MSc in Business Analytics

Machine Learning and Content Analytics

Project – Argumentation Mining

Anastasia Polichronopoulou f2822022
Athanasios Papakonstantinou f2822024

Table of Contents

1. Introduction

The aim of the project is the development of models in order to recognize the structure and the argument of some abstracts.

Our datasets consist of 1017 abstracts. For the sentences of each abstract, we have their structure, i.e., “background”, “conclusion”, “method”, “neither”, “objective”, “result”

Dataset length: 1017 abstracts

	document	sentences	labels
666	doi: 10.1111/1365-2656.12572	[The ontogeny of tolerance curves: habitat qua...	[NEITHER, BACKGROUND, BACKGROUND, OBJECTIVE, M...
948	doi: 10.3390/tropicalmed3040111	[Low Praziquantel Treatment Coverage for Schis...	[NEITHER, BACKGROUND, OBJECTIVE, METHOD, METHO...
319	doi: 10.1021/acschemneuro.7b00314	[Novel Trimodal MALDI Imaging Mass Spectrometr...	[NEITHER, BACKGROUND, BACKGROUND, OBJECTIVE, M...
37	doi: 10.1002/chem.201604700	[Covalent Modification of Highly Ordered Pyrol...	[NEITHER, BACKGROUND, RESULT, RESULT, RESULT]
394	doi: 10.1038/ncomms15733	[An autonomous organic reaction search engine ...	[NEITHER, BACKGROUND, OBJECTIVE, METHOD, METHO...

Figure 1: Structure Dataset

and their argument, i.e., “claim”, “evidence”, “neither”

Dataset length: 1017 abstracts

	document	sentences	labels
444	doi: 10.1038/s41558-019-0419-7	[Drivers of declining CO2 emissions in 18 deve...	[NEITHER, NEITHER, CLAIM, NEITHER, NEITHER, EV...
851	doi: 10.1371/journal.pone.0193890	[Optimized 3D co-registration of ultra-low-fie...	[NEITHER, NEITHER, NEITHER, NEITHER, NEITHER, ...
783	doi: 10.1186/s12859-019-2791-8	[DNAscan: personal computer compatible NGS ana...	[NEITHER, NEITHER, NEITHER, NEITHER, NEITHER, ...
462	doi: 10.1038/s41586-019-1423-9	[Weak average liquid-cloud-water response to a...	[NEITHER, NEITHER, NEITHER, CLAIM, EVIDENCE, E...
688	doi: 10.1111/joim.12492	[Current controversies in determining the main...	[NEITHER, NEITHER, NEITHER, NEITHER, NEITHER, ...

Figure 2: Arguments' Dataset

Also, each abstract belongs to a project and each project is being funded by an EU call.

Dataset length: 1017 abstracts

	document	project	eu_call	sentences	labels
711	doi: 10.1126/scitranslmed.aad3106	The proposal, VSV-EBOVAC, directly addresses t...	H2020-EU.3.1.7.13.	[Ebola vaccine R&D: Filling the knowledge gaps...	[NEITHER, NEITHER]
259	doi: 10.1016/j.redox.2019.101123	MASSTRPLAN will train the next generation of i...	H2020-EU.1.3.1.	[Impact of inhibition of the autophagy-lysosom...	[NEITHER, NEITHER, NEITHER, NEITHER, NEITHER, ...
93	doi: 10.1007/s00382-019-04840-y	The goal of PRIMAVERA is to deliver novel, adv...	H2020-EU.3.5.1.	[Impact of model resolution on Arctic sea ice ...	[NEITHER, NEITHER, NEITHER, CLAIM, NEITHER, NE...
654	doi: 10.1101/514125	The increasing occurrence of multidrug-resista...	H2020-EU.1.3.2.	[Extracellular mycobacterial DNA drives diseas...	[NEITHER, NEITHER, NEITHER, NEITHER, NEITHER, ...
284	doi: 10.1021/acs.accounts.7b00495	We outline a 5 year programme that introduces ...	H2020-EU.1.1.	[Exploring Strategies To Bias Sequence in Natu...	[NEITHER, NEITHER, NEITHER, NEITHER, NEITHER, ...

