AMSI VACATION RESEARCH SCHOLARSHIPS 2019–20

EXPLORE THE MATHEMATICAL SCIENCES THIS SUMMER

Embeddings for Detecting Outliers

Daniel Stratti
Supervised by Laurence Park
Western Sydney University

Vacation Research Scholarships are funded jointly by the Department of Education and the Australian Mathematical Sciences Institute.

AMSI AUSTRALIAN MATHEMATICAL SCIENCES INSTITUTE

Australian Government Department of Education

Contents

Abstract

A ranking containing ordered items from a set is often required for events such as awarding prizes (ranking the participants submissions), grants (ranking the grant applications), and accepting papers into a conference (ranking the papers). In all of these applications the rank is usually aggregated from a set of experts review of a literature submission. Existing research into ranking precision has shown the time each expert spends per submission has a large effect to the overall precision of the final aggregated rank. This project investigates the utility of word embeddings for projecting literature submissions into high a dimensional space in order to cluster similar papers, allocate papers to relevant experts as well as identify outlier subject matter addressed by papers. This analysis builds the basis for a human in the loop AI system to optimize the reviewal process.

Introduction

Many events such as awards, grants & conferences require a panel of experts to review and rank individual submissions. Rarely is it possible to have a single set of experts review each individual submission due to time constraints.

To combat this, a usual approach is to have a subset of experts review a subset of submissions and then aggregate the resulting ranks to form a more accurate representation of the true rank of each submission. Park & Stone (2015) identified that a major influencing factor contributing to the overall precision of this rank aggregation was the amount of time each reviewer was able to spend reviewing the submission.

This insight was the inspiration of this project, which involved first identifying points along the reviewal process that could be altered to increase the time each expert has available for each submission. The process of reviewing conference papers was examined due to the availability of conference papers, and the insight into the reviewal process the academic supervisor of this project could provide.

In the general case of a conference, a fixed number of papers are assigned to an expert for review during a fixed time period. The papers may be assigned based on expert preferences constrained by the submission coverage. The assigned submissions usually cover similar topic areas at a high level, however, can also vary in subject matter greatly. As Identified by Zhang et al. (2020) this variation can lead to an unnecessary increase in the time spent reviewing each submission

This project focuses on two areas of the reviewal process. The allocation of a set of submissions to most relevant expert(s) & the identification of submissions within a set which subject matter varies most from all other submissions within that set. Unlike the work of Zhang et al. (2020) which focus on the allocation of papers based on a multi-label hierarchical classification, the project focuses on the use of word embeddings to reduce any time spent on additional research that would be required for the expert to fully comprehend the addressed subject matter of a submission.

This is accomplished by projecting the literature submissions into high dimensional vector space. The investigation also suggests that for each expert, their own publications are also projected into the same vector space and the cosine similarity between the

submissions and the experts vector space are used to assign relevant papers. Similarly, the cosine similarity between one submission and all others within an assigned set can be used to establish a statistic in order to identify how similar the addressed topic matters are to one another (Triwijoyo & Kartarina, 2019).

1.1 Statement of Authorship

The supervisor of the project, Dr Laurence Park, contributed to the algorithm selection and guided the implementation, and creation of the similarity statistic. All data collection, development of code, experimentation and analysis was completed by the author, Daniel Stratti.

2. Methodology

2.1 Data Collection and Pre-processing

To complete the experimentation and analysis of the project, 1404 papers were scrapped from the "ArXiv Computer Science" (2020) website. Each paper scrapped belonged to between 1 and 5 categories as labelled by the arxiv.org website and each document would be used to simulate a paper submitted to a conference.

Each of the documents were prepared for analysis using the following process. The text was extracted from each document and all stop words defined in the Gensim: Text Preprocessing (n.d.) python package were removed. Using the same package, the documents were then tokenised and lemmatized to reduce all words to the base definition of each. The documents were then split into training and test sets following a 70:30 ratio and ready for algorithmic vectorisation.

