CMPE -255 Data Mining Project Report

A survey of sentence embeddings and their applications in scientific paper classification and clustering

Authors:-
Sangram Jagtap
Omkar Nagarkar
Atharva Jadhav
Purvil Patel

Content

Abstract

Mathematical Average of word embeddings are a common baseline for more sophisticated sentence embedding techniques. However, they typically fall short of the performance of more complex models such as BERT and InferSent. In recent years, we have seen significant improvements in the field of sentence embeddings and especially towards the development of universal sentence encoders that can be used for transfer learning in a wide variety of downstream tasks.

Academic Paper Retrieval systems are widely used in academic institutions to store and categorize Scientific papers and articles. There is an extensive literature about clustering these stored papers into categories and finding connections between them using citation links, but these old methods do not account for the content of the papers. We propose a new method for unsupervised clustering of papers by using Concatenated Power Means Sentence embeddings and Universal Sentence Encoder embeddings of abstracts using Natural Language Processing. We cluster 2 datasets of papers by using sentence embeddings of abstracts - using three different word embedding algorithms, Universal Sentence Encoder, and the K Means algorithm. We compare the clustering quality of each approach using Silhouette Score and Davies Bouldin Score metrics.

Problem Statement

Today, many online repositories of scientific papers such as ACM Portal, IEEE Digital Library, and ScienceDirect allow users to explore the resources by subjects. This service provides quick access to a broad range of articles in a particular area of research. Subject classification of papers is mostly done manually according to the keywords authors provide. Some journals and proceedings ask authors to select one or more subjects from a list of subjects when they are submitting the paper. However, classifying / clustering a large collection of scientific resources with regard to a set of subjects is an error-prone and time consuming task. In automatic techniques for subject classification of documents, a simple approach is to do a keyword-based search for subject term or some of its synonyms in paper's title, keywords, and full text. On one hand, title and keywords provide only limited information which may lead to inaccurate decisions and on the other hand, processing the whole text of a paper also takes a long time. Moreover, this method fails if an article is using the semantically equivalent terms but not exactly the same subject words. Here we propose to use sentence embedding and word embedding to this downstream task of document clustering specifically classifying and clustering Scientific papers based on abstracts. The work done for this can be also be used for any generic text classification and document clustering tasks.

Objectives

1. To create Sentence Embeddings for Scientific Papers
2. To implement a neural net based similarity operator for Sentence Embeddings
3. To create Sentence Embeddings Classifier
4. To create Scientific Paper Classifier using Sentence Embeddings
5. To create Scientific Paper Clusterings using Sentence Embeddings
6. To create POC application demonstrating the use of the above objectives along with metadata retrieval

Introduction

Preface

Since “Networks of Scientific Papers” [1] was published in 1965, there has been diverse research focused on clustering academic papers by considering links between them. Academic papers have been clustered using Co-Citation Analysis, Bibliographic Coupling, and Direct Citation relations [2] [3] [4], subject-based algorithmic classification of different granularities [5], by using non-parametric methods such as Adaptive Weights Clustering on their JEL classification tags [2], [6]. But all of these methods do not take the actual content of the papers into account for clustering and categorizing them.

With the latest advancements in Natural language processing, several ways of clustering papers, documents, and texts are being explored e.g. clustering Biomedical publications using TF-IDF, Latent Semantic Indexing, Topic Modeling [7], clustering social media posts for Pharmacovigilance using word embeddings [8].

Word Embeddings

Word embeddings are nowadays pervasive on a wide spectrum of Natural Language Processing (NLP) and Natural Language Understanding (NLU) applications. These word representations improved downstream tasks in many domains such as machine translation, syntactic parsing, text classification, and machine comprehension, among others. Ranging from count-based to predictive or task-based methods, in the past years, many approaches were developed to produce word embeddings, such as Neural Probabilistic Language Model, Word2Vec, GloVe, and more recently ELMo, to name a few. Although most of the recent word embedding techniques rely on the distributional linguistic hypothesis, they differ on the assumptions of how meaning or context are modeled to produce the word embeddings. These differences between word embedding techniques can have unsuspected implications regarding their

performance in downstream tasks as well as in their capacity to capture linguistic properties. Nowadays, the choice of word embeddings for particular downstream tasks is still a matter of experimentation and evaluation. Even though word embeddings produce high-quality representations for words (or sub-words), representing large chunks of text such as sentences, paragraphs or documents is still an open research problem. The tantalizing idea of learning sentence representations that could achieve good performance on a wide variety of downstream tasks, also called universal sentence encoder is, of course, the major goal of many sentence embedding techniques.

Sentence Embeddings

Sentence embeddings are dense vectors that summarize different properties of a sentence (e.g. it's meaning), thereby extending the very popular concept of word embeddings to the sentence level. Universal sentence embeddings have recently gained considerable attention due to their wide range of possible applications in downstream tasks. In contrast to task-specific representations, such as the ones trained specifically for tasks like textual entailment or sentiment, such sentence embeddings are trained in a task-agnostic manner on large datasets. As a consequence, they often perform better when little labelled data is available. To a certain degree, the history of sentence embeddings parallels that of word embeddings, but on a faster scale: early word embeddings models were complex and often took months to train before Mikolov et al. (2013) presented a much simpler method that could train substantially faster and therefore on much more data, leading to significantly better results. Likewise, sentence embeddings originated from the rather resource-intensive 'Skip-thought' encoder-decoder model, before successively less demanding models were proposed that are much faster at train and/or test time. The most popular state-of-the-art approach is the so-called InferSent model, which learns sentence embeddings with a rather simple architecture in a single day (on a GPU), but on very high-quality data, namely, Natural Language Inference Data. InferSent has also set the standards in measuring the usefulness of sentence embeddings by requiring the

embeddings to be “universal” in the sense that they must yield stable and high-performing results on a wide variety of so-called “transfer tasks”.

Concatenated Power Means

Surprisingly, in the paper proposed by Ruckle et al. (2018), a relatively simple method for creating sentence embedding has been proposed. Which gives computationally inexpensive but comparable performance to already established sentence embedding models.

As new and powerful sentence embedding models are being developed each and every few months, we propose to implement and analyse various sentences embedding models in the downstream task of Classifying scientific papers based on the abstracts.

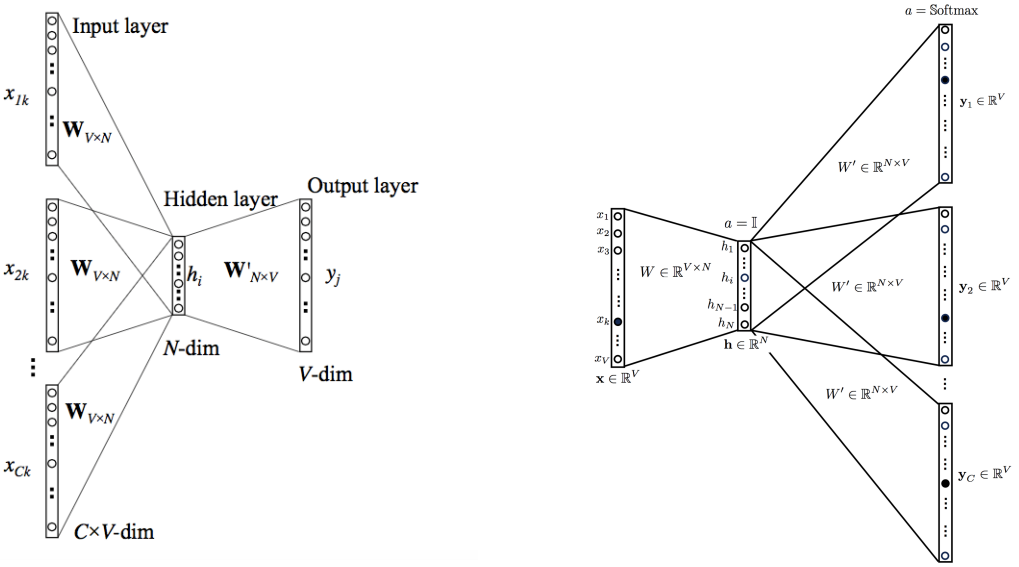
Related Work

Word Embeddings

Word2Vec

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Distributed representations of words and phrases and their compositionality. In Proceedings of the 26th International Conference on Neural Information Processing Systems - Volume 2 (NIPS'13)

Word2Vec is one of the most popular techniques to learn word embeddings using shallow neural network. It was developed by Tomas Mikolov in 2013 at Google. It can be obtained using two methods (both involving Neural Networks): Skip Gram and Continuous Bag Of Words (CBOW).



CBOW Model takes the context of each word as the input and tries to predict the word corresponding to the context. Here The task is predicting the target word by the

input context. The target word is the word coming sequentially after the context words.

In Skip Gram Model we input the target word into the network. The model outputs probability distributions. For each context position, we get probability distributions of context probabilities, one for each word.

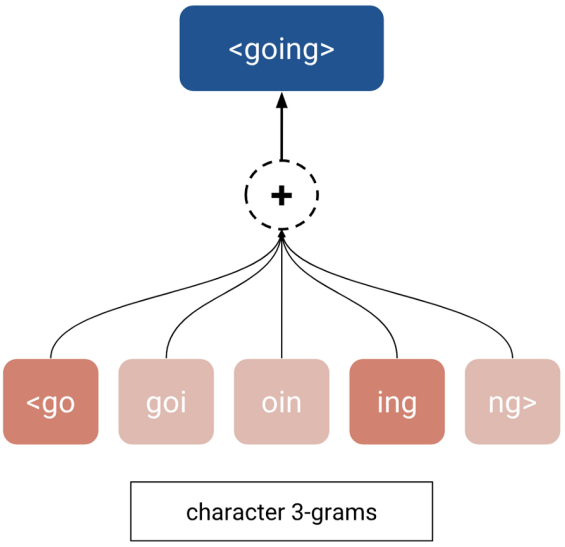
In both cases, the network uses back-propagation to learn.

Here, the weights of the hidden layers are considered as word embeddings for the target word.

FastText

Bojanowski, P., Grave, E., Joulin, A., & Mikolov, T. (2017). Enriching Word Vectors with Subword Information. Transactions of the Association for Computational Linguistics, 5, 135–146. https://doi.org/10.1162/tacl_a_00051

Joulin, Armand, et al. "Bag of Tricks for Efficient Text Classification." ArXiv:1607.01759 [Cs], Aug. 2016. arXiv.org, http://arxiv.org/abs/1607.01759.

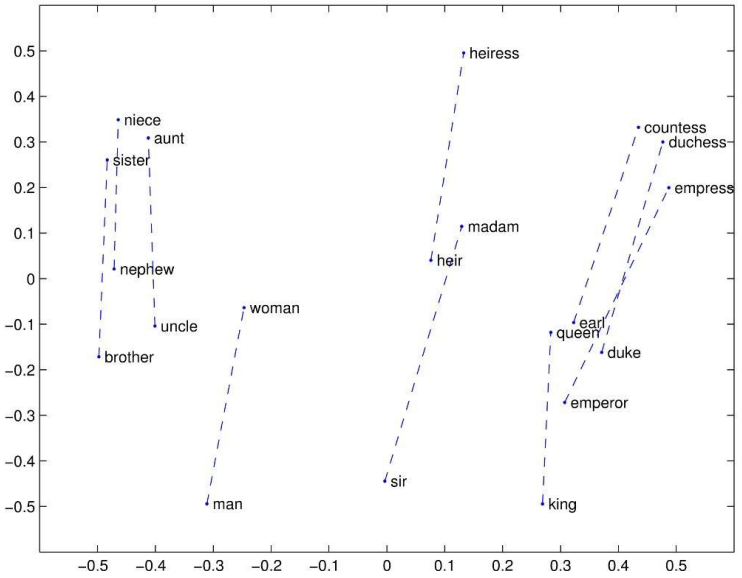


FastText is an extension to Word2Vec proposed by Facebook in 2016. Instead of feeding individual words into the Neural Network, FastText breaks words into several n-grams (sub-words). The word embedding vector for a word will be the sum of all its n-grams. After training the Neural Network, we get word embeddings for all the n-grams given the training dataset. Rare words can now be properly represented since it is highly likely that some of their n-grams also appear in other words. Although it takes a longer time to train a FastText model (number of n-grams > number of words), it performs better than Word2Vec and allows rare words to be represented appropriately.