Figure 3: Abstracts with the project and EU call they belong

```
Dataset length: 13 eu_calls
```

	eu_call	text
0	H2020-EU.3.1.1.	[SOCIETAL CHALLENGES - Health, demographic cha...
1	H2020-EU.1.1.	[EXCELLENT SCIENCE - European Research Council...
2	H2020-EU.3.1.6.	[SOCIETAL CHALLENGES - Health, demographic cha...
3	H2020-EU.3.1.3.	[SOCIETAL CHALLENGES - Health, demographic cha...
4	H2020-EU.1.3.1.	[Fostering new skills by means of excellent in...

Figure 4: Dataset of the EU calls and their description

Except recognizing the structure and argument of each sentence we also want to cluster the abstracts according to the previous characteristics.

2. Argument-Structure Prediction

2.1. Arguments' Dataset Analysis

Firstly, we should check the insights of our dataset. As we can see in Figure 5 most of the sentences have no argument ("Neither").

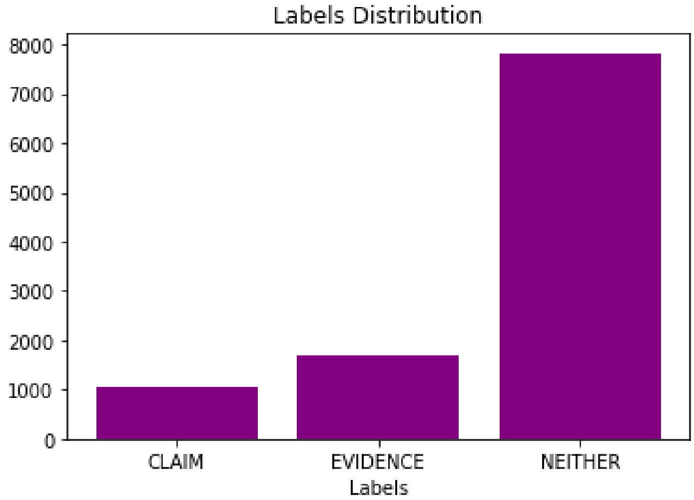


Figure 5: Labels' Distribution

Continuing, since our models will be based on the words that each sentence contains, we take a look at the most common words of our dataset.

As we can see in figure 6, there are a lot of symbols in the most common "word" and also the word "we" appears twice with small and capital "w". Moreover, a lot of "stop words" appear but this type of words gives no information for the sentence. Another common problem is the different grammatic types of the words, since each one will be considered as a different word.

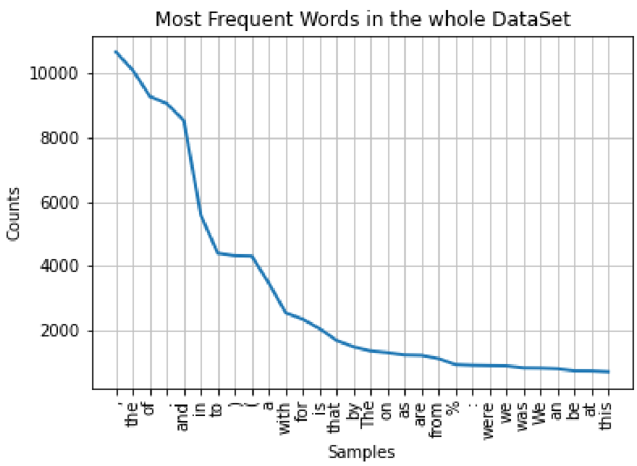


Figure 6: Most frequent words of all the abstracts

After fixing all the above problems, let's look again at the most common words of our abstracts:

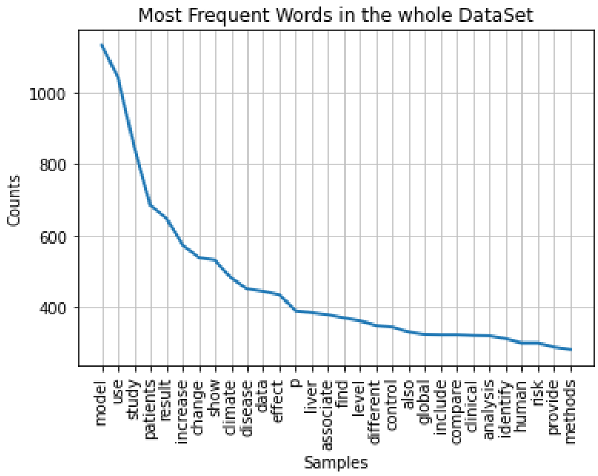


Figure 7: Most frequent words of all the abstracts (fixed)

In figure 8, we can see the most frequent words only for claims and evidence, respectively.

