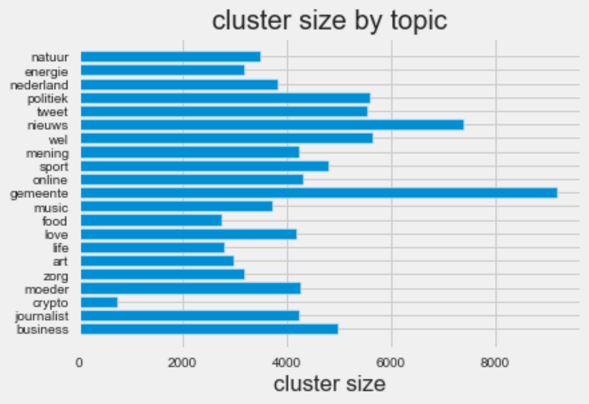
We analyse the data by getting the 10 most frequent words per cluster and their size (frequency of biographies in a cluster). We find that the baseline puts almost all of the biographies into one cluster (see figure 1). This cluster is not very meaningful containing words such as: "nieuws" and "vader". The full results of the baseline can be seen in the appendix figure 3.

Figure 1: Size of each cluster in the baseline



When looking at the results, we find a few different clusters with meaningful types of users. If we look at the results of our model we see a lot of different clusters, which almost all contain clear themes. Furthermore, we see that they are more evenly distributed see figure 2.

Figure 2: Size of each cluster in our model



To visualise the results of our model even more, we plot the different clusters in a graph. We did this by taking the size of the cluster into account. For every cluster, we took the centroid and decomposed it into a 2d array, for which we used PCA.

We gave the node the title of the most frequent word of the clusters. In most cases, this gives a good indication of the cluster. In a few cases, it is not so clear, therefore we provide all of the results in figure 7 in the appendix.

These results give a good indication of what the cluster is about. For example in the "art" cluster, we see that most of it contains photography. The graph can be seen in the appendix in figure 8. Note that the decomposition of the clusters results in information loss. The distance between the dots is explained as the Euclidean distance and not the cosine similarity. For example in figure 8 it looks like the cluster "journalist" is closer to "zorg" than to "nederland". However, when calculating the cosine similarity between those vectors, it appears that "journalist" is closer to "nederland" instead to "zorg".

Nevertheless, the graph gives a good indication of the types of users on Dutch Twitter. We can also clearly observe that the embeddings are more robust than the word counts embeddings since some words are present in multiple clusters, but are different in how they fit in a cluster with respect to the other clusters. For example, the word "marketing" occurs both in the "business" cluster and in the "online" cluster. We find that the clusters are about professions, hobbies, meanings, and descriptions of the users themselves. It is possible that a user fits into multiple of these clusters. For example, we find that a lot of people put "she/her" (which turns into "sheher" due to pre-processing) in their biography. Additionally, we find that very broad topics such as sport and business contain multiple instances inside the 10 most frequent words. For example: for business, we find manager and consultant; for sport, we find football and cycling.

Overall, we find that the 10 most frequent words of the cluster give a great indication of what the cluster is about. In many cases, only the most frequent word already gives enough information. We feel that we have successfully found the characteristics of Dutch Twitter. Based on a simple K-means algorithm on user biographies.

5 Conclusion and future work

In this study, we aimed to understand the behaviour and characteristics of different groups of Dutch Twitter users by using k-means clustering to analyse their biographies. We pre-processed a data set of 10 million Dutch tweets from 2021 and applied a k-means algorithm to the pre-processed biographies. Our model was evaluated through analyses of the plotted clusters and comparison with a baseline.

We found that the clusters generated by our model provided good indications of the types of users on Dutch Twitter. Our analysis revealed common characteristics among various clusters, such as professions, hobbies, and descriptions of the users themselves. The 10 most frequent words of the clusters gave a great indication of what the clusters were about, and in many cases, the most frequent word alone provided enough information.

The research could contribute to the field of machine learning by demonstrating the usability of k-means clustering for Twitter data analysis, providing a framework for further research in this field.

Our results contribute to the development of techniques to profile Twitter user behaviour, providing valuable information for researchers, organisations, and businesses seeking to engage with this demographic.

Future research could further refine the clustering algorithm, explore other clustering techniques such as hierarchical k-means, and experiment with different matrix updating algorithms.