Glove

Pennington, J., Socher, R., & Manning, C. (2014). Glove: Global Vectors for Word Representation. Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP). Presented at the Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP). https://doi.org/10.3115/v1/d14-1162

$$\hat{J} = \sum_{i,j} f(X_{ij})(w_i^T \tilde{w}_j - \log X_{ij})^2$$

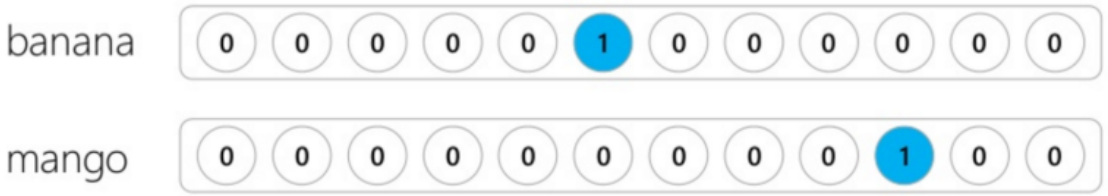


GloVe stands for “Global Vectors”. GloVe captures both global statistics and local statistics of a corpus, in order to come up with word vectors. GloVe method is built on an important idea that you can derive semantic relationships between words from the co-occurrence matrix. The co-occurrence matrix denotes how many times a word has co-occurred with another word. After that a mathematical equation of cost is optimized to get the word embedding. Here, $f(X_{ij})$ is the word embedding and X is the co-occurrence matrix. GloVe does not use any neural network models, but it relies on stochastic gradient descent to optimize.

Sentence Embeddings

Bag of Words

Bag of Words vector representations are the most commonly used traditional vector representation. Each word or n-gram is linked to a vector index and marked as a 0 or a 1 depending on whether it occurs in a given document. TF-IDF is adept for classifying documents as a whole, but word embeddings are better at identifying contextual content.



Power Mean

Andreas Ruckle, Steffen Eger, Maxime Peyrard, Iryna Gurevych. 2018. Concatenated p-mean Word Embeddings as Universal Cross-Lingual Sentence Representations. *ArXiv, abs/1803.01400*.

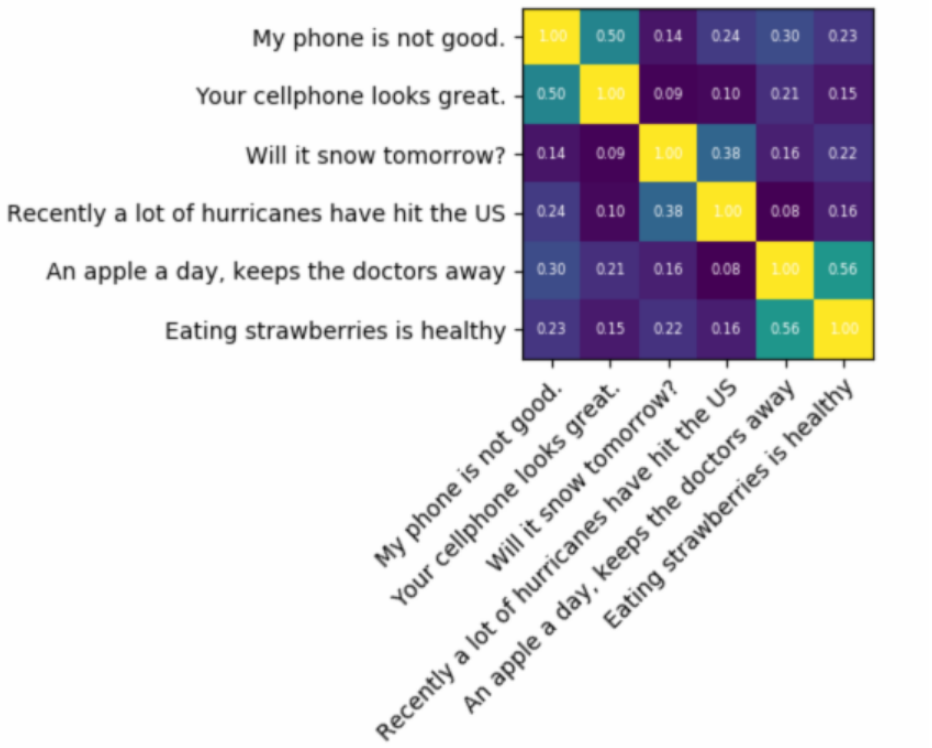
Centroid method summarize a sequence of embeddings w1 ,..., wn ∈ R by component-wise arithmetic averages. Power Means method generalises the average word embeddings by retrieving many well-known means such as the arithmetic mean (p=1), the geometric mean (p=0), and the harmonic mean (p=−1). When p=±∞, the power mean specializes to the minimum (p=−∞) and maximum (p= +∞) of the sequence.

$$\forall i = 1, \ldots, d: \quad \frac{w_{1i} + \cdots + w_{ni}}{n} \quad \left(\frac{x_1^p + \cdots + x_n^p}{n} \right)^{1/p}; \quad p \in \mathbb{R} \cup \{\pm\infty\}$$

Universal Sentence Encoder

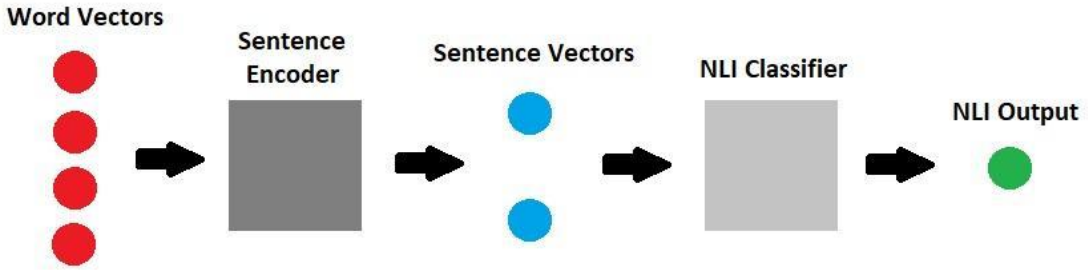
Cer, D., Yang, Y., Kong, S., Hua, N., Limtiaco, N., John, R.S., Constant, N., Guajardo-Cespedes, M., Yuan, S., Tar, C., Sung, Y., Strope, B., & Kurzweil, R. (2018). Universal Sentence Encoder. ArXiv, abs/1803.11175.

The Universal Sentence Encoder encodes text into high dimensional vectors that can be used for text classification, semantic similarity, clustering, and other natural language tasks. The pre-trained Universal Sentence Encoder is publicly available in Tensorflow-hub. Pre-trained sentence encoders aim to play the same role as word2vec and GloVe, but for sentence embeddings: the embeddings they produce can be used in a variety of applications, such as text classification, paraphrase detection, etc. Typically they have been trained on a range of supervised and unsupervised tasks, in order to capture as much universal semantic information as possible. It comes with two variations i.e. one trained with Transformer encoder and other trained with Deep Averaging Network (DAN). The two have a trade-off of accuracy and computational resource requirement. While the one with Transformer encoder has higher accuracy, it is computationally more intensive. The one with DNA encoding is computationally less expensive and with little lower accuracy.



InferSent

The authors describe solutions to two important questions that arise in building a sentence embedding model – the type of task to be used for training the sentence vectors and the preferable neural network architecture to use to generate sentence encodings.The authors, first, generate sentence vectors using a sentence encoding architecture and word vectors as input. This encoder is then followed by a classifier that takes the encoded sentences as input and trains the sentence vectors.

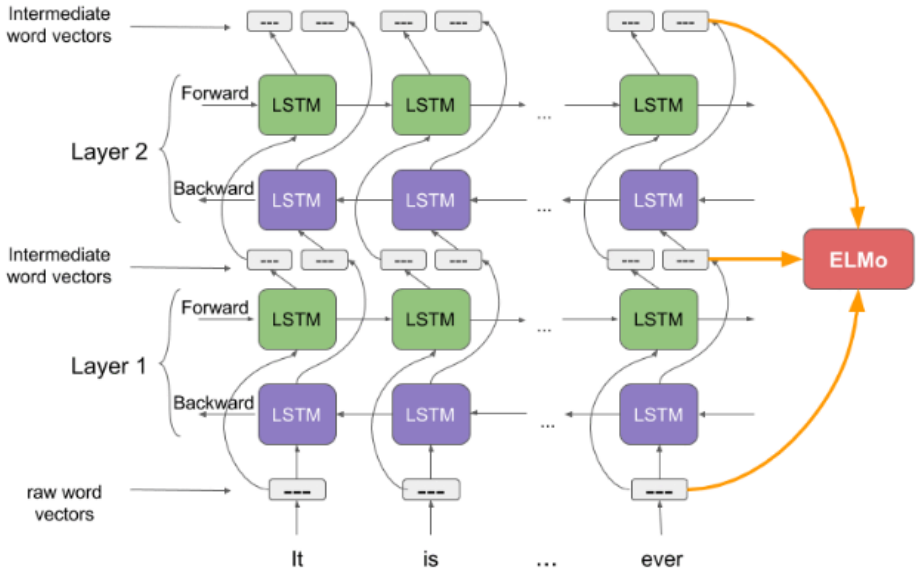


They show that sentence embeddings generated from models trained on a natural language inference classifier (more details below) give the best results in terms of accuracy on downstream tasks. They also explore a number of different architectures for sentence encoding. Namely, standard recurrent models such as LSTMs and GRUs, a self-attentive network and a hierarchical convolutional network.

ELMo

Matthew Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, Luke Zettlemoyer. Association for Computational Linguistics, 2018. https://www.aclweb.org/anthology/N18-1202.pdf

Embedding from Language Model uses representations from a bi-directional LSTM that is trained with a language model objective on a large text dataset. ELMo representations are a function of the internal layers of the bi-directional Language



Model (biLM), which provides a very rich representation about the tokens. Like in fastText, ELMo breaks the tradition of word embeddings by incorporating sub-word units, but ELMo has also some fundamental differences with previous shallow representations such as fastText or Word2Vec. It uses a deep representation by incorporating internal representations of the LSTM network, therefore capturing the meaning and syntactical aspects of words. Since ELMo is based on a language model, each token representation is a function of the entire input sentence, which can overcome the limitations of previous word embeddings where each word is usually modeled as an average of their multiple contexts.

Downstream Tasks of Sentence Embeddings

Text Classification

Text Classification, which is in general to mark a text fragment with a label depending on its content to categorise various datasets with text classified by their semantics. Sentence embeddings are a hot topic in natural language processing (NLP) because they facilitate better text classification than using word embeddings alone. Sentence embeddings have been used in many different classification tasks such as sentiment analysis, review classification, etc

Paraphrase detection

Yu, Jianfei, and Jing Jiang. “Learning Sentence Embeddings with Auxiliary Tasks for Cross-Domain Sentiment Classification.” Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, Association for Computational Linguistics, 2016, pp. 236–246. ACLWeb, doi:10.18653/v1/D16-1023.