Figure 8: Most frequent words for claims and evidence (respectively)

2.2. Greedy Classifier

Man can notice that there are some common words for both like "model", thus in order to create a classifier we have found the words that belong only in evidence and only in claims.

We build a greedy classifier and check it for 3 different cases.

Words used	Agreement
Top 30 words for claims and evidence	0.43
Words only in claims and only in evidence	0.47

Top 50 words that are only in claims and only in evidence	0.51

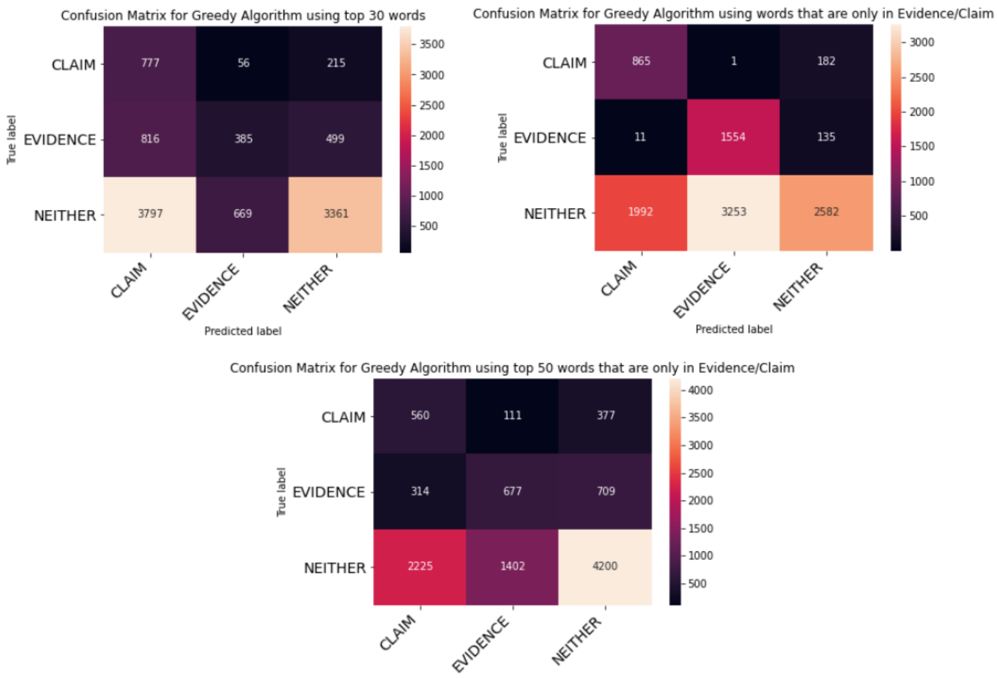


Figure 9: Confusion Matrixes for the Greedy Algorithm

From figure 9, we can see that our algorithm founds right in the first case 4523 (42,77%) labels, in the second 5001 (47,29%) and in the last one 5437 (51,41%). In the first case we observed that in many "Neither" labels it falsely considers them as "Claim" in contrary to the second case where it falsely considers them as "Evidence".

2.3. FastText Approach

Now we used the fasttext approach in order to predict the argument labels for the sentences. We split our dataset in train and validation, and convert it to the expected format for the model.

The first classifier gave us precision 0.75. In order to make it better, we should preprocess the data and change the model's variables.

We found the optimal number for each variable:

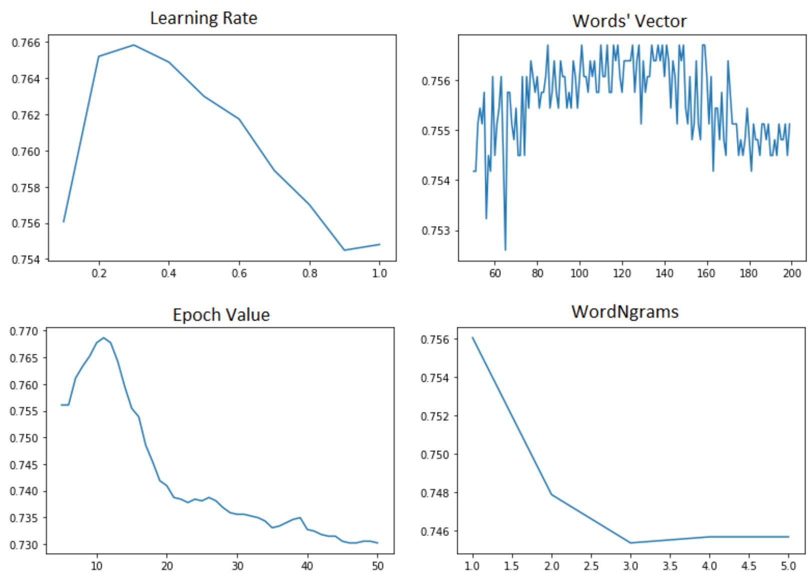


Figure 10: Optimal values for each variable

We trained a model using the optimal values obtained above but we didn't get any better accuracy, so by trying different values we got the best model we accuracy 0.77 which had:

- ✓ `lr=0.3`
- ✓ `epoch = 11`
- ✓ `wordNgrams = 2`
- ✓ `dim = 100`

The model using FastText with the above variables founds correctly 2449 of the labels and as shown below found right the "Neither" labels in most of the cases.

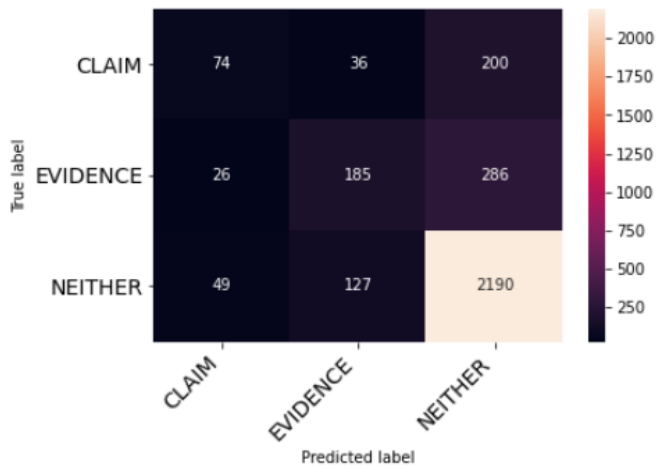


Figure 11: Confusion Matrix for the best model using FastText approach

In the following table we can see that "Neither" has the biggest precision, i.e., the most correctly predicted labels among all the labels. It also has the biggest recall, i.e., the most correctly predicted labels among the real labels. Finally, again “Neither” has the biggest support value, which is logical since most of our data had “Neither” labels and, also, it can be derived from the confusion matrix since the predicted “Neither” has the most values (2676).

	precision	recall	f1-score	support
__label__CLAIM	0.496644	0.238710	0.322440	310.000000
__label__EVIDENCE	0.531609	0.372233	0.437870	497.000000
__label__NEITHER	0.818386	0.925613	0.868703	2366.000000
accuracy	0.771825	0.771825	0.771825	0.771825
macro avg	0.615546	0.512185	0.543004	3173.000000
weighted avg	0.742033	0.771825	0.747850	3173.000000

Figure 12: Classification Report

2.4. Structure Labels

Now we will use the fasttext approach again for predicting the structure labels of each sentence. Firstly, let’s take a look at our dataset’s insights.

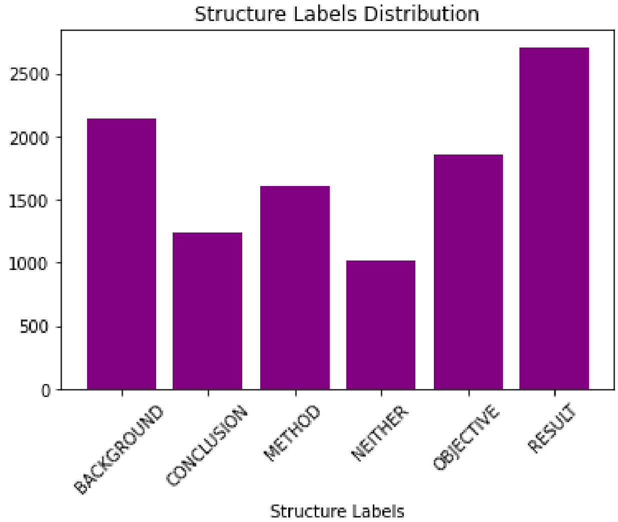


Figure 13: Structure Labels Distribution

As we can see above in figure 10, most of the sentences are either result either background.