2.2 Vectorisation

The foundation of the proposed analysis hinges on an algorithmic ability to encode the subject matter of a literature submission into a high dimensional vector space. With recent traction in the area of Natural Language Processing (NLP) Moody (2016), suggests that vectorisation methods that use word embeddings and topic modelling can successfully encode the underlying meaning of unstructured literature documents.

Based off this the below four methods of vectorisation were assessed, three of which are unsupervised techniques for extracting word, document and topic embeddings. The Term frequency inverse document frequency (TF-IDF) algorithm was also assessed as a baseline to measure if any improvement was introduced via the use of embeddings.

- Word2Vec, (Mikolov, Chen, Corrado, & Dean, 2013)
- Latent Dirichlet Allocation, (Blei, Ng, & Edu, 2003)
- Doc2Vec, (Le & Mikolov, 2014)
- TF-IDF, (Jones, 1972)

In order to compare the performance of each algorithms ability to encoding a documents subject matter, the K-nearest neighbours (K-NN) algorithm was used with the target labels specified via the arxiv.org website. A rigged test set containing similar papers with known outliers was then be used to assess the performance of each vector space in the identification outlier documents.

It is important to note that using the labels provided by the arxiv.org website could introduce bias into the experiments and the results are dependent on the accuracy of classification given via the arxiv.org website. The experiments conducted within this project follow the assumption that these target labels are accurate.

In order to visualise the high dimensional vector spaces produced by each algorithm, multi-dimensional scaling was used to reduce the vector spaces into three dimensions. This was only used as a visual representation and all analysis was done in the higher dimensions.

2.2.1 Word2Vec

The Word2Vec algorithm was originally proposed by Mikolov, Chen, Corrado, & Dean (2013) with famously successful results in encoding the meanings of words within a high dimensional vector space. The authors showed that the word2vec vectors encode relationships between items as their vector difference, i.e. taking the vector representing king, minus the vector representing man and add the vector representing woman, would result in a vector similar to that of the vector for the word queen.

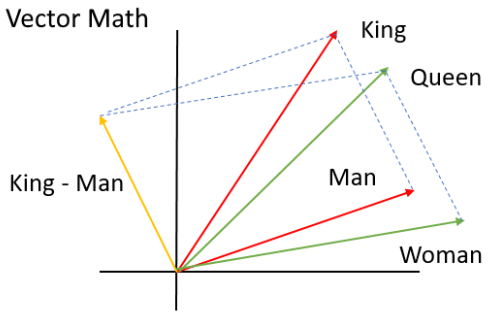

Figure 1: ("VectorMath.png (530×340)," n.d.)

This result suggests that the underlying meaning of words (such as gender), is encoded into the vector space and this was an influencing factor in the algorithm selection for this project.

The skip-gram form of word2vec was examined in this project (Mikolov et al., 2013). The vectors for each specified term $\boldsymbol{J(\theta)}$ are constructed by maximising the likelihood of a context term $\boldsymbol{w_{t+j}}$ given the specified term $\boldsymbol{w_t}$. The context terms are bound within a window $\boldsymbol{m}$ of terms surrounding the specified term as shown in the below formulation:

$$P(w_{t+j}|w_t;\theta) = \frac{e^{(w_t \cdot w_{t+j})}}{\sum_{j=1}^{N} e^{(w_t \cdot w_{t+j})}}$$

$$L(\theta) = \prod_{t=1}^{T} \prod_{\substack{-m \le j \le m \\ j \neq 0}} P(w_{t+j}|w_t;\theta)$$

$$J(\theta) = -\frac{1}{T} \log L(\theta)$$

In order to produce a single vector representative of the entire document, the average of the word vectors within a document was taken