Paraphrase detection is used to determine whether two text fragments are paraphrases of each other. The pair of text fragments could be labeled by a score reporting degree of text similarity or by a binary mark reporting the existence of a similarity. The vector representation of each paraphrase is extracted from an encoder-decoder model which is trained on sentence paraphrase pairs.

Text Similarity/Entailment

Text similarity has to determine how ‘close’ two pieces of text are both in surface closeness (lexical similarity) and meaning (semantic similarity). Instead of doing a word for word comparison, we also need to pay attention to context in order to capture more of the semantics. To consider semantic similarity we need to focus on

phrase/sentence/paragraph levels (or lexical chain level) where a piece of text is broken into a relevant group of related words prior to computing similarity. We know that while the words significantly overlap, these two phrases actually have different meanings. Sentence embeddings give a condensed form of meaning and context of a sentence.

Summarization

Padmakumar, Aishwarya and Akanksha Saran. “Unsupervised Text Summarization Using Sentence Embeddings.” (2016).

Text summarization is the process of distilling the most important information from a source (or sources) to produce an abridged version for a particular user (or users) and task (or tasks). Dense vector representations of words, and more recently, sentence vectors, have been shown to improve performance in a number of NLP tasks. There are methods to perform unsupervised extractive and abstractive text summarization using sentence embeddings.

Critical Assessment of sentence Embeddings

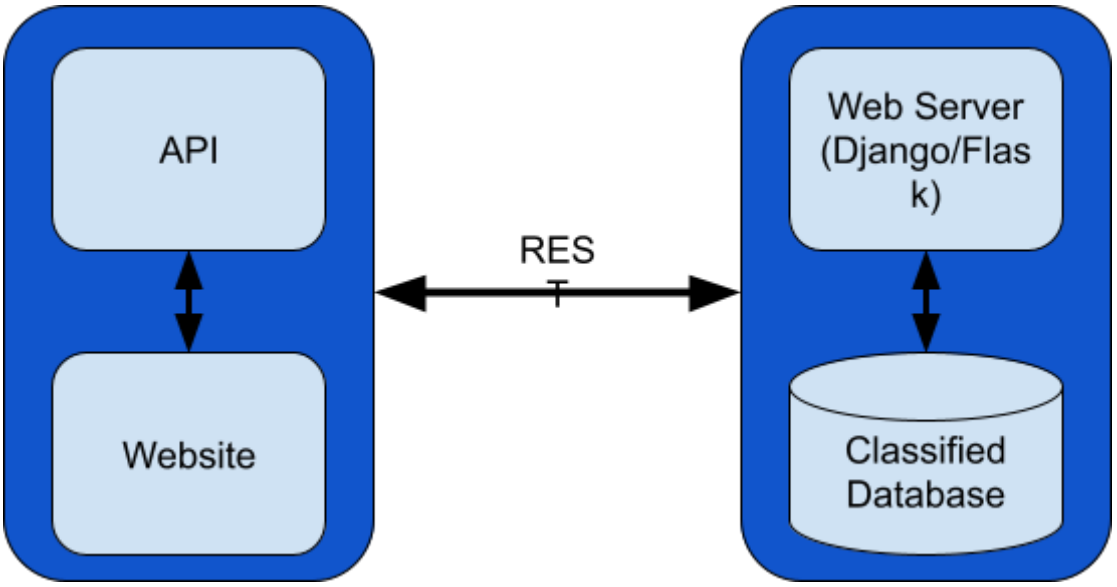
Parameters	Methods to Create sentence Embeddings						
	Word2vec (BOW)	p-Mean	Fastext BOW	GLOVE (BOW)	USE (DAN)	USE (Transformer)	Infersent
Learning Method	Unsupervised	Unsupervised	Unsupervised	Unsupervised	Unsupervised augmented to Supervised	Unsupervised augmented to Supervised	Supervised
Order of Words	NOT considered	NOT considered	NOT considered	NOT considered	Considered	Considered	Considered
Word Frequency	NOT considered	NOT considered	NOT considered	NOT considered	NOT considered	NOT considered	NOT considered
Semantic relation between text	NOT considered	NOT considered	NOT considered	NOT considered	Considered	Considered	Considered
Needs Training	NO	NO	NO	NO	Yes	Yes	Yes

It has been observed that though the methods of generating sentence embeddings vary a lot in terms of complexity(low [BOW] to high[transformer]), the results in practical

tasks, such as text similarity, text classification, entailment analysis etc., do not vary as much with complexity. We therefore try to modify some baseline sentence embeddings to achieve enhanced results while saving resources.

