Using NER and Doc2Vec to cluster South African criminal cases

by

Carel Kagiso Nchachi

Submitted in partial fulfillment of the requirements for the degree
Masters (Computer Science)
in the Faculty of Engineering, Built Environment and Information Technology
University of Pretoria, Pretoria

November 2021

Publication data:

Carel Kagiso Nchachi. Using NER and Doc2Vec to cluster South African criminal cases. Masters thesis, University of Pretoria, Department of Computer Science, Pretoria, South Africa, November 2021.

Using NER and Doc2Vec to cluster South African criminal cases

by

Carel Kagiso Nchachi
E-mail: u13140443@tuks.co.za

Abstract

The judicial system is the central pillar of law and order across the world. It is responsible for maintaining order amongst citizens and also solving litigations that arise. Although this system has worked quite well, there still exists several challenges, such as racial biases in cases, shortage of legal professionals and inconsistencies with regards to rulings in cases. These challenges need to be addressed in order to maintain law and order in society and to help strengthen the criminal justice system.

Researchers have incorporated Natural Language Processing (NLP) techniques to help address some of these challenges. Focusing primarily on three legal applications, which are Legal Judgment Prediction (LJP), Similar Case Matching (SCM) and Legal Question Answering (LQA)[28].

SCM focuses on identifying the relationships among cases using the available information. In other words, SCM is focused on segmenting or grouping legal cases. This is especially useful for Common Law judicial systems, where judicial decisions are based on similar and representative cases that have happened in the past. South Africa uses this type of judicial system.

Although good progress has been made in SCM applications, there currently exists several challenges found in the these models. These challenges include using entities found in a legal document to improve the matching of similar cases and the interpretability of

these models.

In this research we will focus on applying the SCM application on South African criminal cases, by creating a model that will be able to match similar crime cases together. This model will also solve the two challenges currently faced in SCM applications.

We found that using a Named Entity Recognizer (NER) with a Paragraph Vector-Distributed memory (PV-DM) model produced better results than using conventional PV-DM or TFIDF model. This model also overcomes the current SCM challenges as it uses the entities found in cases as the main variables for the model (using the NER model). Since the entities help explain how the model mapped similar case, this makes the model also interpretable.
Based on the accuracy (similarity score) of the model, we can use this model as tool to segment criminal cases in real life.
Keywords: Justice, NER, judicial, PV-DM, Similar Case Matching

Supervisor : Dr. V. Marivate
Department : Department of Computer Science
Degree : Master of Science

"We need to keep making our streets safer and our criminal justice system fairer - our homeland more secure, our world more peaceful and sustainable for the next generation."

Barack Obama (2016)

"Injustice anywhere is a threat to justice everywhere."

Martin Luther King, Jr (1963)

Acknowledgements

- I would like to thank Dr. Marivate for assisting me completing my research;
- I would like to thank my family and friends for providing the emotional support that I needed to complete this research.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

In chapter 12 of the National Development Plan (NDP), which is the South African government's strategic plan of eliminating poverty and reducing inequality by 2030, entails building safer communities[5]. One of the key points to achieve this goal is by strengthening the criminal justice system.

The NDP further explains that inspiring public confidence in the criminal justice system was necessary to prevent crime and increase levels of safety. The perceptions that criminals evade the law, arrests do not lead to convictions or the courts are unfair dent the public's confidence in the criminal justice system[5]. The judicial system is responsible for the conviction of crimes, thus playing an important role in the criminal justice system. Challenges such as biases in cases (either racial or gender), shortage of legal professionals and inconsistencies with regards to the ruling of cases remains prevalent in judicial systems across the world.

NLP can help solve some these challenges and also improve the judicial system in general. Researches have used NLP in LJP, SCM and LQA applications to address some of these challenges. Legal Judgment Prediction and Similar Case Matching applications are considered to be the main function for judgement in Civil and Common Law[28].

In Common Law judicial systems the judicial decisions are based on similar and rep-

resentative cases that have happened in the past. South Africa uses this type of system, thus meaning that utilizing the SCM application can improve the South African judicial system. This is achieved by matching similar cases to identify any inconsistencies in rulings and assist legal professionals in decision making, e.g. Judges can use all the similar cases found for their current case to help decide on a ruling and attorneys can use similar cases to prepare for their current cases.

Good progress has been made with regards to SCM applications. Leitner[17] manually annotated 54,000 entities from 6,700 legal cases, then mapped them to 19 semantic classes. Using a Conditional Random Fields (CRFs) and bidirectional Long-Short Term Memory Networks (BiLSTMs) model, they achieved an F1-score of 95.95%, preforming best on the following labels: judge, court and law [17]. Although this model shows great performance in entity recognition, this has not yet been used to cluster cases.

Howe[12], using LSA-based linSVM was able to classify 6,227 Singapore Supreme Court judgments, which had 51 legal area labels. The model produced a F1 score of 63.2%, with a precision and recall rate of 83.4% and 57.8% respectively[12].
This approach cannot be used for effective document clustering of legal cases as it does not consider the inner relatedness of cases. E.g. Two cases might be classified as fraud cases but are different based on other entities such as the sentencing and law used.

This research aims to create a model that will segment cases according to the details of each case. This will be achieved by using a Named Entity Recogniser (NER) with a Doc2vec model. This model will address the two challenges faced in Similar Case Matching and will also help group relevant cases together, which can then be used by law professional and researchers. This would help improve the South African justice system and help South Africa reach its NDP goals.

1.1 Objectives

- Conduct a literature survey of document segmentation and entity recognition techniques in the field of Natural Language Processing.
- Train and test a Named Entity Recognition (NER) model
- Train and test four different document segmentation models (one of them being a NER with a doc2vec model)
- Evaluate the results

1.2 Contributions

- Overcoming the current SCM challenges, by using NER with a doc2vec model for document segmentation, as this has not been done before.
- Building a documentation segmentation tool for the South African criminal justice system

1.3 Thesis Outline

The remainder of this thesis is structured as follows:

- **Chapter 2** covers the literature survey.
- **Chapter 3** focuses on the methodology and the model results.
- **Chapter 4** gives the final conclusion and the future work.
- **Appendix A** describes the crime categories found in the cases.
- **Appendix ??** covers full model results.