Again, as before, we split our dataset in train and validation, and convert it to the expected format for the model.

The first classifier gave us precision 0.55. In order to make it better, we should preprocess the data and change the model’s variables.

We found the optimal number for each variable:

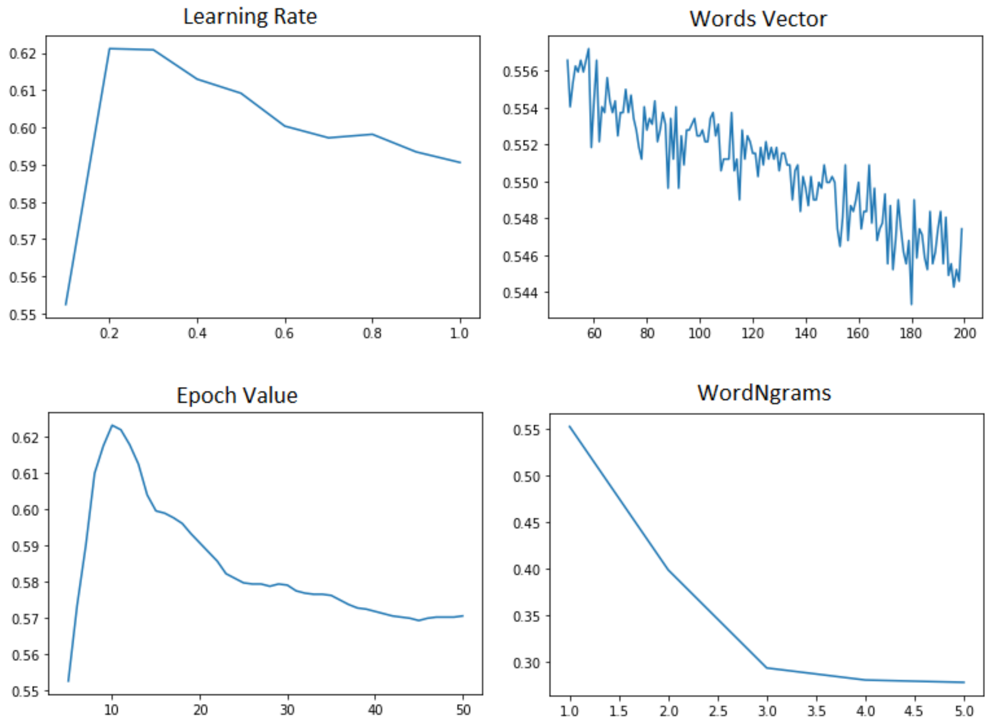


Figure 14: Optimal values for each variable

Again, we trained a model using the optimal values, but we got the best accuracy (0.62) with the following values:

- ✓ `lr=0.1`
- ✓ `epoch = 10`
- ✓ `wordNgrams = 1`
- ✓ `dim = 58`

The model using FastText with the above variables founds correctly 1985 of the labels and as shown below found right the "Result" labels in most of the cases.

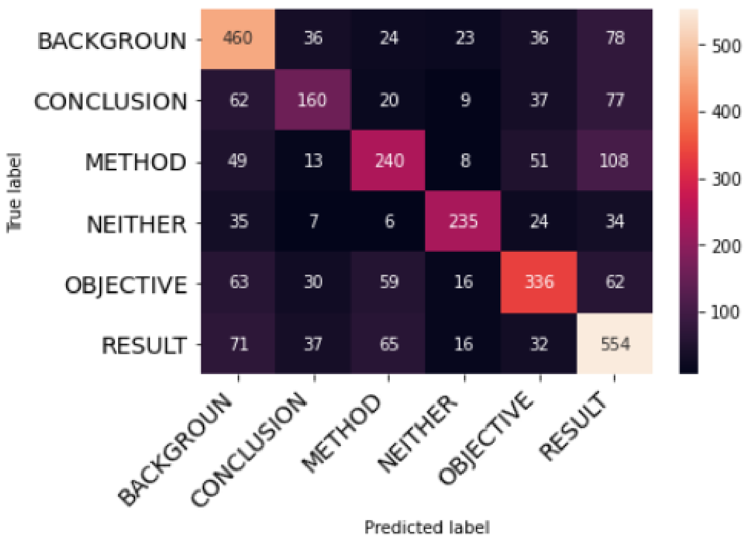


Figure 15: Confusion Matrix for the best model using FastText approach'