References

Keeley McLean David Latham Annabel Alnajran, Noufa Crockett. 2017. Cluster analysis of twitter data: A review of algorithms. In *Proceedings of the 9th International Conference on Agents and Artificial Intelligence*, pages 239–249.

Jiawei Han, Micheline Kamber, and Jian Pei. 2012. *Data Mining*, volume 3rd edition. Morgan Kaufmann.

Masaki Kohana, Shusuke Okamoto, and Masaya Kaneko. 2013. A clustering algorithm using twitter user biography. *2013 16th International Conference on Network-Based Information Systems*, pages 432–435.

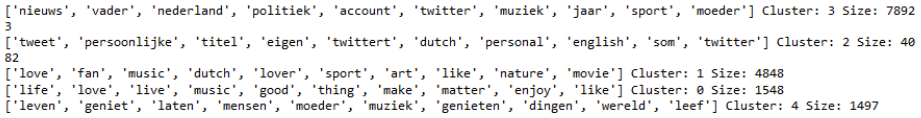
F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. 2011. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830.

Deepali Preeti Arora and Shipra Varshney. 2016. Analysis of k-means and k-medoids algorithm for big data. *Procedia Computer Science*, vol. 78.

Nils Reimers and Iryna Gurevych. 2020. Making monolingual sentence embeddings multilingual using knowledge distillation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics.

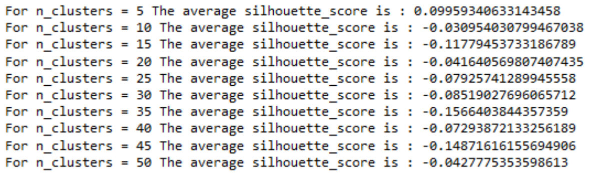
A Appendix

Figure 3: Baseline results



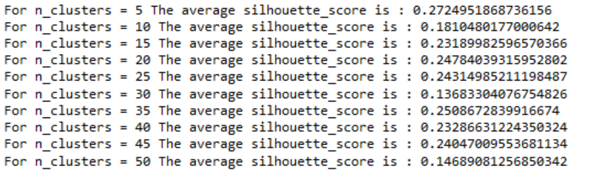
```
['nieuws', 'vader', 'nederland', 'politiek', 'account', 'twitter', 'muziek', 'jaar', 'sport', 'moeder'] Cluster: 3 Size: 7892
3
['tweet', 'persoonlijke', 'titel', 'eigen', 'twittert', 'dutch', 'personal', 'english', 'som', 'twitter'] Cluster: 2 Size: 40
82
['love', 'fan', 'music', 'dutch', 'lover', 'sport', 'art', 'like', 'nature', 'movie'] Cluster: 1 Size: 4848
['life', 'love', 'live', 'music', 'good', 'thing', 'make', 'matter', 'enjoy', 'like'] Cluster: 0 Size: 1548
['leven', 'geniet', 'laten', 'mensen', 'moeder', 'muziek', 'genieten', 'dingen', 'wereld', 'leef'] Cluster: 4 Size: 1497
```

Figure 4: Baseline silhouette scores normal



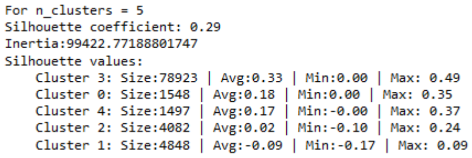
```
For n_clusters = 5 The average silhouette_score is : 0.09959340633143458
For n_clusters = 10 The average silhouette_score is : -0.030954030799467038
For n_clusters = 15 The average silhouette_score is : -0.11779453733186789
For n_clusters = 20 The average silhouette_score is : -0.041640569807407435
For n_clusters = 25 The average silhouette_score is : -0.07925741289945558
For n_clusters = 30 The average silhouette_score is : -0.08519027696065712
For n_clusters = 35 The average silhouette_score is : -0.1566403844357359
For n_clusters = 40 The average silhouette_score is : -0.07293872133256189
For n_clusters = 45 The average silhouette_score is : -0.14871616155694906
For n_clusters = 50 The average silhouette_score is : -0.0427775353598613
```