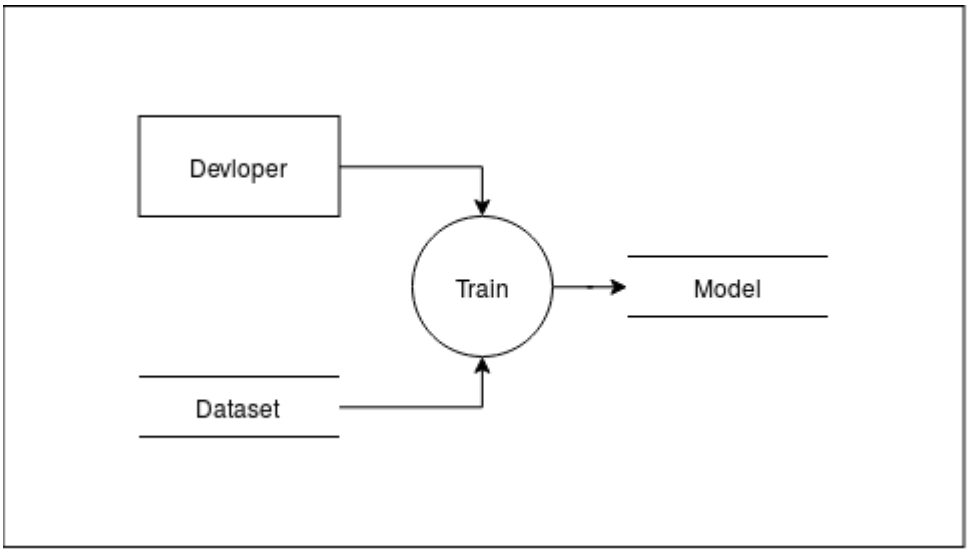
Design

Application High Level Design

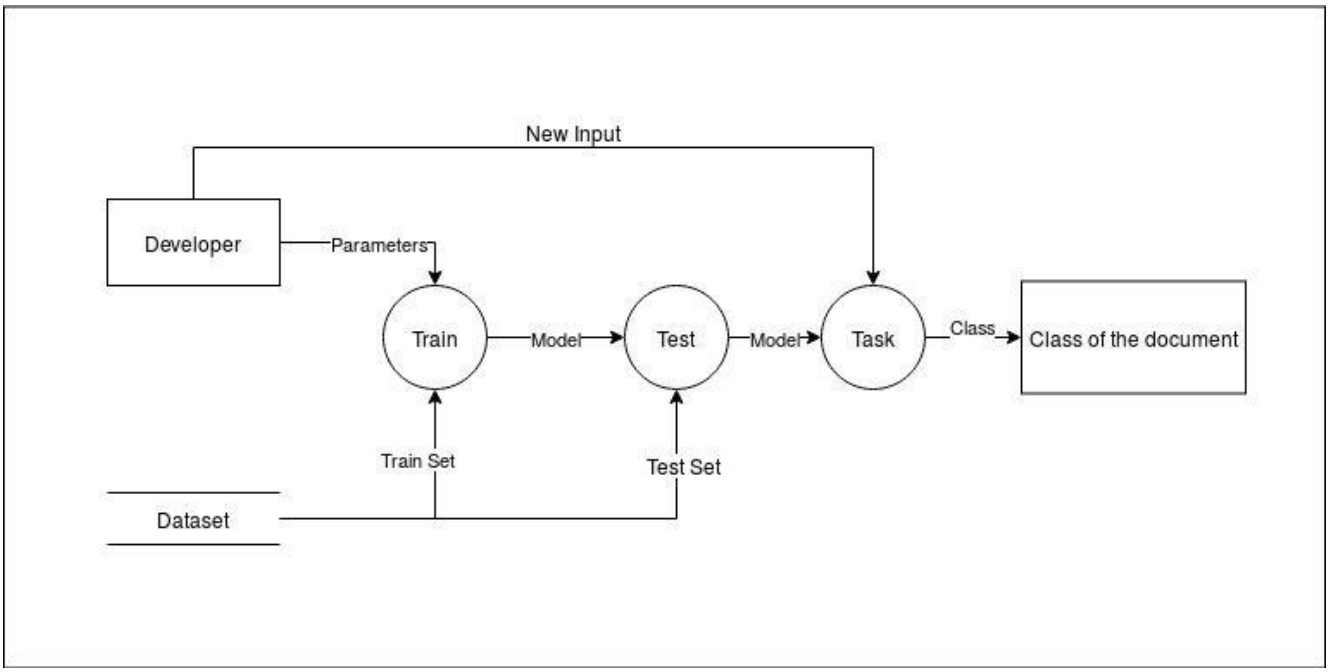


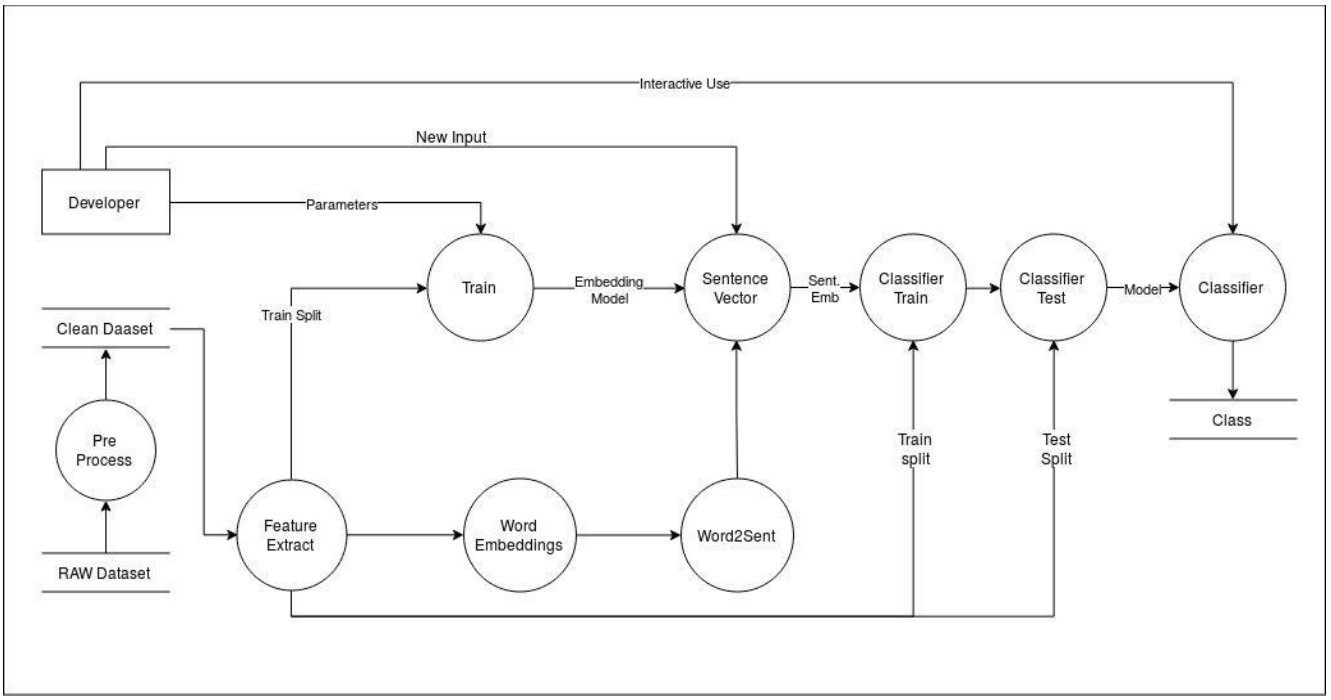
Data Flow Diagram For Model Training

DFD-0



DFD-1

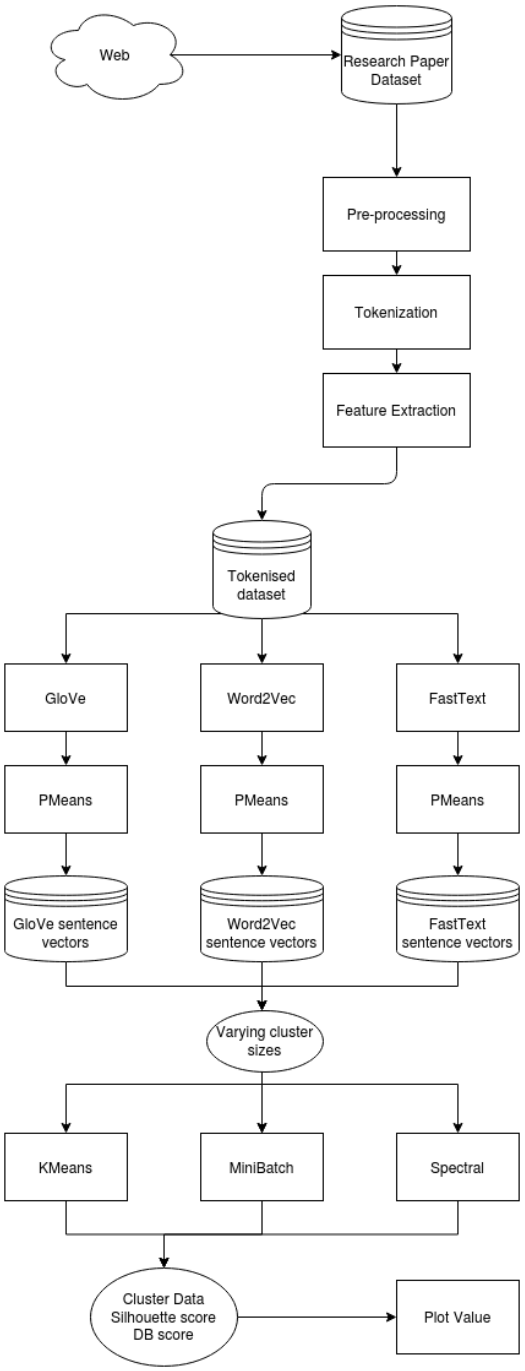

New Input
Developer
Parameters
Train
Model
Test
Model
Task
Class
Class of the document
Train Set
Test Set
Dataset


Interactive Use
New Input
Developer
Parameters
Train
Embedding Model
Sentence Vector
Sent. Emb
Classifier Train
Classifier Test
Model
Classifier
Clean Daaset
Train Split
Pre Process
Class
Train split
Test Split
Feature Extract
Word Embeddings
Word2Sent
RAW Dataset

DFD-2

Methodology

Overview



Dataset

We collected the first dataset from https://core.ac.uk/services/dataset/, the website offers multiple datasets varying in the year of publication and total size. We chose the metadata dataset published in 2013 as it suited our need of having just the abstract, as we did not have any use of the body for clustering. Second Evaluation dataset was collected from arxiv.org using the website scraper library. We downloaded papers of 50 Fields from arxiv.

Sentence Embeddings

The abstracts were lemmatized, stop words were removed and converted to arrays of word embeddings For every word embedding type. We used GloVe 50d, Word2Vec 100d, and FastText 300d word embeddings.
Concatenated Power Means sentence embedding and centroid sentence embeddings were created for every word embedding type. The Power Means method generalizes the average word embeddings by retrieving many well-known means such as the arithmetic mean (p=1), the geometric mean (p=0), and the harmonic mean (p=−1). When p=±∞, the power means converges to the minimum (p=−∞) and maximum (p= +∞) of the sequence.

$$\left(\frac{x_1^p + \cdots + x_n^p}{n}\right)^{1/p} ;\ p \in R \cup \{\pm\infty\}$$

Here x is word embedding and n is the number of words.
We used p=1, +∞, −∞, 2, and 4.
We chose pmeans because of its established accuracy in multiple downstream tasks and low computational power requirement [12] [13].
centroid Sentence embedding is power means with p=1.

$$\left(\frac{x_1 + \cdots + x_n}{n}\right)$$

We compared pmeans with the simplest sentence embedding algorithm i.e. centroid. We did not compare it with SIF-Sentence Embeddings [14] since the number of words repeated in paper abstracts is very less.
We used Kmeans, MiniBatch Kmeans, and spectral clustering algorithms.

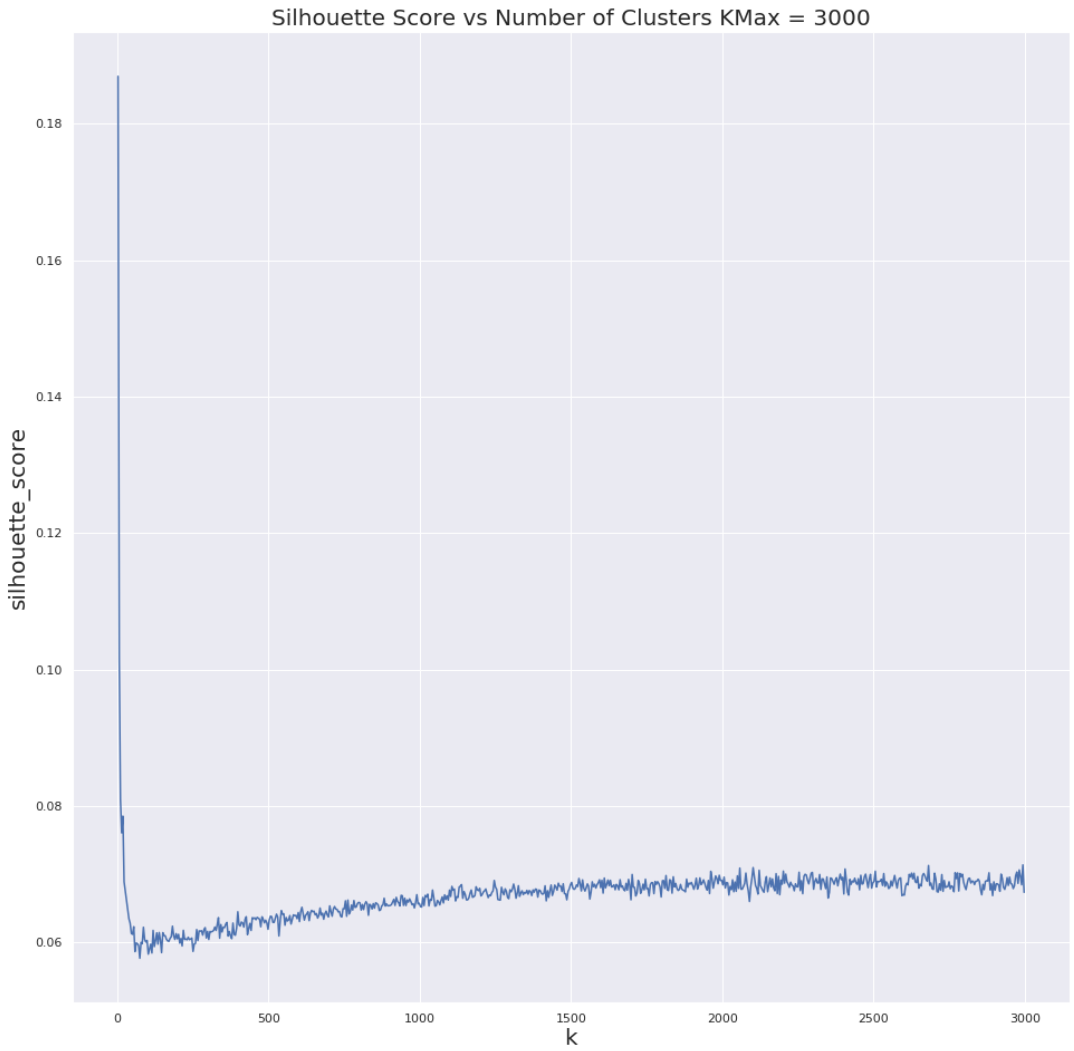
Selection of the number of clusters (K) using Zipf's law We calculated silhouette score for range K = 2, 6, 10, 14, …., 3000 (skipping every 4 digits) with 30000 papers for Kmeans using Pmeans with GloVe. We found out that for every multiple of the square root of 30000 (i.e. 173.2 ≈ 173) the Silhouette score did not increase substantially after 346 (i.e. twice the square root). The Silhouette score was 0.1 to 0.8 for K = 2 to 18. But these did not give us enough clusters from an application point of view.
We trained our clustering model in the following combinations - Algorithm: K Means, Mini Batch K Means, Spectral. Word Embedding: GloVe 50d, Word2Vec 100d, FastText 300d. Sentence Embeddings: pmeans, centroid. Cluster and number of papers pairs: (50, 3000), (250, 15000), (350, 30000). 250 and 350 are approx twice the square root of 15000 and 30000 respectively So there are a total of 54 models.

Clustering

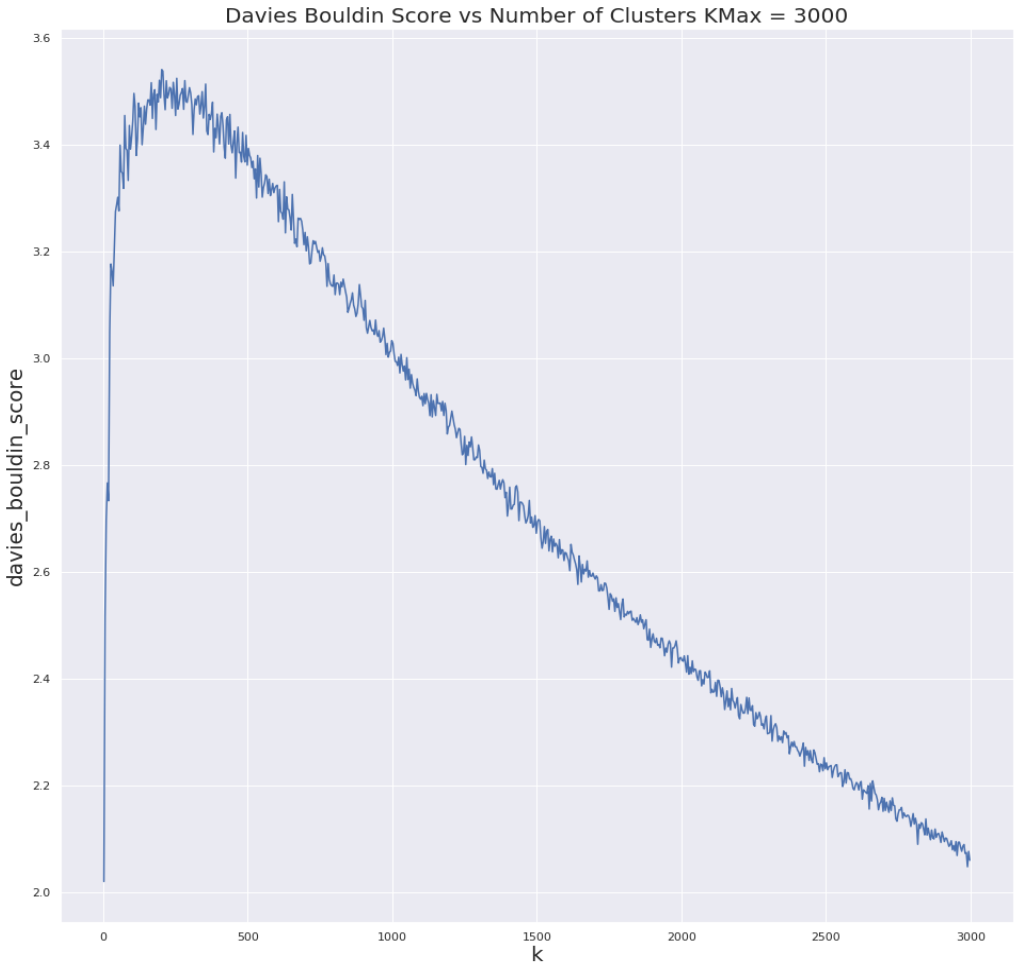
We ran over 67 Clustering models for each set of parameters and number of papers and datasets, models took from 1 to 6 hours to complete clustering. We used google colab cloud platform and High Performance Computing provided by the Computer Engineering and Information Technology department. We compared the Silhouette score and Davies Bouldin Score of every model. Later these models and databases of already clustered papers are used for application.