$$\frac{1}{N}\sum_{i=1}^{N} J(\theta)_i$$

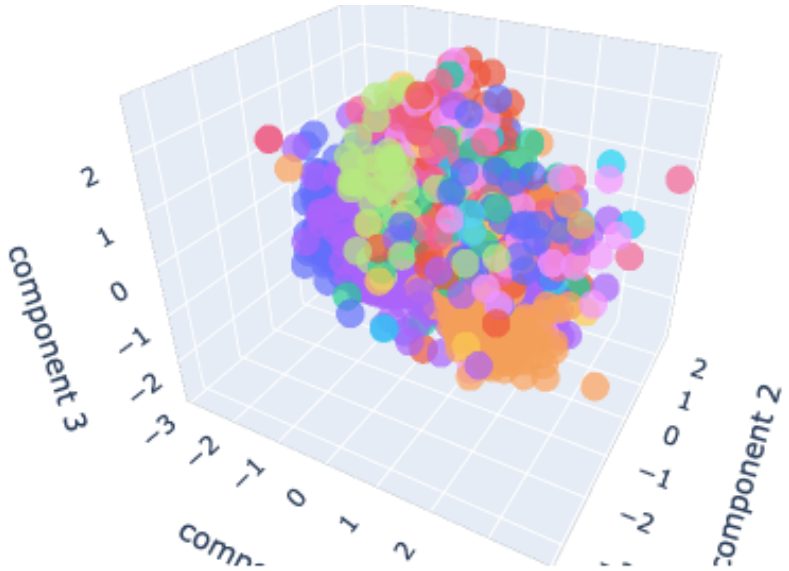

Figure 2: Word2Vec 3D Vector Space (Daniel Stratti, 2020)

Figure 2 shows the Word2Vec document vector space reduced to three dimensions, each point is coloured based on the document target label given by the arxiv.org website. The visualisation shows some clear clustering of documents, in particular those that are orange, purple, green and blue which are representative of the categories Numerical Analysis, Machine Learning, Computation & Language and Computer Vision respectively. From a purely visual assessment the algorithm seems to encode the subject matter addressed by each document quite well and produce appropriate clusters

2.2.2 Latent Dirichlet Allocation

Latent Dirichlet Allocation (LDA) was introduced by Blei et al. (2003) as a method for deriving the distribution of topics a given document contained $\boldsymbol{P(t|d)}$ based on the distribution of terms $\boldsymbol{P(w|t)}$ used within the document as shown in the below formulation:

$$\sum_{t=1}^{T} P(w|t)\, P(t|d)$$

The use of LDA within the work of Zhang et al. (2020) for multi-label classification and by Moody, (2016) for defining context of word embeddings were an influencing factor on the algorithms selection for this project.

In order to produce a vector representation of a single document, the derived distribution of topics a document belonged to was encoded into a vector of dimensions equal to the total number of topics $\boldsymbol{n}$ in a manner similar to one hot encoding