Figure 5: Baseline silhouette scores SVD



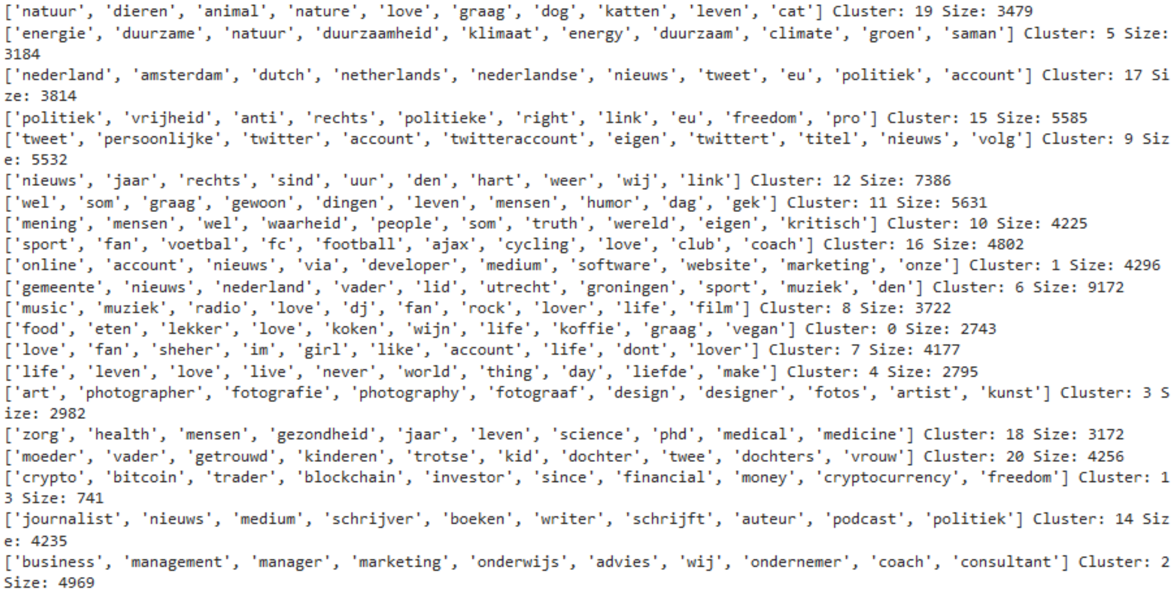
```
For n_clusters = 5 The average silhouette_score is : 0.2724951868736156
For n_clusters = 10 The average silhouette_score is : 0.1810480177000642
For n_clusters = 15 The average silhouette_score is : 0.23189982596570366
For n_clusters = 20 The average silhouette_score is : 0.24784039315952802
For n_clusters = 25 The average silhouette_score is : 0.24314985211198487
For n_clusters = 30 The average silhouette_score is : 0.13683304076754826
For n_clusters = 35 The average silhouette_score is : 0.2508672839916674
For n_clusters = 40 The average silhouette_score is : 0.23286631224350324
For n_clusters = 45 The average silhouette_score is : 0.24047009553681134
For n_clusters = 50 The average silhouette_score is : 0.14689081256850342
```

Figure 6: Baseline silhouette scores per cluster (SVD decomposed)



```
For n_clusters = 5
Silhouette coefficient: 0.29
Inertia:99422.77188801747
Silhouette values:
    Cluster 3: Size:78923 | Avg:0.33 | Min:0.00 | Max: 0.49
    Cluster 0: Size:1548 | Avg:0.18 | Min:0.00 | Max: 0.35
    Cluster 4: Size:1497 | Avg:0.17 | Min:-0.00 | Max: 0.37
    Cluster 2: Size:4082 | Avg:0.02 | Min:-0.10 | Max: 0.24
    Cluster 1: Size:4848 | Avg:-0.09 | Min:-0.17 | Max: 0.09
```