Results

Clustering Quality as a function of Dataset Size



As it can be seen, the Clustering Quality as measured by the Silhouette score decreases at first and then steadily increases as we increase the cluster size. Highest Silhouette Score can be seen in the lower number of clusters.



Davies Bouldin score is high for clusters 1 to 500 then it decreases linearly.

Clustering Quality for Word and Sentence Embeddings

Top 6 clustering models for different CORE dataset sizes

Model comparison for 30000 papers and 350 Clusters for Kmeans

Sentence Embedding Algorithm	Word Embedding	Silhouette Score	Davis Bouldin Score
Centroid	GloVe	0.0814	2.4284
Centroid	word2Vec	0.0740	2.5006
Centroid	FastText	0.0657	2.7902
Pmeans	GloVe	0.0631	3.5175
Pmeans	word2Vec	0.0621	3.5023
Pmeans	FastText	0.0574	3.8057

Model comparison for 15000 papers and 250 Clusters for Kmeans

Sentence Embedding Algorithm	Word Embedding	Silhouette Score	Davis Bouldin Score
Centroid	GloVe	0.0816	2.3878
Centroid	word2Vec	0.0776	2.4808
Centroid	FastText	0.0674	2.7141
Pmeans	GloVe	0.0635	3.3782
Pmeans	word2Vec	0.0617	3.3748
Pmeans	FastText	0.0595	3.7380

Model comparison for 3000 papers and 50 Clusters for Kmeans

Sentence Embedding Algorithm	Word Embedding	Silhouette Score	Davis Bouldin Score
Centroid	GloVe	0.0871	2.2623
Centroid	word2Vec	0.0770	2.3245
Centroid	FastText	0.0735	2.5742
Pmeans	GloVe	0.0643	3.1729
Pmeans	word2Vec	0.0604	3.1012
Pmeans	FastText	0.0331	3.3654

Top 6 Clustering models for ARXIV dataset

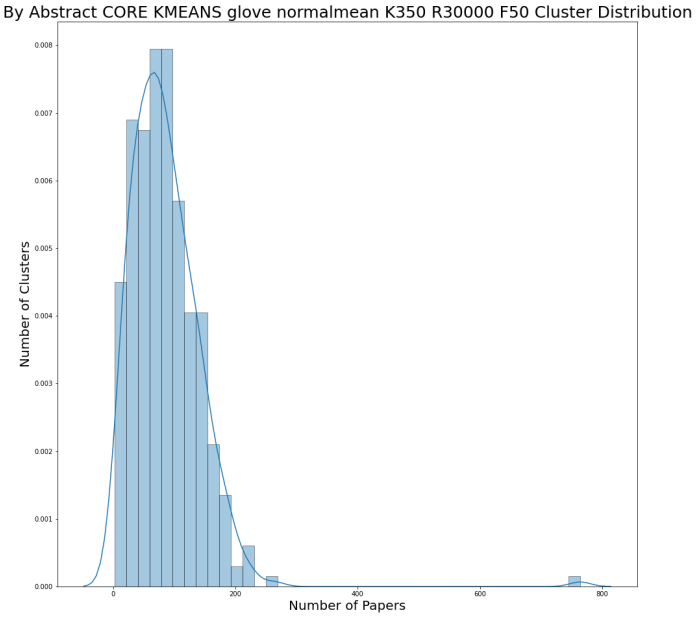
We clustered the Arxiv dataset after evaluation of the CORE dataset to compare clustering by title and by abstract.

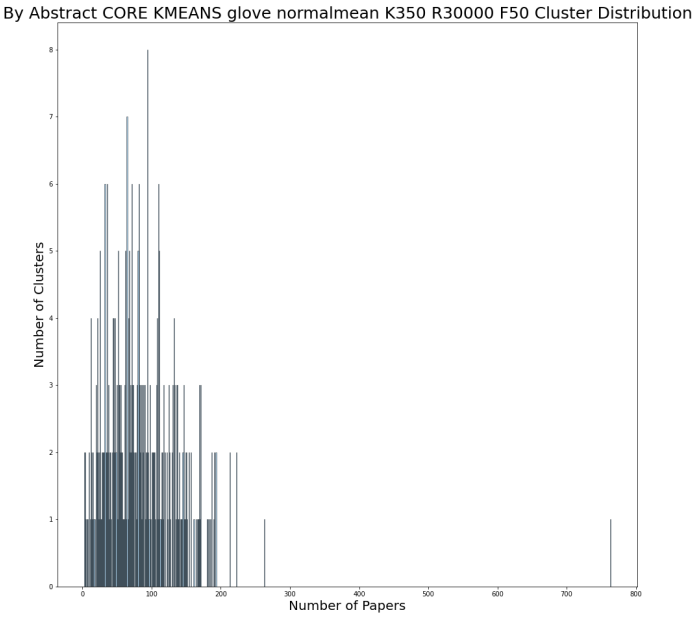
Sentence Embedding Algorithm	Word Embedding	Part of Paper used	Silhouette Score	Davis Bouldin Score
Pmeans	GloVe	Title	0.0185	3.4173
USE	-	Title	0.0170	3.9892
Pmeans	GloVe	Abstract	0.0123	4.0187
USE	-	Abstract	0.0115	4.1362
Centroid	GloVe	Title	-0.0044	4.1330
Centroid	GloVe	Abstract	-0.0306	5.7553

Cluster Distributions of Datasets

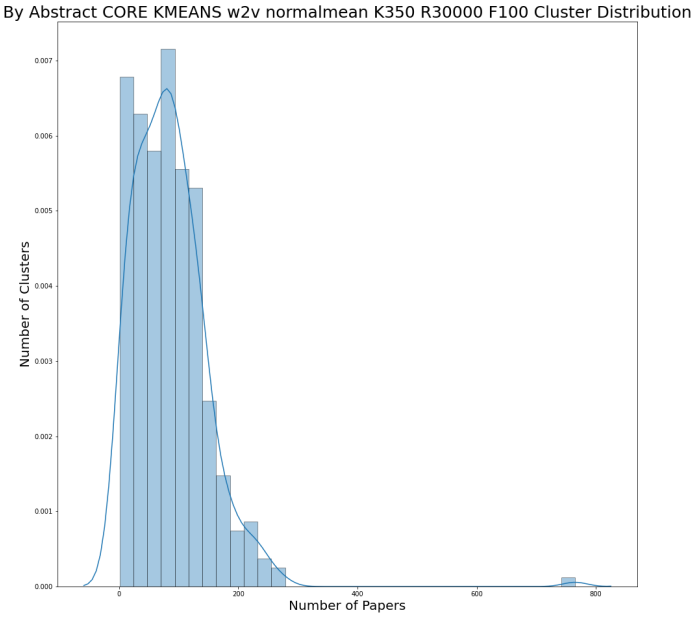
Below are the distributions for the CORE and Arxiv dataset using the various sentence embeddings. The graphs show how many clusters are there containing a certain number of papers.

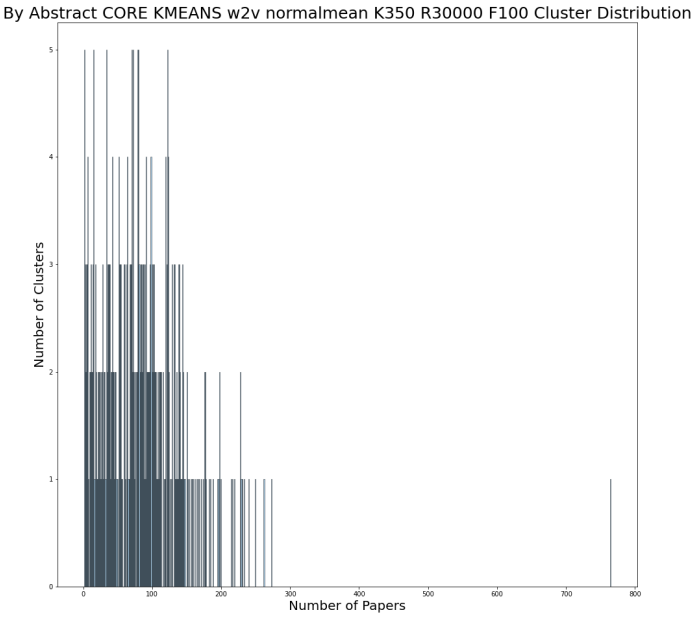
CORE Dataset using Paper Abstracts + Glove + Centroid



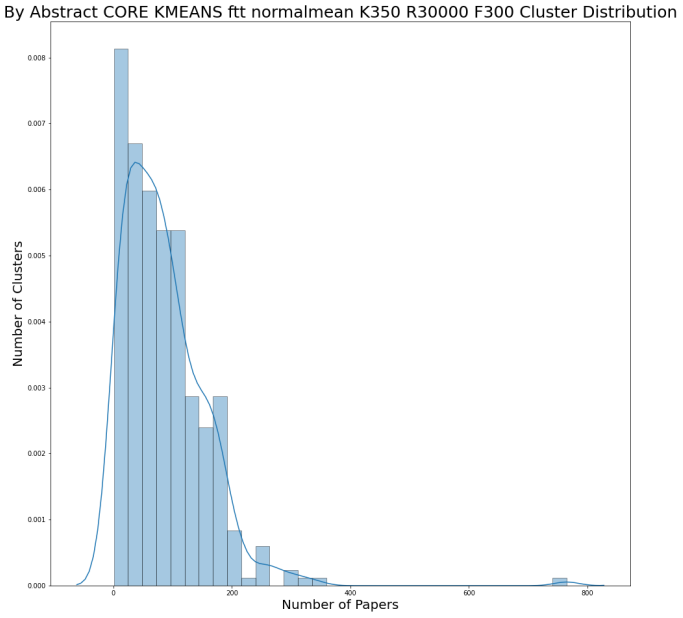


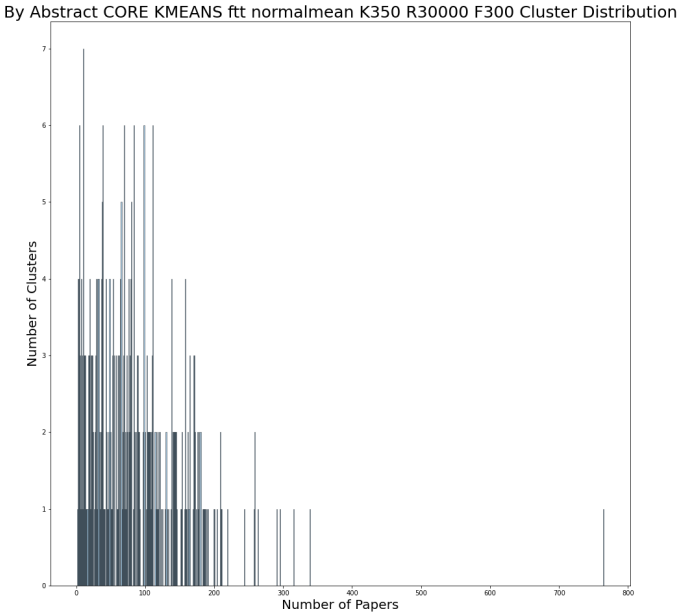
CORE Dataset using Paper Abstracts + Word2Vec + Centroid