Chapter 2

Literature Survey

The chapter focuses on the literature survey of this dissertation.

2.1 Natural Language Processing for document similarity

Natural Language Processing is study that combines linguistics, computation and statistics. NLP falls part of artificial intelligence with the main objective of understanding and expressing human language[15]. Applications such as language translation, semantics, speech recognition, text summarization and information retrieval all fall within NLP[25]. It has been implemented and used across all industries, this includes email filtering, which entails determining whether an email is spam or ham, and smart assistants such as Amazon's Alexa and Apple's Siri.

Historically, language processing was done based on a set of rules that was coded into a software in order to examine a sentence. With the increase in data, this approach became redundant as it was nearly impossible to write rules that would be able to satisfy the amount of data available[20].
Modern approaches use data driven approaches to process language. NLP is the collective name for all these approaches to processing language.

2.1.1 Term Frequency Inverse Document Frequency (TDIF)

Various representation of words exists in NLP, with the most common ones being variants of Bag of Words (BOW) models[14]. Bag of words models describe the occurrence of words that appear at least once within a document. Each word represents a feature in the model.

Information regarding the structure, order of sentences or semantics of the words are discarded. The document is represented as a multidimensional vector, which contains the frequency associated with each word in the document. This representation of text is referred to as a vector space representation[1].

The number of distinct words in a data set determines the overall dimensionality of the vector space. The terms in a dictionary can be represented in some arbitrary order as t_1,t_2,...,t_N, where N is the total number of distinct words (or features). The ith document is represented as an ordered set of N values (X_{i1},X_{i2},..,X_{iN}),where the value X_{ij} is a weight that measures the importance of the jth term t_j in the ith document. This is referred to as an N-dimensional vector[2].

There are various methods used to calculate the weights. The most common method is counting the number of occurrences of each term in a specific document, this is known as term frequency. Another method is using binary representation where 0 and 1 indicates the absences and presences of the term in the specified document respectively[2].

A more complex method of calculating the weights is to combine the term frequency with a measure of the rarity of a term in the corpus (which is the entire set of documents). This method is called the Term Frequency Inverse Document Frequency (TFIDF)[2]. The formula of the TFIDF is written below:

$$TFIDF(t,d) = TF(t,d) \times IDF(t) \tag{2.1}$$

$$IDF(t) = \log \frac{1+n}{1+df(t)} + 1 \tag{2.2}$$

where:
t denotes the terms,
d denotes each document,
IDF(t) measures the importance of the word,
df(t) denotes number of documents in which the term t appears,
n is the total number of documents

An example of using this method is as follows:
A document that contains 200 words, with the word dog appearing 5 times in the document. The TF(t,d), which is the term frequency, of the word dog is then (5 / 200) = 0.025. We assume that we have a corpus of 10 million documents and the word dog appears in a thousand of these documents. Then, the IDF(t), which is the inverse document frequency, is calculated as log(10,000,000 / 1,000) = 4. Thus, to calculate the TFIDF we take the product of the term frequency and the inverse document frequency: 0.025 * 4 = 0.1.

TFIDF is found to be a simple, but efficient algorithm for document similarity[24]. Furthermore, its straightforward encoding makes it ideal for creating a basis for more complex algorithms and retrieval systems.
TFIDF has one major drawback though,it struggles to understand the semantics of the document[24]. It is not able to identify the meanings of words such as synonyms.

2.1.2 Named Entity Recognizer

Information extraction is one of the branches of NLP. Information extraction is used to help add meaning to raw data in order for it to be easier processed by the computer. Information Extraction is important for data mining, information retrieval, machine translation and summarization. Only computationally transparent, semantically classified and well defined extracted information is useful for information systems. Hence, recognizing entities and semantically meaningful relations between entities is a vital component to providing focused information access[22].

Named entity recognition (NER) is one of the sub-tasks of information extraction, which identifies entities such as proper nouns in a text and classifies them to the appropriate named entity classes[22]. Examples of these entity classes include person (name or surname), city, country and organization.

NER can be used for the following:

- Information Retrieval:
  This task identifies and retrieves relevant documents from a set of data based on a query input. Approximately 71% of the queries in search engines contain named entities[27].
  NER can be used to identify the named entities in the query, as well as extract the relevant document by using their classified named entities and their relatedness to the query.

- Question Answering:
  This task takes questions as input and returns a correct/relevant answer. Similar to the information retrieval task, NER is used to analyse the questions in order to recognise the named entities found within that question, which will be later used to identify the correct documents and construct the answer for that question[27].

- Machine Translation:
  This task autonomously translates a text from a certain language to another. NER is used to help decide on whether named entities should be either meaning-translated or phoneme-translated[27].

- Text Clustering:
  NER can be used for search result clustering, by creating the clusters using the ratio of entities contained in each cluster[27]. This approach helps in both creating and explaining the clusters.

There are two main approaches to developing NER systems, which are the Linguistic approach, which is based on handcrafted rules, and the statistical approach, which is

based on data driven approaches[22].
These two approaches are fully explained in table 2.1.

Features	Linguistic	Statistical
Resources Exhaustion	Uses well designed and tested language grammar, lexicons, tagset and test corpus[22].	Uses well annotated training corpus with a considerable amount of Named Entities[22].
Accuracy of the tagger	With considerable amount of time, expertise and efforts, a precision and recall of 95% and 99% respectively can be reached.	Well designed tagset and tagger can produce an accuracy of up to 95 – 97%.
Portability to other domains	Easy to adopt grammar with little requirement of correction or improvement in some particular domain[22].	Taggers accuracy are dependent upon the coverage of Named Entities in the training corpus for a particular domain[22].
Towards 100% output	Non linguistic methods can be used to resolve tagging remained by linguistic tagger[22].	Difficult to improve after 97% accuracy.