In the following table we can see that "Neither" has the biggest precision, i.e., the most correctly predicted labels among all the labels. But "Result" has the biggest recall, i.e., the most correctly predicted labels among the real labels. Finally, "Result" has the biggest support value, which is logical since most of our data had "Result" labels and, also, it can be derived from the confusion matrix since the predicted "Result" has the most values (913).

	precision	recall	f1-score	support
__label__BACKGROUND	0.621622	0.700152	0.658554	657.000000
__label__CONCLUSION	0.565371	0.438356	0.493827	365.000000
__label__METHOD	0.579710	0.511727	0.543601	469.000000
__label__NEITHER	0.765472	0.689150	0.725309	341.000000
__label__OBJECTIVE	0.651163	0.593640	0.621072	566.000000
__label__RESULT	0.606791	0.714839	0.656398	775.000000
accuracy	0.625591	0.625591	0.625591	0.625591
macro avg	0.631688	0.607977	0.616460	3173.000000
weighted avg	0.626063	0.625591	0.622575	3173.000000

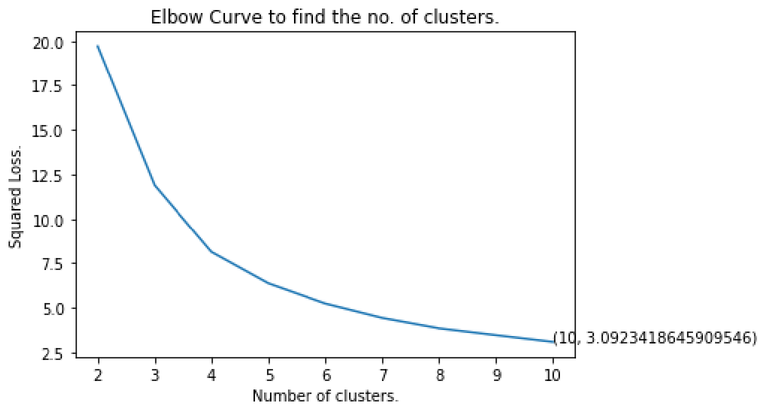
Figure 16: Classification Report

3. Abstract Clustering

In the last part of our project, we want to create cluster for our abstracts by using document embeddings for the abstract, project or EU Call and words embeddings for the arguments.

3.1. Clustering using DE from the abstract

First of all, we should preprocess the text and find the embeddings and then find the optimal number of clusters in order to train our model.




```
The optimal number of clusters obtained is -  10
The loss for optimal cluster is -  3.0923418645909546
```