$$X_i = [t_1, t_2, \ldots, t_n]$$

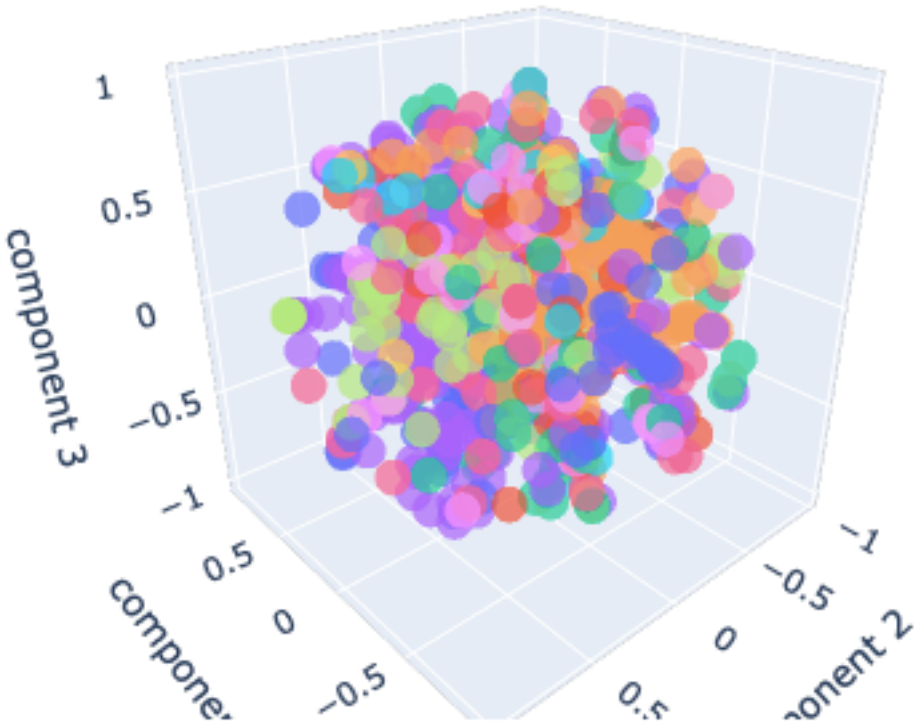

Figure 3: LDA 3D Vector space (Daniel Stratti, 2020)

Figure 3 is the LDA vector space projected down to 3 dimensions following the same colour scheme. Unlike the Word2Vec vector space, when projected down to three dimensions the LDA algorithm has visibly less defined clusters. Although there are still groups of orange (Numerical Analysis) and purple (Machine Learning) the overall visualisation depicts LDA as producing a less segregated vector space.

2.2.3 Document Embedding with Paragraph Vectors

The Doc2Vec algorithm built off the success of Word2Vec and incorporates the information of the document a term occurs into the embedding, this was an influential factor in the algorithm's selection for this project.

First proposed by Le & Mikolov (2014) the Distributed Memory Paragraph Vector (DM-PV) implementation of the Doc2Vec algorithm was utilised. The vectors for each specified document $\boldsymbol{L}$ are constructed by maximising the likelihood of a specified term within the document $\boldsymbol{w_{t,j}}$ given the surrounding context words within that document $\boldsymbol{w_{t-k,j}}$. Again, just like in the Word2Vec algorithm, the context terms are bound within a window $\boldsymbol{k}$ of terms surrounding the specified term as shown in the below formulation:

$$y = b + Uh(w_{t-k,j}, \ \dots, w_{t+k,j}; W)$$

$$P\left(w_{t,j} \middle| w_{t-k,j}, \ \dots, w_{t+k,j}; d_j\right) = \frac{e^{y_{w_t}}}{\sum_{j=1}^{N} e^{y_i}}$$

$$L = \frac{1}{J}\sum_{j=1}^{J} \frac{1}{N_j - 2k} \sum_{t=k}^{N_j - k} \log P\left(w_{t,j} \middle| w_{t-k,j}, \ \dots, w_{t+k,j}; d_j\right)$$