Figure 7: Model results



```
['natuur', 'dieren', 'animal', 'nature', 'love', 'graag', 'dog', 'katten', 'leven', 'cat'] Cluster: 19 Size: 3479
['energie', 'duurzame', 'natuur', 'duurzaamheid', 'klimaat', 'energy', 'duurzaam', 'climate', 'groen', 'saman'] Cluster: 5 Size:
3184
['nederland', 'amsterdam', 'dutch', 'netherlands', 'nederlandse', 'nieuws', 'tweet', 'eu', 'politiek', 'account'] Cluster: 17 Si
ze: 3814
['politiek', 'vrijheid', 'anti', 'rechts', 'politieke', 'right', 'link', 'eu', 'freedom', 'pro'] Cluster: 15 Size: 5585
['tweet', 'persoonlijke', 'twitter', 'account', 'twitteraccount', 'eigen', 'twittert', 'titel', 'nieuws', 'volg'] Cluster: 9 Siz
e: 5532
['nieuws', 'jaar', 'rechts', 'sind', 'uur', 'den', 'hart', 'weer', 'wij', 'link'] Cluster: 12 Size: 7386
['wel', 'som', 'graag', 'gewoon', 'dingen', 'leven', 'mensen', 'humor', 'dag', 'gek'] Cluster: 11 Size: 5631
['mening', 'mensen', 'wel', 'waarheid', 'people', 'som', 'truth', 'wereld', 'eigen', 'kritisch'] Cluster: 10 Size: 4225
['sport', 'fan', 'voetbal', 'fc', 'football', 'ajax', 'cycling', 'love', 'club', 'coach'] Cluster: 16 Size: 4802
['online', 'account', 'nieuws', 'via', 'developer', 'medium', 'software', 'website', 'marketing', 'onze'] Cluster: 1 Size: 4296
['gemeente', 'nieuws', 'nederland', 'vader', 'lid', 'utrecht', 'groningen', 'sport', 'muziek', 'den'] Cluster: 6 Size: 9172
['music', 'muziek', 'radio', 'love', 'dj', 'fan', 'rock', 'lover', 'life', 'film'] Cluster: 8 Size: 3722
['food', 'eten', 'lekker', 'love', 'koken', 'wijn', 'life', 'koffie', 'graag', 'vegan'] Cluster: 0 Size: 2743
['love', 'fan', 'sheher', 'im', 'girl', 'like', 'account', 'life', 'dont', 'lover'] Cluster: 7 Size: 4177
['life', 'leven', 'love', 'live', 'never', 'world', 'thing', 'day', 'liefde', 'make'] Cluster: 4 Size: 2795
['art', 'photographer', 'fotografie', 'photography', 'fotograaf', 'design', 'designer', 'fotos', 'artist', 'kunst'] Cluster: 3 S
ize: 2982
['zorg', 'health', 'mensen', 'gezondheid', 'jaar', 'leven', 'science', 'phd', 'medical', 'medicine'] Cluster: 18 Size: 3172
['moeder', 'vader', 'getrouwd', 'kinderen', 'trotse', 'kid', 'dochter', 'twee', 'dochters', 'vrouw'] Cluster: 20 Size: 4256
['crypto', 'bitcoin', 'trader', 'blockchain', 'investor', 'since', 'financial', 'money', 'cryptocurrency', 'freedom'] Cluster: 1
3 Size: 741
['journalist', 'nieuws', 'medium', 'schrijver', 'boeken', 'writer', 'schrijft', 'auteur', 'podcast', 'politiek'] Cluster: 14 Siz
e: 4235
['business', 'management', 'manager', 'marketing', 'onderwijs', 'advies', 'wij', 'ondernemer', 'coach', 'consultant'] Cluster: 2
Size: 4969
```