Figure 17: Optimal number of clusters

0	143
1	154
2	39
3	102
4	175
5	5
6	104
7	188
8	86
9	21

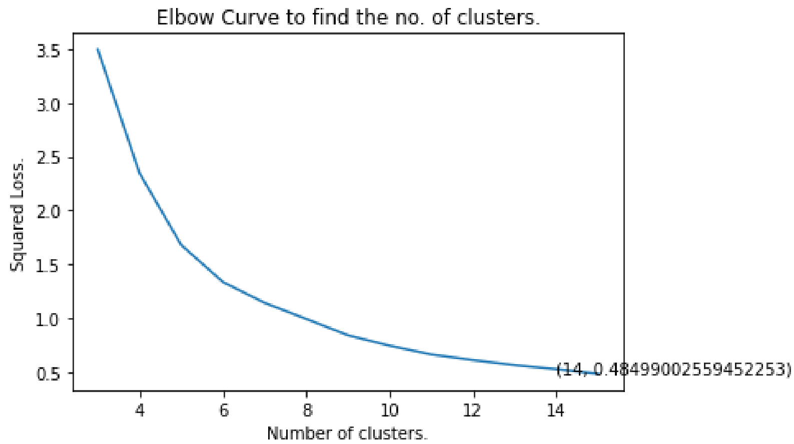
Figure 18: Number of abstracts at each cluster

```
Top terms per cluster:
Cluster 0: ['use', 'study', 'result', 'model', 'increase', 'patients', 'effect', 'show', 'change', 'p']
Cluster 1: ['model', 'study', 'patients', 'use', 'change', 'climate', 'increase', 'result', 'cloud', 'show']
Cluster 2: ['patients', 'liver', 'disease', 'fibrosis', 'p', 'associate', 'nafld', 'level', 'study', 'use']
Cluster 3: ['use', 'show', 'study', 'result', 'core', 'filaments', 'pattern', 'ventricular', 'two', 'order']
Cluster 4: ['model', 'use', 'study', 'patients', 'result', 'increase', 'show', 'data', 'change', 'effect']
Cluster 5: ['model', 'liver', 'change', 'use', 'aerosol', 'climate', 'study', 'disease', 'cloud', 'show']
Cluster 6: ['use', 'study', 'show', 'result', 'model', 'zikv', 'data', 'new', 'provide', 'demonstrate']
Cluster 7: ['dragline', 'silk', 'torsional', 'humidity', 'spider', 'scaffold', 'supramolecular', 'intermolecular', 'interactions', 'among']
Cluster 8: ['model', 'climate', 'use', 'change', 'study', 'data', 'global', 'result', 'different', 'show']
Cluster 9: ['model', 'use', 'study', 'show', 'result', 'increase', 'patients', 'different', 'effect', 'p']
```

Figure 19: Top 10 terms per Cluster

3.2. Clustering using DE from the abstracts, project, and EU calls

Again, we should preprocess the text and find the embeddings and then find the optimal number of clusters in order to train our model. Now our text is the combined string of the abstract's text with the text of the project and the EU Call it belong.




```
The optimal number of clusters obtained is -  14
The loss for optimal cluster is -  0.48499002559452253
```

Figure 20: Optimal number of clusters

0	90
1	36
2	142
3	51
4	17
5	28
6	155
7	83
8	48
9	149
10	66
11	44
12	67
13	41

Figure 21: Number of abstracts at each cluster

```
Top terms per cluster:
Cluster 0: ['health', 'diseases', 'million', 'age', 'cost', 'well-being', 'europe', 'include', 'people', 'increase']
Cluster 1: ['researchers', 'europe', 'new', 'science', 'research', 'base', 'state', 'many', 'model', 'scientific']
Cluster 2: ['climate', 'change', 'model', 'global', 'develop', 'impact', 'use', 'risk', 'focus', 'include']
Cluster 3: ['train', 'researchers', 'new', 'research', 'analytical', 'esrs', 'oxidative', 'modifications', 'tool', 'detect']
Cluster 4: ['image', 'mri', 'ulf', 'new', 'use', 'current', 'magnetic', 'measurements', 'meg', 'brain']
Cluster 5: ['train', 'researchers', 'research', 'br/', 'devices', 'new', 'optical', 'materials', 'switch', 'supramolecular']
Cluster 6: ['researchers', 'europe', 'research', 'br/', 'science', 'new', 'base', 'state', 'scientific', 'many']
Cluster 7: ['model', 'research', 'three', 'br/', 'partner', 'european', 'biomedicine', 'coe', 'infrastructures', 'use']
Cluster 8: ['research', 'mobility', 'researchers', 'experience', 'career', 'cross-sector', 'opportunities', 'model', 'fault', 'fluid']
Cluster 9: ['health', 'diseases', 'research', 'million', 'age', 'virus', 'zikv', 'well-being', 'cost', 'clinical']
Cluster 10: ['aβ', 'researchers', 'ad', 'csf', 'research', 'new', 'europe', 'science', "'s", 'base']
Cluster 11: ['metabolic', 'train', 'researchers', 'drug', 'research', 'provide', 'health', 'knowledge', 'tool', 'innovative']
Cluster 12: ['new', 'researchers', 'base', 'europe', 'use', 'science', 'properties', 'us', 'research', 'many']
Cluster 13: ['health', 'disease', 'liver', 'diseases', 'age', 'million', 'europe', 'cost', 'systems', 'well-being']
```

Figure 22: Top 10 terms per Cluster