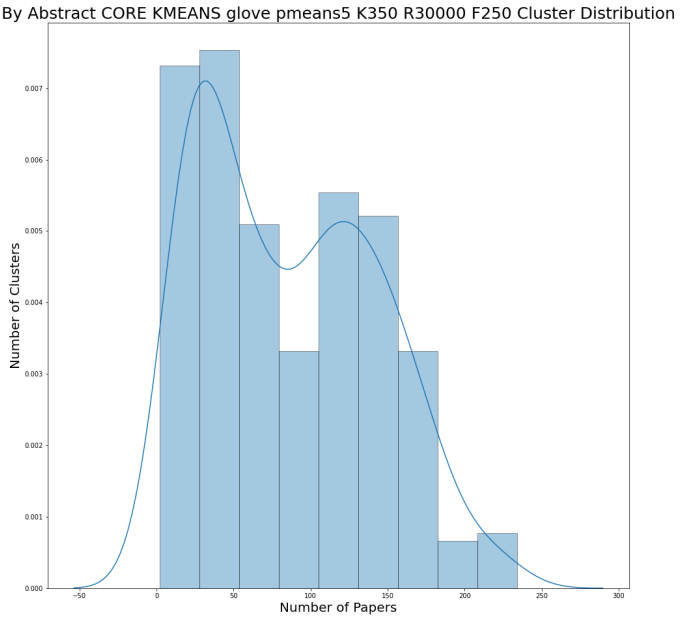


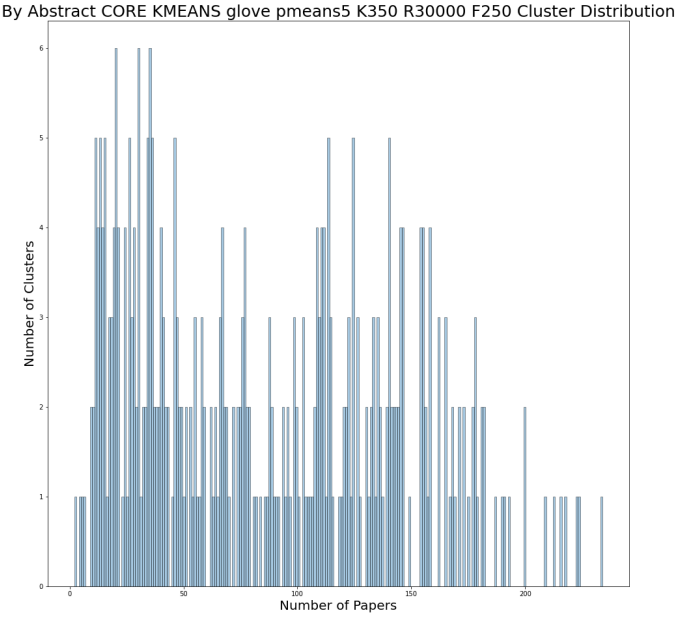
CORE Dataset using Paper Abstracts + FastText + Centroid



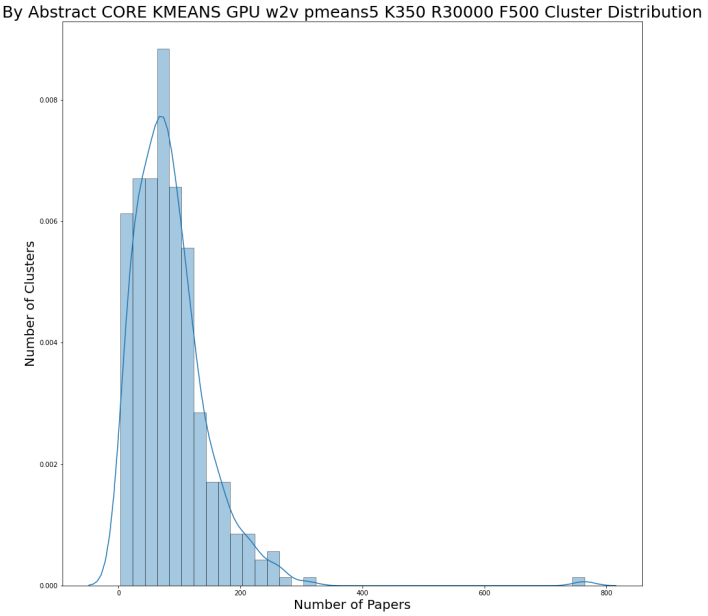


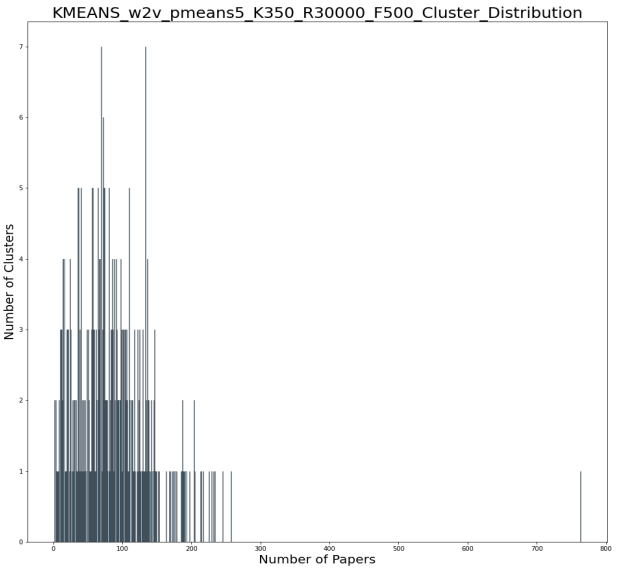
CORE Dataset using Paper Abstracts + Glove + Pmeans



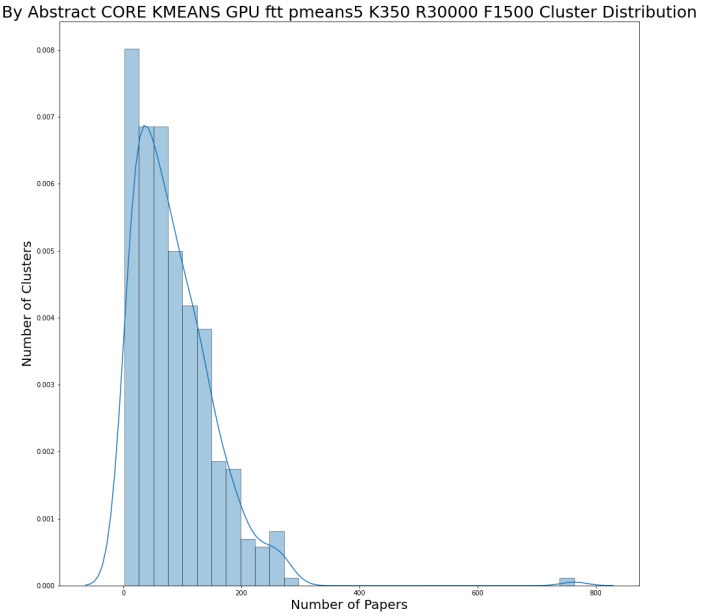


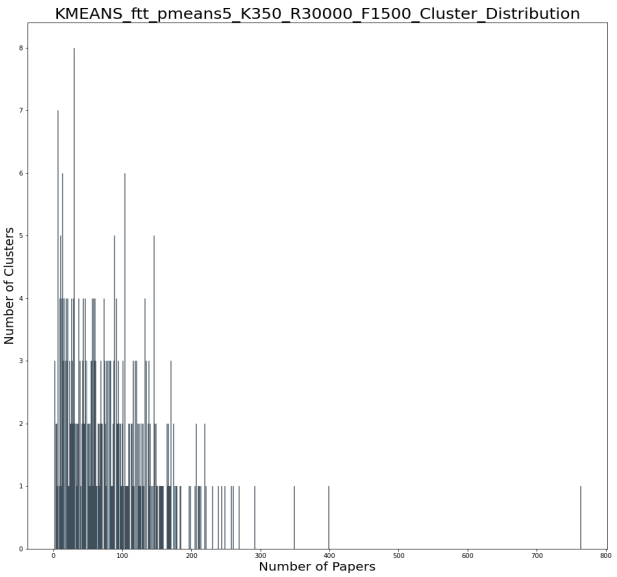
CORE Dataset using Paper Abstracts + Word2Vec + Pmeans



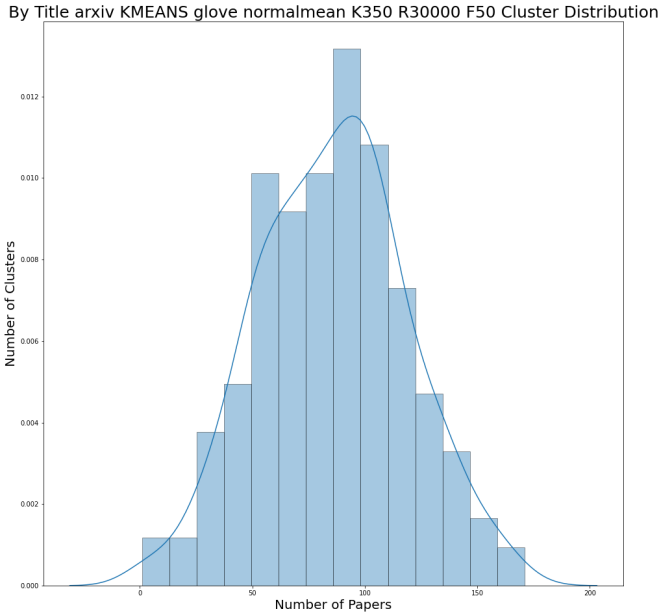


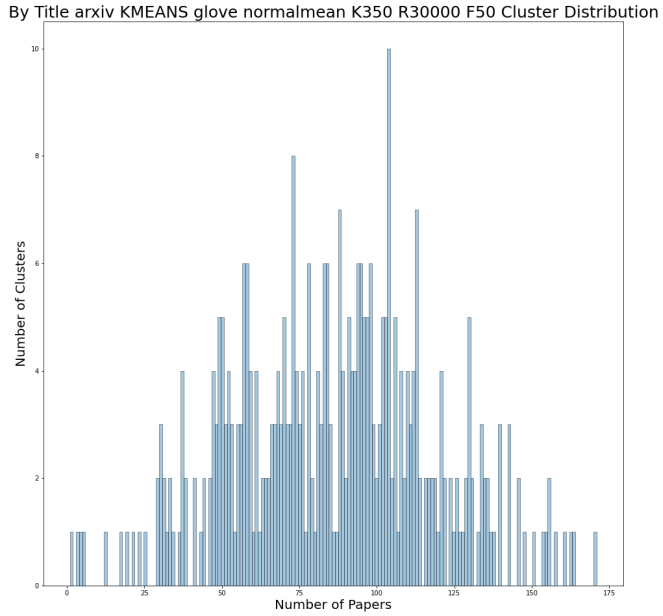
CORE Dataset using Paper Abstracts + FastText + Pmeans