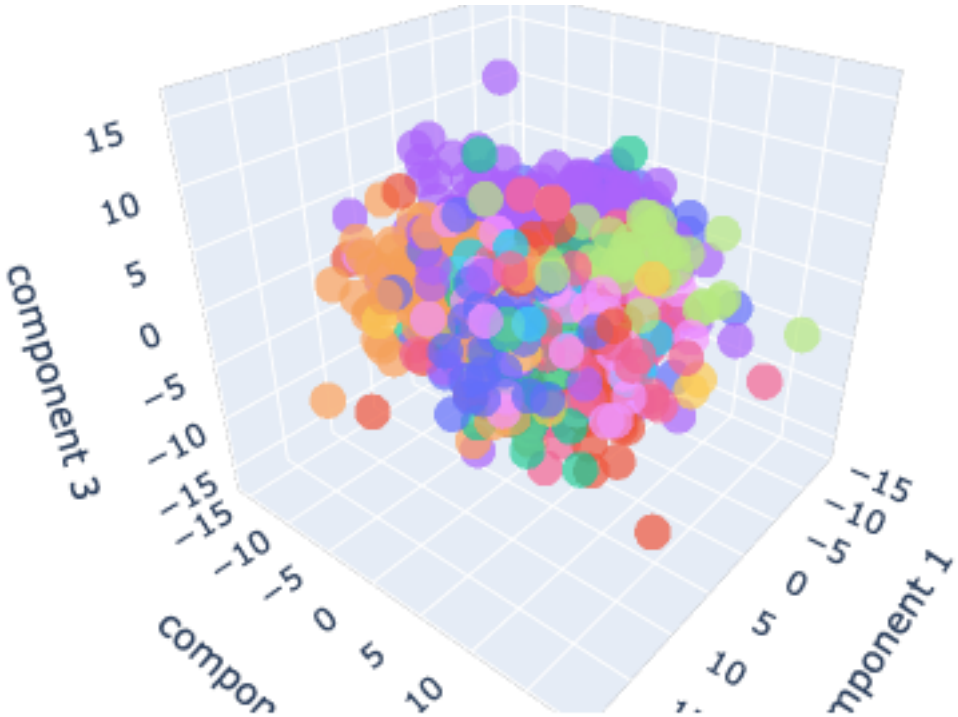

Figure 4: Doc2Vec 3D Vector Space (Daniel Stratti, 2020)

Figure 4 is the Doc2Vec document vector space reduced to three dimensions, following the same colour scheme as the previous visualisation. The visualisation, just like Word2Vec shows some clear clustering of documents, in particular those that are orange (Numerical Analysis), purple (Machine Learning), green (Computation & Language) and blue (Computer Vision). Based on a visual assessment of the three algorithms presented so far, it seems that using word embedding improves the segregation of categories compared to topic modelling. However, as these visualisation are reduced to three dimensions via multi-dimensional schilling, some information is likely not visible.

2.2.4 Term Frequency Inverse Document Frequency

The TF-IDF algorithm first introduced by Jones (1972), is a widely implemented and relatively simple algorithm in terms of computational complexity. The implementation of TF-IDF is a supervised approach to creating document vectors, which should not encode the underlying meaning of words as suggested by the word embedding approaches. For this reason, the algorithm was selected as a baseline measurement that any difference in using the word embeddings could be compared against.

The TF-IDF algorithm constructs a document vector of dimensions $\boldsymbol{X}$ where $\boldsymbol{X}$ is the total number of distinct terms within the corpus of documents $\boldsymbol{D}$ the algorithm is trained on. The resulting term weights $\boldsymbol{w_{i,j}}$ of each element within the vector is shown in the below formulation:

$$w_{i,j} = \frac{n_{i,j}}{\sum_k n_{i,j}} * \ log \frac{N}{|\{d \in D: t \ \in d\}|}$$

- $n_{i,j}$: The frequency of term i in document j

This forces terms that occur in many documents to have a low weight whilst terms that occur in few documents to have a larger weight.