Table 2.1: NER development approaches

2.1.3 Word2Vec and Doc2Vec

Word representation learning is one of the research areas in Semantics, which is a branch in NLP. One of the methodologies of word representation is word embedding.
Word embedding entails learning low-dimensional vectors from text corpora, one way to achieve this is by exploiting neural networks. These models have achieved state-

of-the-art performances in several NLP tasks, especially when integrated into a neural network architecture. These models have been shown to provide good prior knowledge, due to their generalisation ability. These models were first popularised by the Word2vec model[3].
The word2vec model proposes two models, the Continuous Bag-of-Words Model (CBOW) and the Continuous Skip-gram Model (Skip-gram)[19].

Continuous Bag-of-Words Model (CBOW)

The architecture of this model is similar to the feedforward Neural Network Language Model. The non-linear hidden layer is removed from the model and the projection layer is shared for all words and not only for the projection layer. This would mean that all words vectors are averaged.
The history of the order of the words does not influence the projection and the model also uses words from the future[19]. Figure 2.1 shows the CBOW architecture.

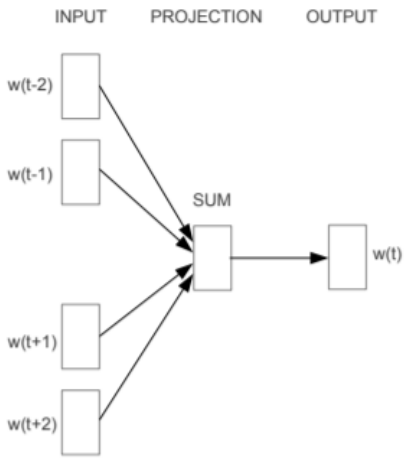

Figure 2.1: CBOW architecture[19]

Continuous Skip-gram Model (Skip-gram)

The Skip-gram model is similar to the CBOW, with the exception that this model tries to predict one word based on another word in the same sentence, rather than basing the

prediction on the context.
Each current word is used as an input to a log-linear classifier that has a continuous projection layer, which predicts words that are within a specific range before and after the current word [19]. Figure 2.2 shows the Skip-gram architecture.

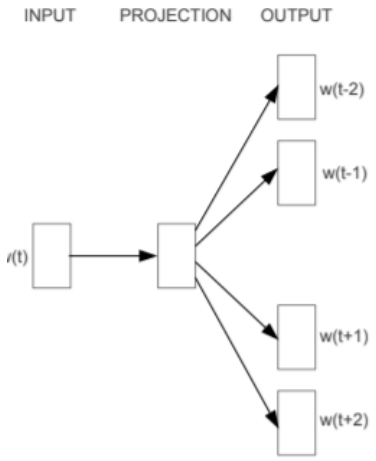

Figure 2.2: Skip-gram architecture[19]

The models described above are only able to work on a word level and cannot go beyond that. The paragraph to vector (doc2vec) solves this issue as it is able to construct representations of input sequences of variable length, this means that it is applicable to texts such as sentences, paragraphs and documents.
Paragraph to vector models are similar to word2vec models in that the paragraph vectors also contribute to the prediction task of the next word[16]. The paragraph to vector proposes 2 models, the distributed memory (PV-DM) model and the distributed bag of words (PV-DBOW) model.

Paragraph Vector: Distributed memory model (PV-DM)

The PV-DM maps every paragraph to a unique vector and every word to a unique vector as well. Both the paragraph and word vectors are either averaged/concatenated to predict the next word in the context.

The paragraph token could be considered as another word, since it acts as a memory that remembers what is currently missing from the context or paragraph.
The context is set to a fixed length and sampled from a sliding window over the paragraph[16].

Only across all the contexts generated from the same paragraph is the paragraph vectors shared. However, the word vector is shared across all paragraphs[16]. Stochastic gradient descent using back propagation is used to train both paragraph and word vectors.
Figure 2.3 shows the PV-DM architecture.

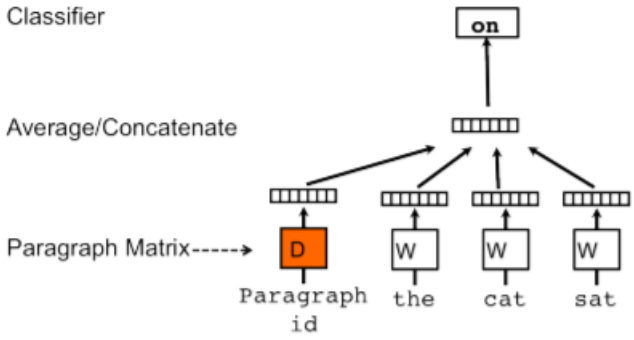

Figure 2.3: PV-DM architecture[16]

Paragraph Vector: Distributed bag of words (PV-DBOW))

PV-DBOW predicts words by ignoring the context of the words that are in the input. In other words, given each iteration of stochastic gradient descent, a text window is sampled, then after a random word is sampled from that text window and this would form a classification task that is given to the paragraph vector.
This model is not only conceptually simple,but it also requires less data storage. Only the softmax weights are stored (the PV-DM stores the softmax weights and the word vectors). This model is similar to the Skip-gram model[16].

Figure 2.4 shows the PV-DM architecture.

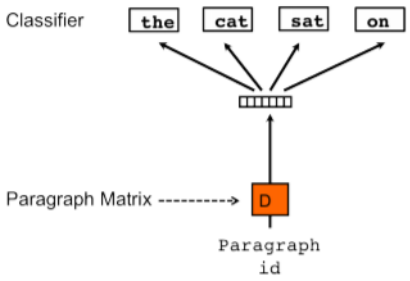

Figure 2.4: PV-DBOW architecture[16]

2.2 Evaluation measures

2.2.1 Classification measures

All classifier models, whether binary or multi-class, have the primary goal of predicting the outcome of a certain event. To determine whether a model produces good predictions, we need to compare the predicted values to the actual values, this is usually done in the test and training cycle of model development. The most common way is by using a confusion matrix.

		Predicted class	
Actual Class		Class = Yes	Class = No
	Class = Yes	True Positive	False Negative
	Class = No	False Positive	True Negative

Figure 2.5: Confusion matrix

True Positives (t_p)

Refers to values that are positive and were predicted correctly by the classifier. In figure 2.5, the t_p's would be when the predicted values and actual values are both yes.