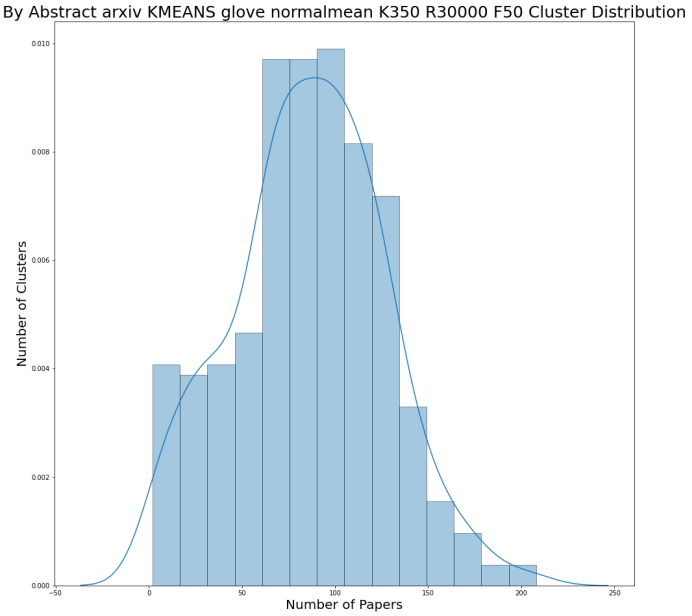


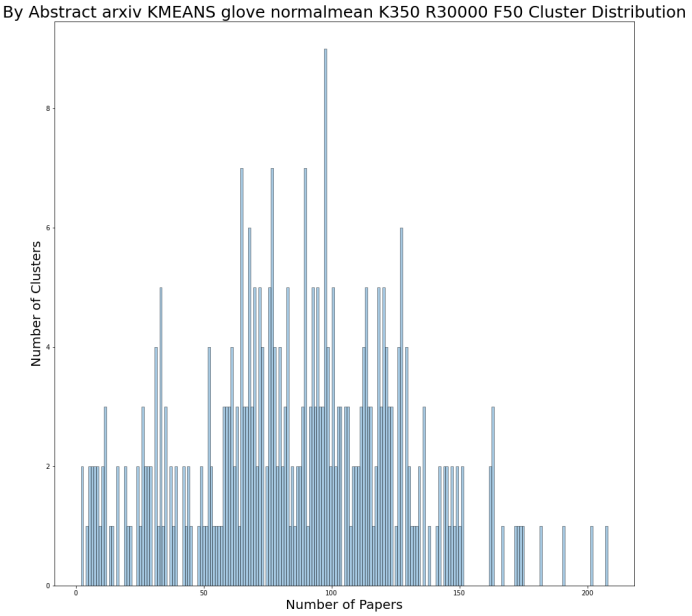
Arxiv Dataset using Paper Titles + Glove + Centroid



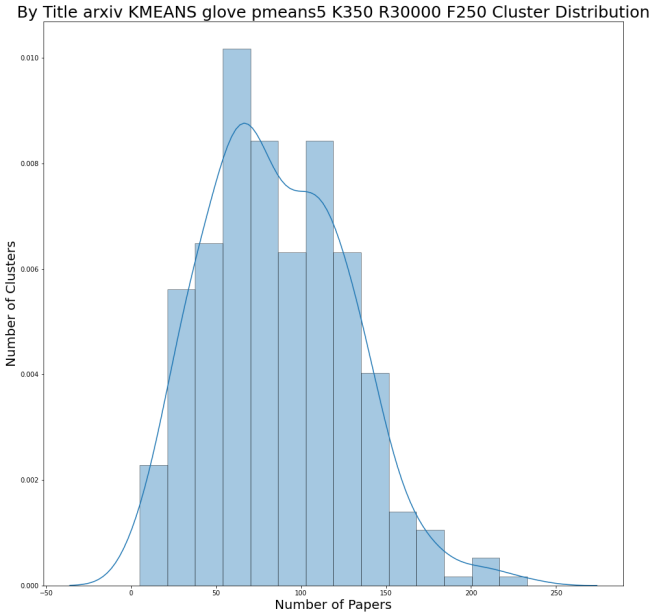


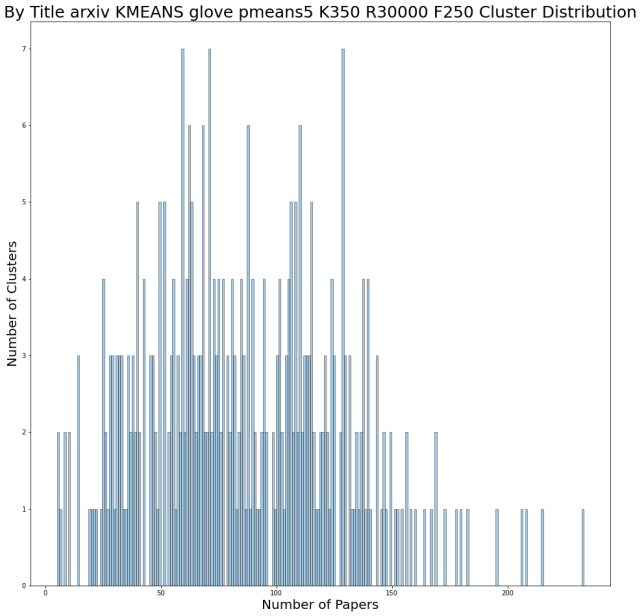
Arxiv Dataset using Paper Abstracts + Glove + Centroid



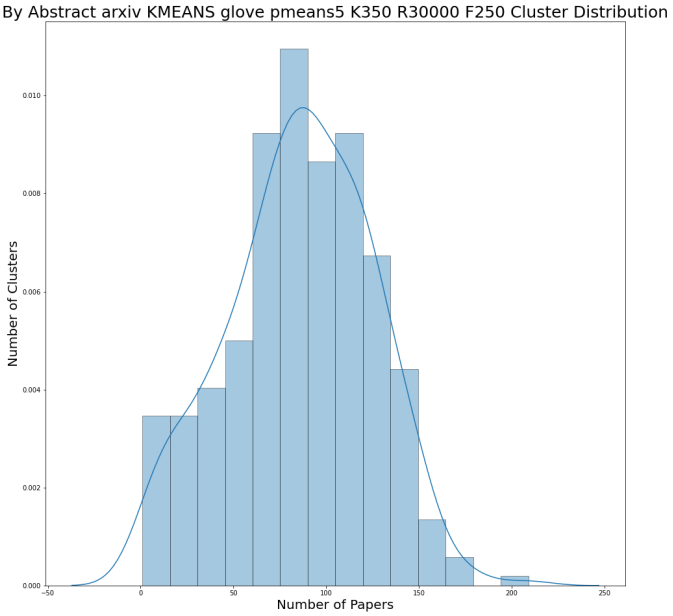


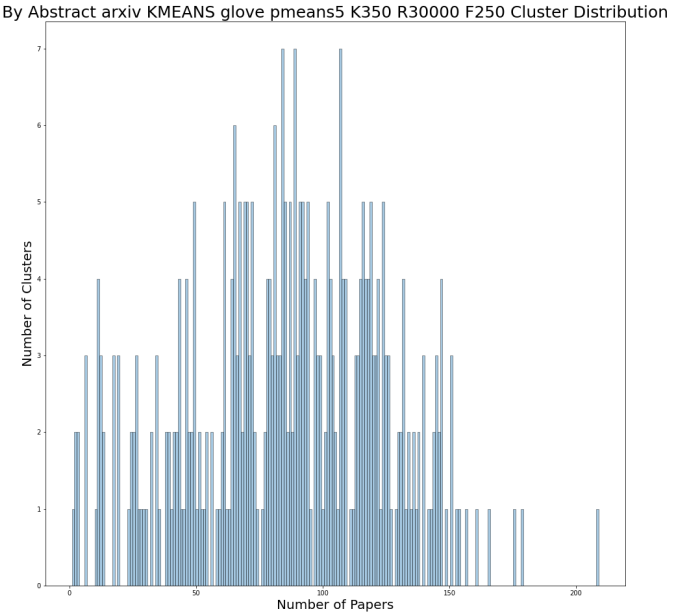
Arxiv Dataset using Paper Titles + Glove + Pmeans



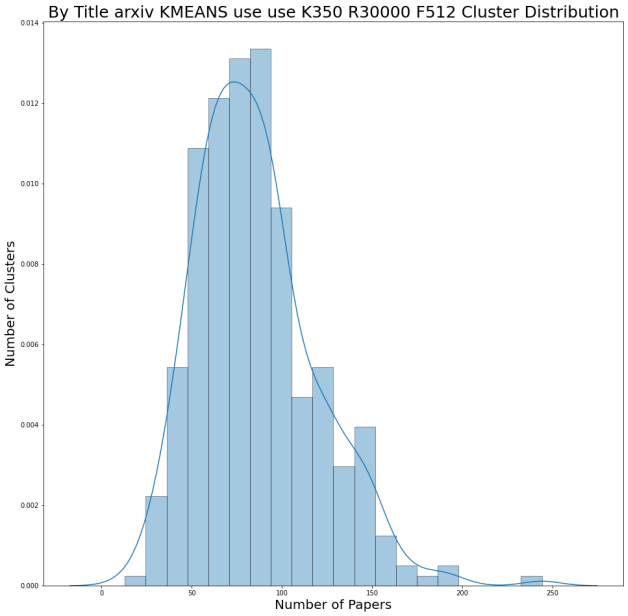


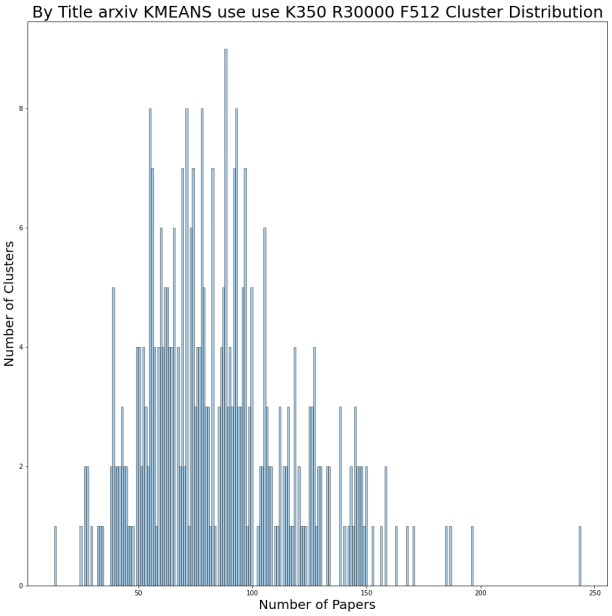
Arxiv Dataset using Paper Abstracts + Glove + Pmeans



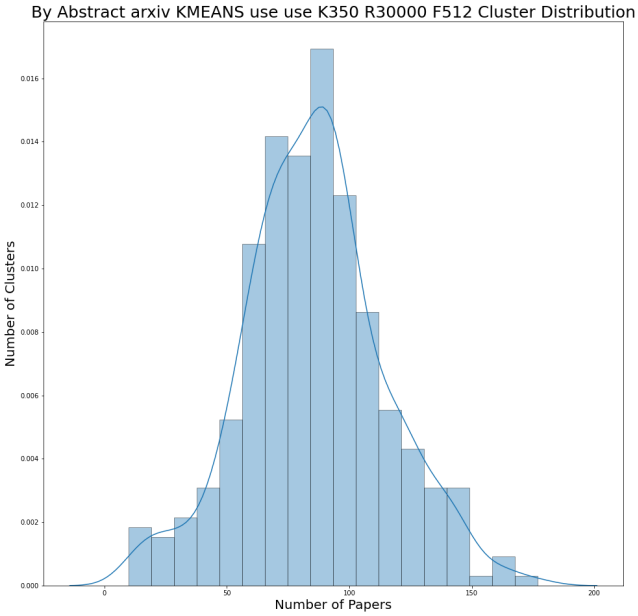


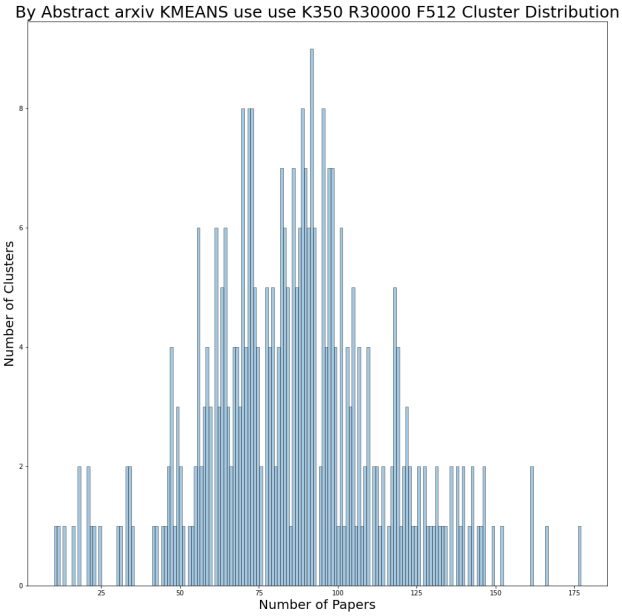
Arxiv Dataset using Paper Titles + Universal Sentence Encoder





Arxiv Dataset using Paper Abstracts + Universal Sentence Encoder





Similarity

Test paper and it's top 2 closest papers from it's cluster are given below.