Figure 8: All clusters with their most frequent word from our model

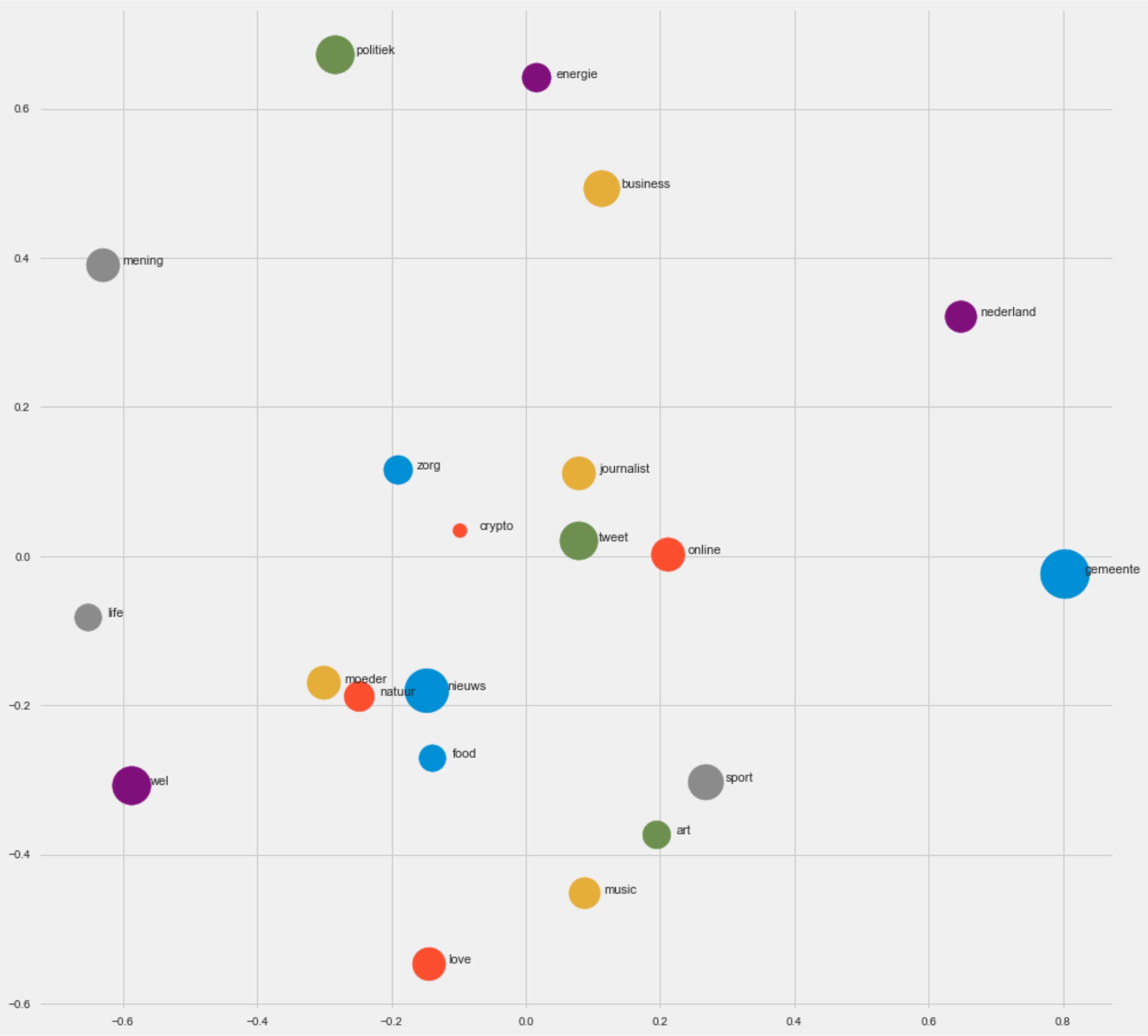


Figure 9: Elbow method visualisation of the ’normal’ baseline

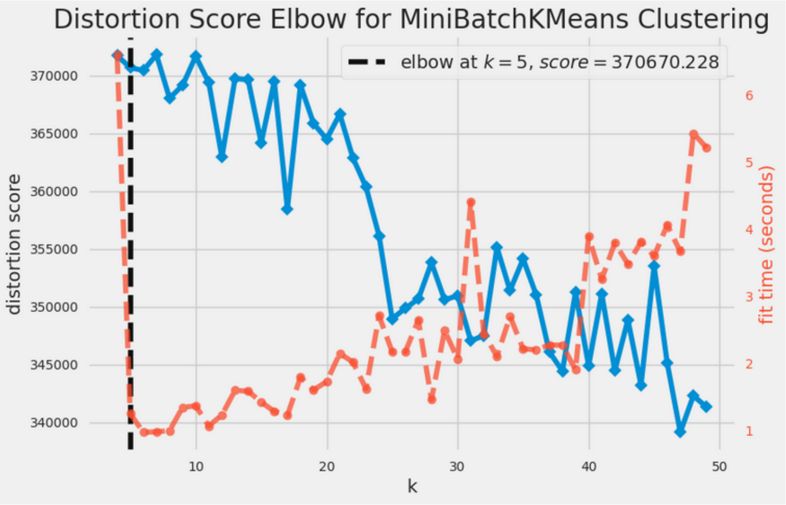


Figure 10: Elbow method visualisation of the decomposed baseline