True Negatives (t_n)

Refers to values that are negative and were predicted correctly by the classifier. In figure 2.5, the t_n's would be when the predicted values and actual values are both no.

False Positives (f_p)

This occurs when the predicted value is positive, but the actual value is negative.

False Negatives (f_n)

This occurs when the predicted value is negative, but the actual value is positive.

The following measures can be found from the confusion matrix and are used as performance measures to determine whether a model is good or not.

Accuracy

Accuracy is a ratio that measures the correctly predicted observations to the total observations.

$$Accuracy = \frac{t_p + t_n}{t_p + f_p + f_n + t_n} \tag{2.3}$$

Precision

Precision is a ratio that measures the correctly predicted positive observations to the total predicted positive observations. The precision measure is inversely proportional to the false positive rate.

$$precision = \frac{t_p}{t_p + f_p} \tag{2.4}$$

Recall

Recall is a ratio that measures the correctly predicted positive observations to all observations in actual class. This is also called the sensitivity.

$$recall = \frac{t_p}{t_p + f_n} \tag{2.5}$$

F1-score

F1 score is the harmonic mean of precision and recall[4]. Works better than accuracy for unbalanced class distributions.

$$F1 - score = \frac{2 * (precision * recall)}{precision + recall} \tag{2.6}$$

Matthews correlation coefficient (MCC)

Matthews correlation coefficient calculates the Pearson product-moment correlation coefficient between predicted and actual values[4].

$$MCC = \frac{(t_p * t_n) - (f_p * f_n)}{\sqrt{(t_p + f_p) * (t_p + f_n) * (t_n + f_p) * (t_n + f_n)}} \tag{2.7}$$

2.2.2 Document segmentation measures

In this section, we discuss two scoring text similarity measures, which are the Cosine and Jaccard similarity measures.

Cosine Similarity

Cosine Similarity takes the cosine of the angle of two vectors. The similarity is inversely proportional to the angle, that is the similarity is 1 when the angle is 0. Cosine Similarity is the most commonly used measure in computer linguistics [13].

$$sim(v, w) = cos(\theta) = \frac{vw}{||v||||w||} = \frac{\sum_{i=1}^{n} v_i w_i}{\sqrt{\sum_{i=1}^{n} v_i^2}\sqrt{\sum_{i=1}^{n} w_i^2}} \tag{2.8}$$

Jaccard similarity

Jaccard Similarity is defined as the intersection of two documents divided by the union of the two documents. This can also be referred to as the number of common words over the total number of words.

$$J(doc_1, doc_2) = \frac{doc_1 \bigcap doc_2}{doc_1 \cup doc_2} \tag{2.9}$$

2.3 NLP used in the Judical system

2.3.1 Legal Judgment Prediction

The decision of judgment results in Civil law systems are based on facts and statutory articles, that is, a person will only receive legal sanctions once the prohibited acts,that are prescribed by law, have been violated or breached[28].
Legal Judgement Prediction (LJP) focuses on the prediction of judgement results using both the facts extracted from the case and the statutory articles found in the Civil law system.

Examples of LJP systems

The United States (US) have several offender-assessment instruments, with the majority of Judaical system in the US utilizing one the following instruments [10]:

- Static Risk Assessment (SRA)
- Correctional Offender Management Profiling for Alternative Sanctions (COMPAS)
- Ohio Risk Assessment System (ORAS)
- Women's Risk Need Assessment (WRNA)
- Level of Service Inventory (LSI), LSI-R and LS/CMI

Although all of the models above differ in methodology, they were developed using regression-based techniques, with the goal of high accuracy prediction of judgement results.
Although these models show good performance,they can be still improved.

Challenges

In order to improve the performance of LJP models the following challenges need to addressed:

- For long legal text, global information needs to be obtained by using document reasoning and understanding techniques

- Handling of low-frequent labels is required for LJP
- Interpretability is essential for LJP [28]. Although existing embedding-based methods produce good results, they however work as a black box. This means, that the factors or importance of variables with regards to the predictions are unknown. Implementing these methods in real legal systems could lead to unfairness and ethical issues such as race biases. Using legal knowledge and symbols can help improve the interpretability of LJP models[28].

2.3.2 Similar Case Matching

In the Common law system, judicial decisions are based on similar and representative cases that have happened in the past [28].

Similar Case Matching (SCM) is essential to improving the prediction of judgement results in the Common law system[18]. It focuses on identifying the relationships among cases using the available information, this information can be of different levels of granularity, such as element and fact levels. SCM is considered to be a specific form of semantic matching and it can used for legal information retrieval[28].

Examples of SCM systems

Leitner[17] manually annotated 54,000 entities from 6,700 legal cases, then mapped them to 19 semantic classes, which are person, judge, lawyer, country, city, street, landscape, organization, company, institution, court, brand, law, ordinance, European legal norm, regulation , contract, court decision and legal literature. Using a Conditional Random Fields (CRFs) and bidirectional Long-Short Term Memory Networks (BiLSTMs) model for this data, they achieved a F1-score of 95.95%, preforming best on the following labels: judge, court and law.
Although this model shows great performance in entity recognition, this has not yet been used to cluster cases.

Howe[12], using LSA-based linSVM model was able to classify 6,227 Singapore Supreme

Court judgments, which had 51 legal area labels. The model produced a F1 score of 63.2%, with a precision and recall rate of 83.4% and 57.8% respectively.
This approach cannot be used for effective document clustering of legal cases as it does not consider the inner relatedness of cases. E.g. Two cases might be classified as fraud cases but are different based on other entities such as the sentencing and law used.

Challenges

Improving the performance of SCM models would require the following challenges to be addressed:

- Focusing more on elemental based representation. That is, using entities found in a legal document to match similar cases[28].
- Incorporating legal knowledge into models. Zhong[28] found that using semantic-level matching does not work well for SCM. He further adds that by adding legal knowledge into models this would improve the model performance and also make the model more interpretability.