For CORE Dataset

Word Emb and Sentence Emb.	Paper Titles	Cosine Distance from Test Paper
Test Paper Title	Intracellular mechanisms underlying the nicotinic enhancement of LTP in the rat dentate gyrus	0 (Test Paper title)
GloVe + Centroid	The novel Syk inhibitor R406 reveals mechanistic differences in the initiation of GPVI and CLEC-2 signaling in platelets	0.0486
	An investigation into the role of neurotransmitter receptors in the function of human immune cells	0.0599
GloVe + Pmeans	Phenylephrine preconditioning of isolated ventricular myocytes involves modulation of KATP channels through activation of survival kinases	0.0253
	The tyrosine phosphatase CD148 is an essential positive regulator of platelet activation and thrombosis	0.0254
Word2Vec + Centroid	GLP-1 and Muscarinic Receptor Mediated Activation of ERK1/2 in Pancreatic β-cells	0.0271
	Biochemical investigation of phosphodiesterase type IV post-translational modification, cellular localisation and interaction with associated binding proteins	0.0280
Word2Vec + Pmeans	A population of immature cerebellar parallel fibre synapses are insensitive to adenosine but are inhibited by hypoxia	0.0247
	Excitotoxic ATP and Glutamate Signalling during Central Nervous System Ischaemia	0.0251
Fastext + Centroid	Phenylephrine preconditioning of isolated ventricular myocytes involves modulation of KATP channels through activation of survival kinases	0.0429

	Opposing Changes in Phosphorylation of Specific Sites in Synapsin I During Ca2-Dependent Glutamate Release in Isolated Nerve Terminals	0.0447
Fastext + Pmeans	Scanning peptide array analysis identify overlapping binding sites for the signaling scaffold proteins, beta-arestin and RACK1 in the cAMP-specific phosphodiesterase, PDE4D5	0.0256
	Novel areas of crosstalk between the cyclic AMP and PKC signalling pathways	0.0270

As seen in the Table, The clustered papers are in the same field of literature and the cosine distance between the closest research papers is low.

For Arxiv Dataset

Word Emb and Sentence Emb	Paper Titles (Refer Appendix for full Abstracts)	Cosine Distance from Test Paper
Test Paper Title	High-Resolution Semantic Labeling with Convolutional Neural Networks	0 (Test Paper title)
USE	Recent approaches for instance-aware semantic labeling have augmented convolutional neural networks (CNNs) with complex multi-task	0.258
	Multi-label learning has attracted significant interests in computer vision recently, finding applications in many vision tasks	0.291
GloVe + Pmeans	The development of powerful 3D scanning hardware and reconstruction algorithms has strongly promoted the generation of	0.0157
	Exploiting relationships among objects has achieved remarkable progress in interpreting images or videos by natural language.	0.0172

Conclusion

Older and more tried, researched clustering techniques such as Co-Citation Analysis, Bibliographic Coupling, and Direct Citation relations are better for unsupervised clustering of research papers as compared to Clustering using our proposed technique. Clustering using Natural Language Processing techniques such as keywords, LSI, LDA, topic modeling also give better statistical outcomes than our proposed method. Unsupervised Clustering using Concatenated Power Means and Centroids does not give sufficient statistical score to be deemed viable for unsupervised clustering of research papers. It is more viable for supervised clustering as it gives good, logical similarity scores for research papers in the same cluster, but the nature of the dataset makes it harder for unsupervised clustering to be statistically viable.

The main drawback of using sentence embeddings is the very high dimensional representation of the underlying text. This is useful for other downstream tasks such as classification, sentiment analysis, etc but it is not useful for clustering because high dimensional data makes clustering inherently hard as points are spaced far apart from each other.

This unsupervised clustering method can be used to fingerprint the datasets using their cluster distribution. The unsupervised clustering shows the underlying clusters of research papers which can not be determined by simply grouping papers together by keywords or citation links as it uses the semantic meaning of the titles and abstracts for clustering.

Future Work

Word embeddings and sentence embeddings of reduced size can be used to cluster the papers, dimensionality reduction techniques such as Principal component analysis, Normalization and scaling to redistribute the data points might also prove useful. Other newer models such as Doc2Vec can also be used to represent the whole document instead of just abstract or title. Neural network based sentence embedding models such as Universal Sentence Encoder or Infersent can be used. Concatenated Power means should be tested on a labeled dataset i.e. supervised clustering, to get accuracy, precision, and recall scores to create a comparable metric. Reduction in dimensions of representation embeddings will also allow for rapid distribution fingerprinting.

References

[1] S. Arora, Y. Liang, and T. Ma. A simple but tough-to-beat baseline for sentence embeddings.2017.

[2] A. Bakarov. A survey of word embeddings evaluation methods. arXiv preprint arXiv:1801.09536, 2018.

[3] Y. Bengio, R. Ducharme, P. Vincent, and C. Janvin. A Neural Probabilistic Language Model. The Journal of Machine Learning Research, 3:1137–1155, 2003. ISSN 15324435.doi:10.1162/153244303322533223. http://www.jmlr.org/papers/volume3/bengio03a/bengio03a.pdf.

[4] P. Bojanowski, E. Grave, A. Joulin, and T. Mikolov. Enriching word vectors with subword information. Transactions of the Association for Computational Linguistics, 5:135–146, 2017.

[5] J. Camacho-Collados and M. T. Pilehvar. From Word to Sense Embeddings: A Survey onVector Representations of Meaning. URLhttps://arxiv.org/pdf/1805.04032.pdf.

[6] D. Cer, M. Diab, E. Agirre, I. Lopez-Gazpio, and L. Specia. Semeval-2017 task 1: Semantic textual similarity-multilingual and cross-lingual focused evaluation.arXiv preprintarXiv:1708.00055, 2017.

[7] D. Cer, Y. Yang, S. Kong, N. Hua, N. Limtiaco, R. S. John, N. Constant, M. Guajardo-Cespedes,S. Yuan, C. Tar, Y. Sung, B. Strope, and R. Kurzweil. Universal sentence encoder.CoRR, abs/1803.11175, 2018. http://arxiv.org/abs/1803.11175.

[8] A. Conneau and D. Kiela. Senteval: An evaluation toolkit for universal sentence representations.arXiv preprint arXiv:1803.05449, 2018.

[9] A. Conneau, D. Kiela, H. Schwenk, L. Barrault, and A. Bordes. Supervised learning of universal sentence representations from natural language inference data. InEMNLP, pages 670–680.Association for Computational Linguistics, 2017.

[10] A. Conneau, G. Kruszewski, G. Lample, L. Barrault, and M. Baroni. What you can cram into a single vector: Probing sentence embeddings for linguistic properties.arXiv preprintarXiv:1805.01070, 2018.

[12] B. Dolan, C. Quirk, and C. Brockett. Unsupervised construction of large paraphrase corpora:Exploiting massively parallel news sources. InProceedings of the 20th international conference on Computational Linguistics, page 350. Association for Computational Linguistics, 2004.

[13] F. Hill, K. Cho, and A. Korhonen. Learning distributed representations of sentences from unlabelled data. In2016 Conference of the North American Chapter of the Association forComputational Linguistics: Human Language Technologies, NAACL HLT 2016 - Proceedings of the Conference, pages 1367–1377. Association for Computational Linguistics (ACL), 2016.

[14] J. Howard and S. Ruder.Fine-tuned language models for text classification.CoRR,abs/1801.06146, 2018.13

[15] M. Hu and B. Liu. Mining and summarizing customer reviews. InProceedings of the tenth ACMSIGKDD international conference on Knowledge discovery and data mining, pages 168–177.ACM, 2004.

[16] M. Iyyer, V. Manjunatha, J. Boyd-Graber, and H. Daumé III. Deep unordered composition rivals syntactic methods for text classification. InProceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference onNatural Language Processing (Volume 1: Long Papers), volume 1, pages 1681 -1691, 2015.

[17] A. Joulin, E. Grave, P. Bojanowski, and T. Mikolov. Bag of tricks for efficient text classification.arXiv preprint arXiv:1607.01759, 2016.

[18] D. P. Kingma and J. Ba.Adam: A method for stochastic optimization.CoRR, abs/1412.6980, http://dblp.uni-trier.de/db/journals/corr/corr1412.html#KingmaB14.

[19] R. Kiros, Y. Zhu, R. R. Salakhutdinov, R. Zemel, R. Urtasun, A. Torralba, and S. Fidler. Skip-thought vectors. InAdvances in neural information processing systems, pages 3294–3302,2015.

[20] Q. Le and T. Mikolov. Distributed representations of sentences and documents. InInternationalConference on Machine Learning, pages 1188–1196, 2014.

[21] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick.Microsoft coco: Common objects in context. InEuropean conference on computer vision, pages740–755. Springer, 2014.

[22] M. Marelli, S. Menini, M. Baroni, L. Bentivogli, R. Bernardi, R. Zamparelli, et al. A sick cure for the evaluation of compositional distributional semantic models. InLREC, pages 216–223,2014.

[23] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean. Distributed representations of words and phrases and their compositionality. InAdvances in neural information processing systems, pages 3111–3119, 2013.

[24] M. Pagliardini, P. Gupta, and M. Jaggi. Unsupervised Learning of Sentence Embeddings usingCompositional n-Gram Features. InNAACL 2018 - Conference of the North American Chapterof the Association for Computational Linguistics, 2018.

[25] J. Pennington, R. Socher, and C. D. Manning. Glove: Global vectors for word representation. InEmpirical Methods in Natural Language Processing (EMNLP), pages 1532–1543, 2014. URLhttp://www.aclweb.org/anthology/D14-1162.

[26] M. E. Peters, M. Neumann, M. Iyyer, M. Gardner, C. Clark, K. Lee, and L. Zettlemoyer. Deep contextualized word representations. InProc. of NAACL, 2018.

[27] A. Rücklé, S. Eger, M. Peyrard, and I. Gurevych. Concatenated p-mean word embeddingsas universal cross-lingual sentence representations.CoRR, abs/1803.01400, 2018. http://arxiv.org/abs/1803.01400.

[28] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, andI. Polosukhin. Attention is all you need. InAdvances in Neural Information Processing Systems,pages 6000–6010, 2017.