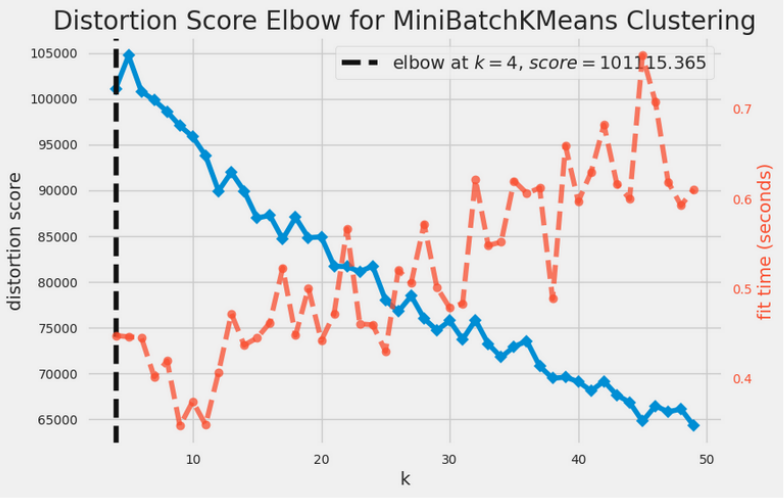


Figure 11: Elbow method visualisation of our model

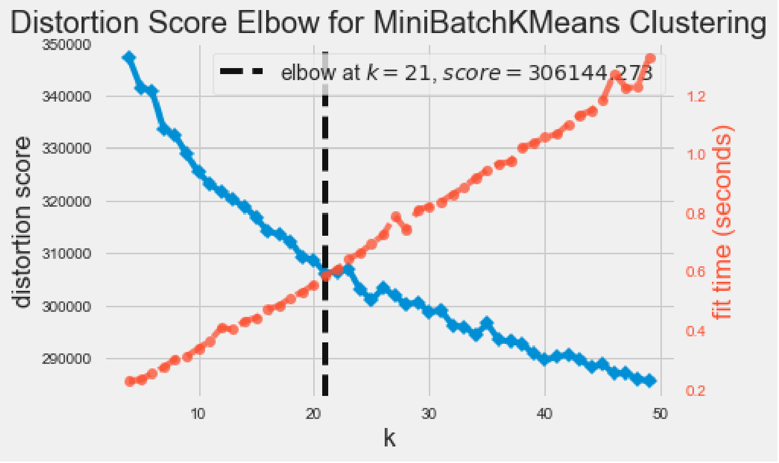
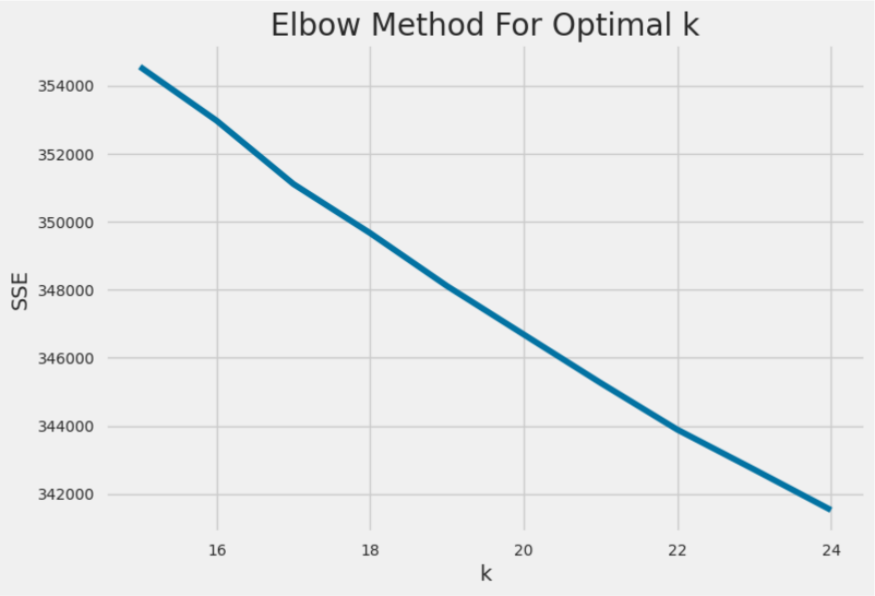


Figure 12: Elbow method visualisation, no clear result


Elbow Method For Optimal k
SSE
k