2.3.3 Legal Question Answering

Legal Question Answering (LQA) focuses on answering queries in the legal domain. Legal professionals are also responsible for providing reliable and accurate legal consultation services to their clients or general public, but due to the lack of legal professionals the work force, clients or the general public getting enough and high-quality consultation services has often at times proved to be a challenge[28]. LQA focuses on addressing this challenge.
The questions that LQA deals with varies, this includes questions that focus on legal concept explanations and questions that focus on the analysis of specific cases.

Challenges

Although much progress has been done in LQA there's still exists a gap between exiting models and humans. To address this the following challenges have to resolved:

- Existing models struggle with legal multi-hop reasoning
- Understanding legal concepts. Current models struggle with knowledge understanding.

2.3.4 Implications of biased data

Just like all models, data plays an important role in building accurate legal models. Accurate and unbiased data is required to create reliable models, if this is not done correctly it could cause a detrimental effect to the justice system.
ProPublica found that COMPAS was biased towards black defendants by assigning them higher risk scores. The reason for this was that the training data had historical discrimination embedded in it [9].
There is no doubt that justice algorithms can help improve the criminal justice system, but this can only be done with unbiased data.

2.4 Summary

This chapter provides a literature survey for this research. The chapter provides a brief discussion on using NLP for document clustering. We then discuss the three legal applications where NLP is utilized.

Chapter 3

Methodology & Results

The chapter focuses on the methodology and results.

3.1 Methodology

3.1.1 Data Collection

The data is extracted from the Juda website and stored as a csv file. The Juda website contains 25 000 South African high court cases that were heard between 1994 and 2018. Only Common law offences still applicable within the South African legal system are considered for this research. The list of these Common law offences was extracted from the South African Police Service (SAPS) website[26].

3.1.2 NER model

To create the NER model we used the first 300 cases in the dataset the training set, while the last 50 cases in the dataset were selected as the test set. The training set is also split into a development and testing set, where 80% of the training data is used for the development set and the other 20% for testing.

The python Beautiful Soup library, which is a library used for pulling data out of HTML and XML files was used to remove all XML characters that occur in both the training

and test set. We then converted all characters into lower casing and replaced the brackets with a hyphen (-), as this helps when annotating the data. Lastly, we only keep all characters, numbers, colons and hyphens in the data, while the rest of the symbols are replaced with an empty space.

To annotate the data we used seven labels. We found that these seven labels can be used to extract the key features found each document. The table below shows the labels, some so the words annotated and how we validated these words.

Labels	Extracted words	Validation of annotations	Number of labels
Crime Type	Robbery, Assault, Rape, Murder, Fraud, Theft, etc (31 different category types)	Used the list of Common law offences extracted from the (SAPS) website validate these annotations[26]	712 labels
Sentence	conviction, set aside, guilty, not guilty, imprisonment, etc	Used the list of Common law terms was extracted from the paralegal advice website to validate these annotations[6]	780 labels
Law SA	Criminal Procedure Act 51 OF 1977, Criminal Law Amendment Act 105 of 1997,etc	Used the list of acts found on the South African government's website to validate these annotations [8]	216 labels
Section	s 11(1), s 3(2), section 2, etc	Went through each act to verify these sections	165 labels
Case Number	CC63/2006, CAR 1/97, c41, etc	We googled each of these case numbers to verify whether these case numbers point to the actual cases	274 labels

Court	Magistrate's Court, High Court, Constitutional Court and the Supreme Court of Appeal	Used the list of judicial courts found on the South African South African Judicial System website to validate these annotations[21].	62 labels
Judge	BHC Pickard JP, Ebrahim AJ, Dhlodhlo ADJP, etc	We googled each of these judges to verify whether they are South African judicial judges	219 labels

Table 3.1: Annotations

The python Spacy package was used to create and test the NER model. Spacy is a Python and Cython library for Natural Language Processing[11]. The library contains predefined models for language tagging, parsing and entity recognition [11]. In order to make the training model robust we added texts that do not have annotation and are completely not related to the law cases. We used common English phrases as the non-annotated text. This would help the model to distinguish whether a document is a criminal case or not.

Figures 3.2 and 3.2 shows examples of annotated cases with the labels assigned to key legal terms.

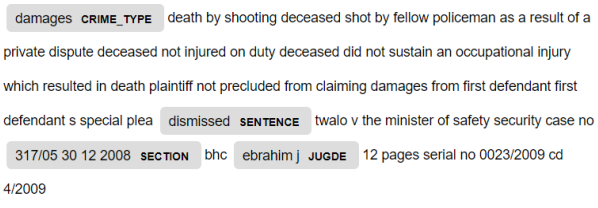

Figure 3.1: Example 1 of annotated case

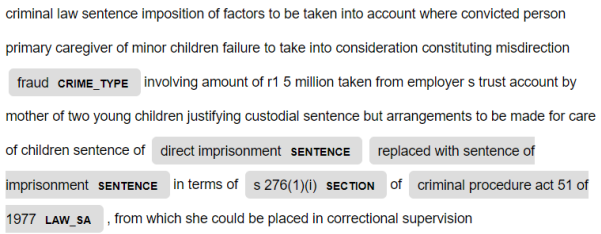

Figure 3.2: Example 2 of annotated case

Model Evaluation

We will evaluate the model by calculating the F1, precision and recall scores on the test data, for each label and the overall results[7].

3.1.3 Doc2Vec and TFIDF models

To create the Doc2Vec and TFIDF models entire data set. We selected 50 cases that were not used for training and testing the NER model as the testing data set. We used stratified sampling to extract these 50 cases. We applied larger weights (0.16) to five crime categories, which are Rape, Theft, Robbery, Assault and Murder. Reason for this is due to the fact that these five crime categories are the largest crime categories in South Africa[26].

The same cleaning methodology as the NER is applied, but with two additional steps. We used Spacy to lemmatize the data and replaced the Nan's found on the Case Number column with the string 'NA'.

Model creation for the TFIDF model

We used the sklearn library to create this model. We will use all the default parameter found in the sklearn package for the tfidfvectorizer[23]. We use a n-gram of 1 to 6 for this model.