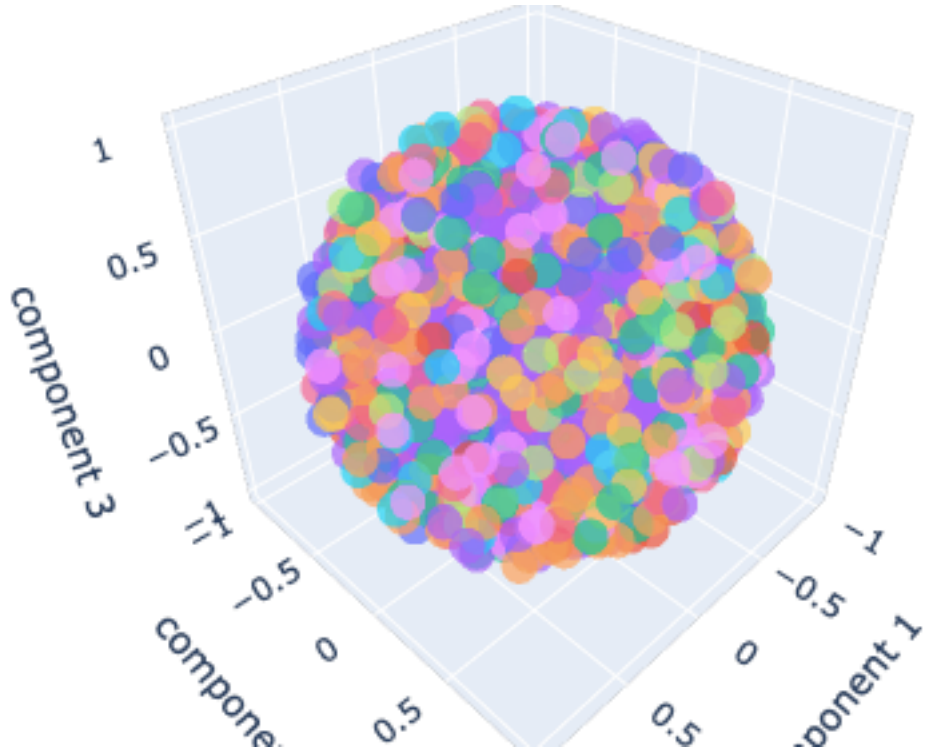

Figure 5: TF-IDF 3D Vector Space (Daniel Stratti, 2020)

Figure 5 shows the TF-IDF document vector space reduced to three dimensions, following the same colour scheme. Unlike the unsupervised embedding techniques, the vector space shows no identifiable clusters, nor seems to be segregating the documents in any intelligible manner. It is however important to note that the TF-IDF algorithm produces vectors with a very high dimension; equal to the length of all distinct terms within a corpus. Reducing a vector of this large dimensionality to a mere three dimensions, may cause all intelligent information to be lost.

2.3 Vector-space Validation

In order to quantify the performance of each algorithm in terms of encoding the underlying subject matter addressed by each document, the K-NN algorithm was used with a slight adaptation.

To address the multi-categorical nature of the target labels for each individual document, the implemented K-NN algorithm located the $\boldsymbol{k}$ closest neighbours as defined by the cosine similarity between a specified document $\boldsymbol{x}$ and all other documents within the corpus $\boldsymbol{D}$.

Cosine similarity was chosen as a distance metric as the “direction” the vector pointed in the high dimensional space was deemed more indicative of the addressed subject matter than the magnitude of the vector. The categories each neighbour belonged

to were then aggregated and sorted by the summation of the cosine similarity of each of the neighbour documents the categories belonged to.

Frequency	Category	Total Similarity Score
12	Robotics (cs.RO)	9.136819841535504
7	Machine Learning (cs.LG)	5.227040657481446
7	Artificial Intelligence (cs.AI)	5.174058726961814

Table 1: Example of K-NN Multi-categorical Classification (Daniel Stratti, 2020)

For instance, a document belonging to the three categories Robotics (cs.RO), Artificial Intelligence (cs.AI) and Systems & Control (eess.SY), the above classification shown in table 1. was produced by the implemented K-NN when examining the Doc2Vec vector space and 15 neighbours. As shown below in figure 6, the choice to use 15 neighbours was based on brute force assessment of classification accuracy from 2 – 30 neighbours.