Model creation for the PV-DM (Doc2vec) and PV-DM & PV-DBOW model

We used the Gensim Doc2vec library to create these models. We will also use Gensim's TaggedDocument library to tag the case number to the cleaned cases.

Model creation for the NER with a PV-DM model

We will run the NER model on the entire data set and extract all the entities from each case. We will then use Gensim's TaggedDocument library to tag the case number to the

extracted cases and run the data through the PV-DM model. The default parameters will be used for all models. Figure 3.3 shows methodology of creating the NER with a PV-DM model.

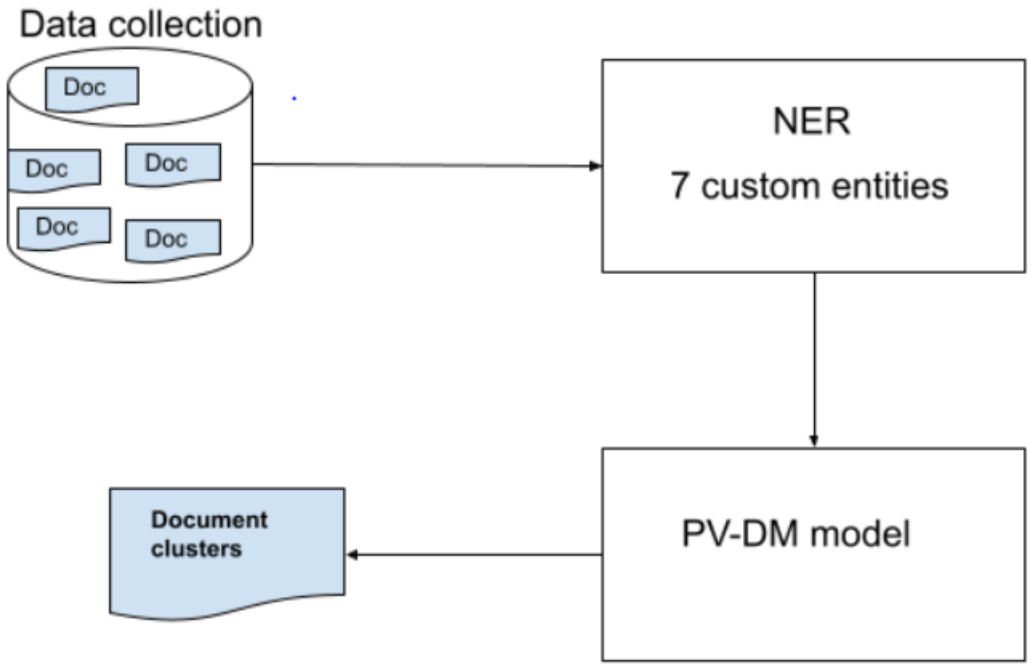

Figure 3.3: Methodology

Model Evaluation

We will evaluate all these models by running them on the test dataset and calculate the cosine similarity score for each model. We compared the best similarity scores produced by each model for each case.

3.2 Results

3.2.1 NER model results

In order to find the best parameters for the NER model, we ran the model using different parameters. We used the batch size and the drop out rate to optimize the model. We used the default parameters for all other parameters. Using the F1 score we found that the best batch size and drop out rate for this model was 8 and 0.5 respectively. Figures 3.4 and 3.5 shows the f1 scores for the overall model and for each label.