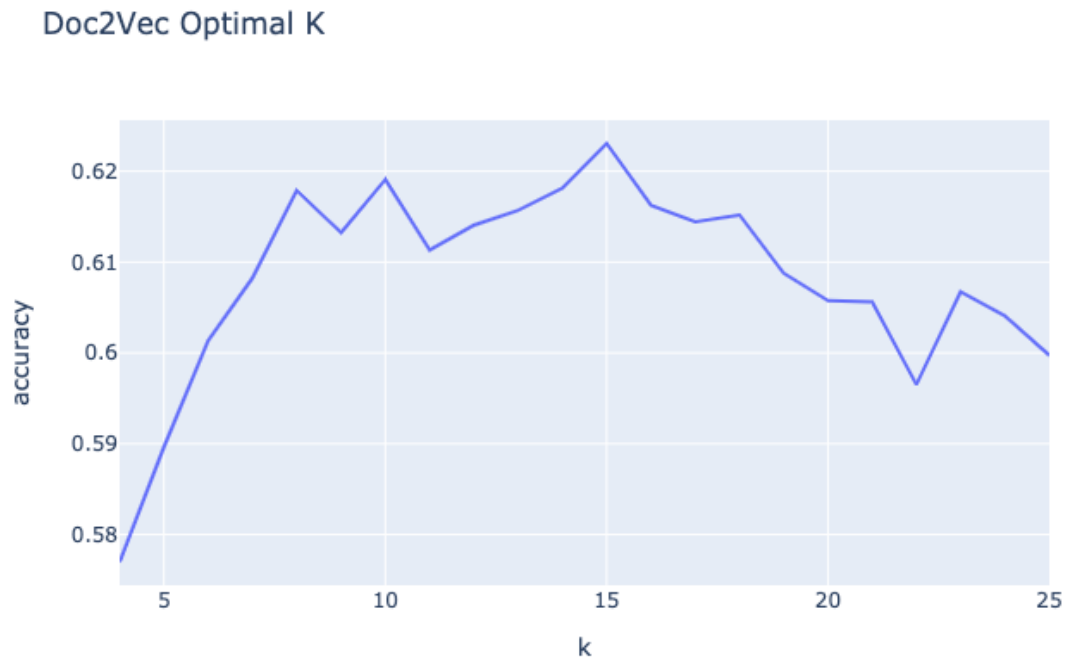

Figure 6: Optimal K (Daniel Stratti, 2020)

2.2.1 Validation Results

Category Count/paper	No. papers	TF-IDF (%)	Word2Vec (%)	LDA (%)	Doc2Vec (%)
1	152	58.55	56.58	12.50	59.21
2	150	50.67	54.00	22.67	60.33
3	100	59.00	60.00	42.33	60.67
4	29	65.52	68.10	48.28	67.24
5	10	66.00	54.00	42.00	66.00
	441	**59.9**	**58.5**	**33.6**	**62.7**

Table 2: K-NN Classification Results (Daniel Stratti, 2020)

Based on the implementation of the K-NN algorithm used, the classification is not binary and thus the results must be interpreted as, on average a vector-space from the algorithms above (excluding LDA) will accurately project a document into an area surrounded by ~60% of documents dealing with similar subject matter.

After executing the K-NN classification the results showed the Doc2Vec algorithm to produce the most accurate vector representations of a document, in terms of encoding the underlying subject matter the document addresses. However, the results of TF-IDF and Word2Vec are extremely close to the best performer with LDA performing the worst overall.

As the results of the TF-IDF, Word2Vec and Doc2Vec algorithms were so close a one way ANOVA test was conducted with the null hypothesis that there is no difference in means. Interestingly, the test produced a p-value of ~0.35 showing that the difference in average classification was negligible and that the null hypothesis could not be rejected.

Based on these results, none of the three algorithms performance was significantly better. This would suggest that the algorithm of least computational intensity, TF-IDF, should be used when projecting documents into a high dimensional vector space for comparison. However, it is important to note that both the Word2Vec and Doc2Vec algorithms are highly affected by the coverage of language used within the training set. A much larger test must be conducted before ruling out the word embedding advantage.

2.2.2 Outlier Statistic

To assess the outlier detection capabilities of the vector space created by each algorithm, a rigged test set was created with known outliers. This test set consisted of 30 Information Theory papers, 33 Network and Internet Architecture papers and 3 Quantum Physics papers. The outlier statistic S_i used to rank each paper was the average cosine similarity of paper i in respect to all other papers with the test set as shown in the below formulation:

$$\cos\theta_j = \frac{\vec{a}\cdot\vec{b}}{|\vec{a}|\,|\vec{b}|}$$

$$S_i = 1 - \frac{1}{N}\sum_{j=1}^{N} |\cos\theta_j|$$

The Doc2Vec algorithm again outperformed the others correctly identifying two of the three quantum physics papers as the top two outliers, the third paper was ranked seventh. All other algorithms did not correctly identify the outlier papers within the top 10 ranks.

3. Conclusion

In the future events such as awards, grants & conferences that require a panel of experts to review and rank individual submissions may optimise the reviewal process. This project suggests that this can be achieved via encoding the subject matter of each submission into a high dimensional vector space.

The results of the above AMSI vocational project suggest that the comparison of using word embeddings to extract the subject matter of a literature submission as opposed to traditional term frequencies yields no significant difference. This in turn suggests that an expert's own literature could be projected into the same vector space and assigned papers based on those most similar to the expert's vector space using the more efficient to compute algorithm (TF-IDF in this case).

Experiments must be conducted to observe if this form of allocation will in fact reduce the time spent in additional research by an expert, and unfortunately fell outside of the scope of this project. If experiments are conducted and time is reduced, based of the work of Park & Stone, (2015) an increase in the overall rank precision can also be expected.

The K-NN & ANOVA test results show that there is no significant improvement of clusters introduced via the use of word embeddings. However, it is important to note that additional research and testing with a larger training set is required before ruling out the embedding advantage.

The identification of outlier papers was shown to be possible using a combination of document and word embeddings. This metric could be used to allow the expert to make an informed decision in the order they choose to review papers and how they allocate their time when doing so. It is important to note that more extensive testing is required.

In conclusion it is highly probable that in the conferences of the future, the allocation of papers for reviewal are done so in a way that is more beneficial to the expert reviewing and the author submitting.

3.1 Future Work

Following up from the project and in addition to the mentioned experiments above, it would of interest to evaluate the effect of optimising the vector spaces with algorithms such as Largest Margin Nearest Neighbours (LMNN). Additionally, exploring alternate embedding techniques such as BERT and setting up a DevOps lifecycle for a continuously learning models could greatly increase performance.

4. References

ArXiv Computer Science. (n.d.). Retrieved February 22, 2020, from https://arxiv.org/list/cs/pastweek?show=1600

Blei, D. M., Ng, A. Y., & Edu, J. B. (2003). *Latent Dirichlet Allocation Michael I. Jordan*. *Journal of Machine Learning Research* (Vol. 3).

Gensim: Text Preprocessing. (n.d.). Retrieved February 22, 2020, from https://radimrehurek.com/gensim/parsing/preprocessing.html

Jones, K. S. (1972). A STATISTICAL INTERPRETATION OF TERM SPECIFICITY AND ITS APPLICATION IN RETRIEVAL. *Journal of Documentation*, *28*(1), 11–21. https://doi.org/10.1108/eb026526

Le, Q., & Mikolov, T. (2014). *Distributed Representations of Sentences and Documents*.

Mikolov, T., Chen, K., Corrado, G., & Dean, J. (2013). *Distributed Representations of Words and Phrases and their Compositionality*.

Moody, C. E. (2016). Mixing Dirichlet Topic Models and Word Embeddings to Make lda2vec.

Park, L. A. F., & Stone, G. (2015). The effect of assessor coverage and assessor accuracy on rank aggregation precision. https://doi.org/10.1145/2838931.2838937

Triwijoyo, B. K., & Kartarina, K. (2019). View of Analysis of Document Clustering based on Cosine Similarity and K-Main Algorithms. *Journal of Information Systems and Informatics*, *1*(2), 164–177.

VectorMath.png (530×340). (n.d.). Retrieved January 30, 2020, from https://miro.medium.com/max/1060/0*aZyIDb5K4kOFBQV1.png

Zhang, D., Author, C., Zhao, S., Duan, Z., Chen, J., & Zhang, Y. (2020). A multi-label classification method using a hierarchical and transparent representation for paper-reviewer recommendation. *ACM Transactions on Information Systems*, *38*(1). https://doi.org/10.1145/3361719