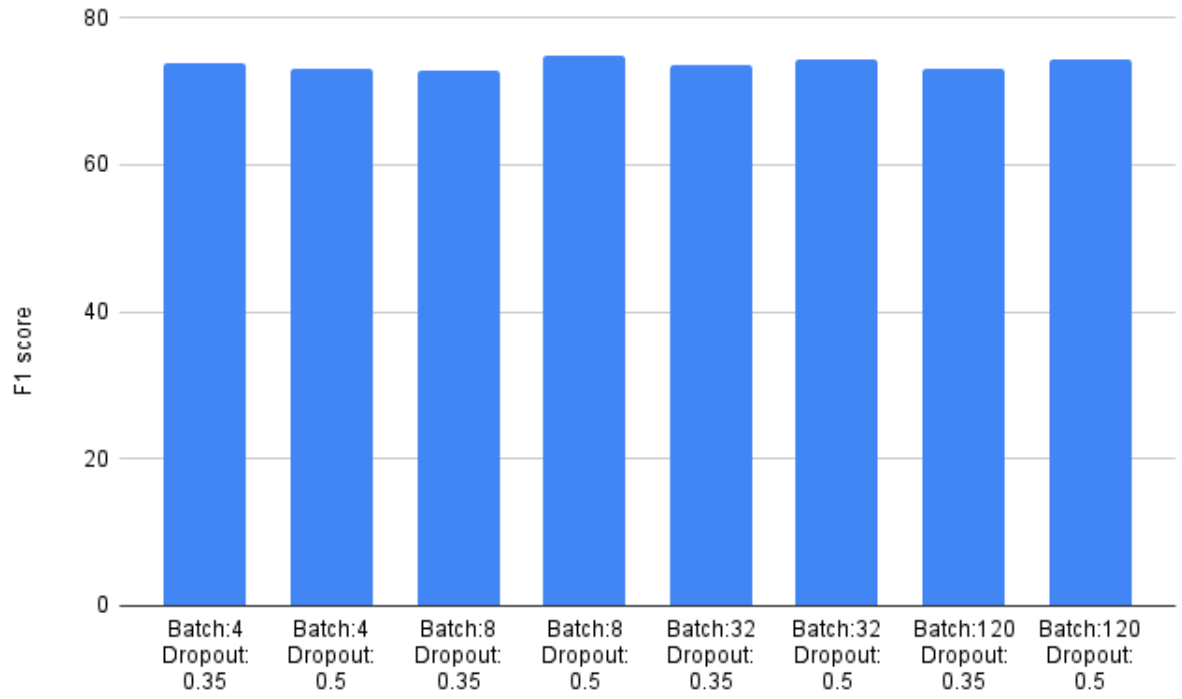

Figure 3.4: NER model parameters

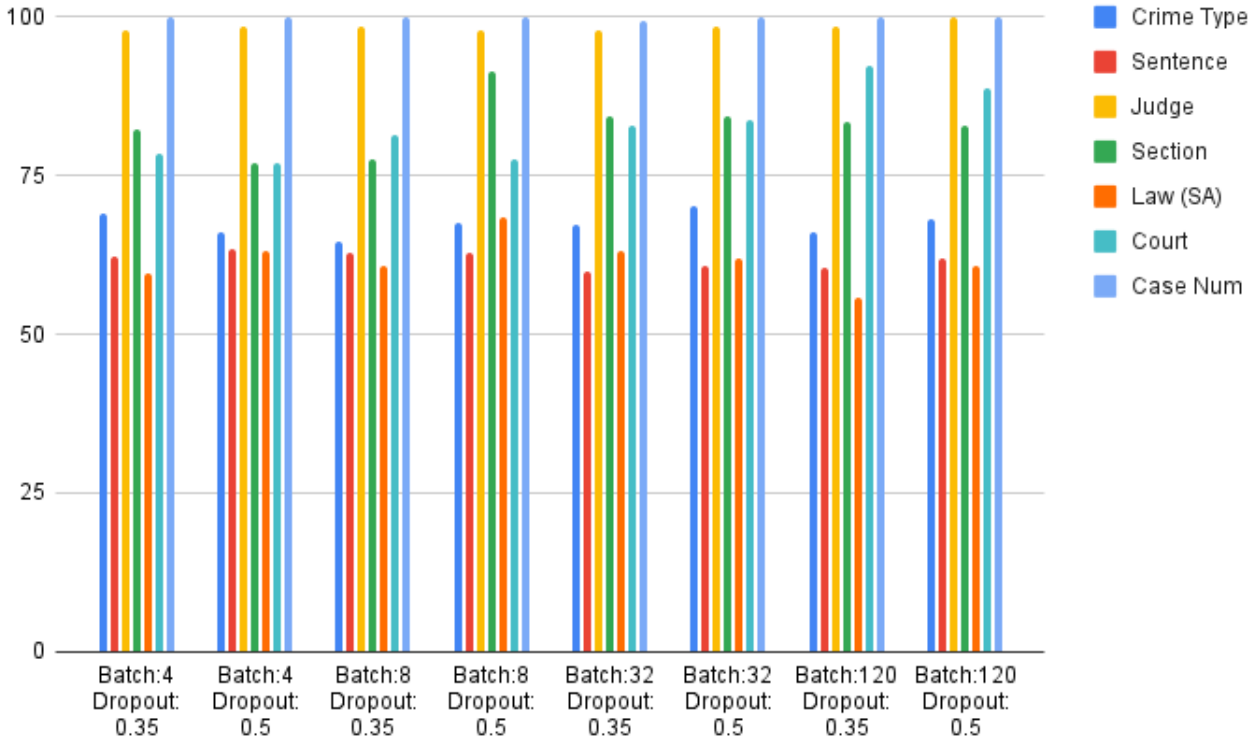

Figure 3.5: NER model parameters for each label

Using the best parameters found above the model produced an overall test F1 score of 74.84%, with an overall scores of 77.84% and 72% for the precision and recall receptively. The model achieved a 97.8% F1 score for the judges labels. The sentence label has the lowest F1 score of 62.91%. The full results of each label can be see on table 3.2.
The model achieved good results and was able to identify label entities in the test set. Adding more training data should improve the model quite significantly.

Table 3.2: NER model test results

Labels	Precision (%)	Recall (%)	F1 score (%)
Overall	77.84	72	74.84
Crime Type	71.66	63.86	67.55
Sentence	65.36	60.63	62.91
Judge	98.52	97.10	97.8
Section	99.85	88.1	91.36
Law (SA)	76.93	61.54	68.37
Court	70.59	85.71	77.421
Case Number	100	100	100

3.2.2 Doc2Vec and TFIDF model results

Figure 3.6 shows the similarity scores for all four models on the test data. The data is shown in descending order based on the similarity score, that is the highest scores are plotted first and the lowest scores are plotted last.

The NER with a PV-DM model produced the best results compared to the other three models. Majority of the test cases (31 cases) had a similarity score of 0.6 and above, with 0.95 being the best similarity score. In general, a 0.6 similarity score is considered a good score.
The PV-DM and PV-DM & PV-DBOW model performed badly achieving a similarity score of less than 0.53. The PV-DM model slightly outperformed the PV-DM & PV-DBOW model, but is clearly outperformed by the NER with a PV-DM model.
The Tfidf model performed the worse than all the other three models. Even with a n-gram of six the model still failed to identify similar cases. This shows that the Tfidf model does not perform well in identify similar cases for long documents.

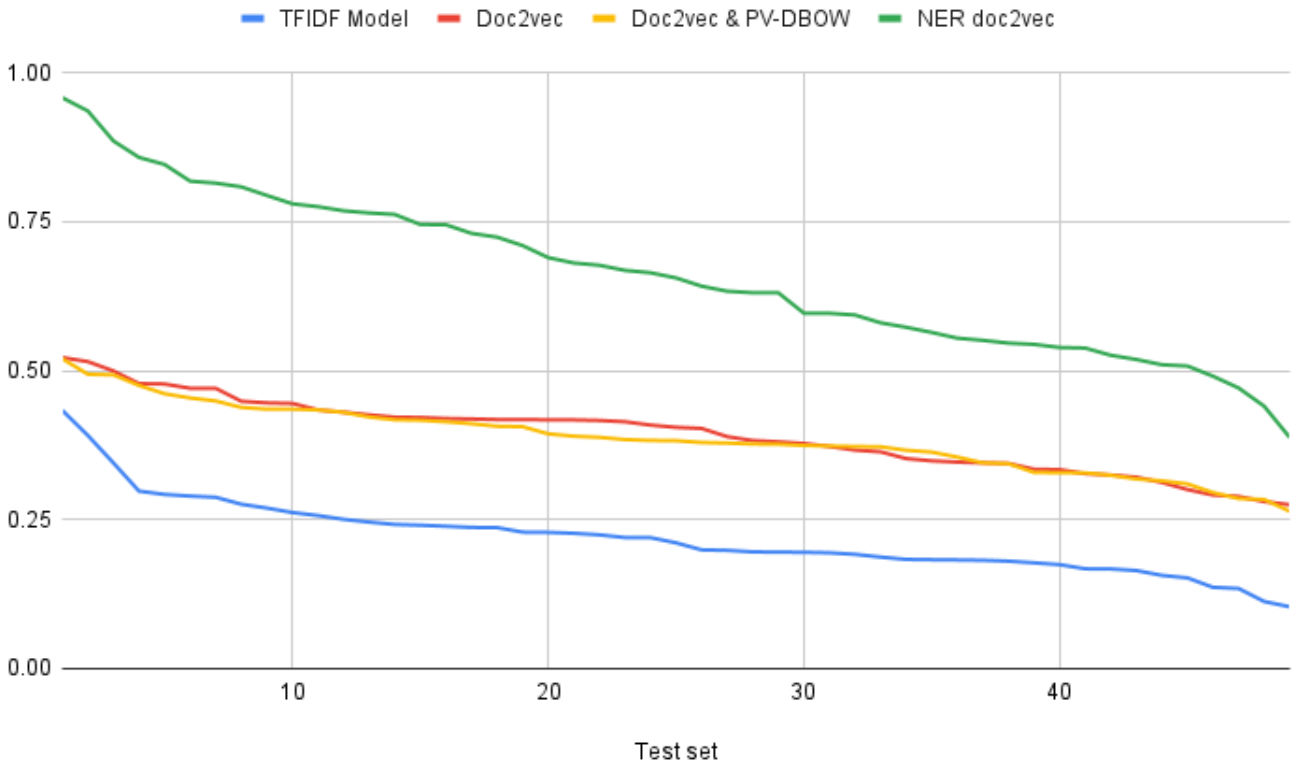

Figure 3.6: Model Similarity Scores

3.2.3 Insights of best model

In this section, we demonstrate how the NER with a PV-DM model can be used to extract insights of similar cases. We illustrate this by using the model's best predicted case (see table 3.3) and extract insights of this case and all the top 6 similar cases that the model predicted to be similar.

Figure 3.7 shows the top 20 similar cases predicted by the model. We will use a 0.700 threshold, this means that only cases that have a similarity score of 0.700 or greater are considered to be similar cases. The 0.7 threshold was derived after testing 20 cases and we found that cases that are considered to be similar have a similarity score of 0.700 or greater.

All top 20 cases that were predicted to be similar to this case have a similarity score of greater than 0.70 threshold and are considered to be similar.

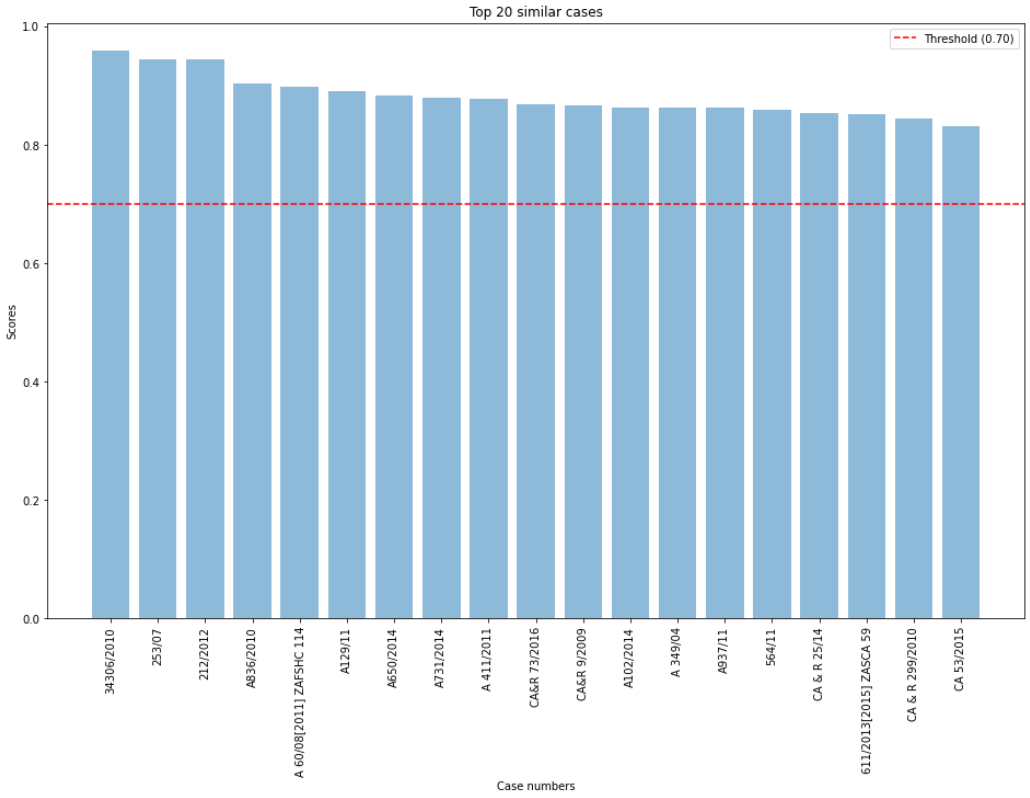

Figure 3.7: Top 20 similar cases

Tables 3.3 and 3.4 shows insights of the selected cases. This case is a fraud case, which was heard in the Gauteng province in 2012. The case took 7 days for a judgement to be made. The only label that was extracted with the NER model is the crime type label,which is the word fraud.

Case No	Case Summary	Judgement Year	Province	Days till judgment

13586/2011	"Fraud. Misappropriation of monies. Respondent fraudulently misappropriating monies from his employer. Claim for those monies."	2012	Gauteng	7

Table 3.3: Selected case example

Case No	Crime_Type	Sentence	Law	Section	Judge	Court
-	'fraud'	-	-	-	-	-

Table 3.4: NER labels extracted from selected case

Tables 3.5 and 3.6 shows insights of of the top six predicted similar cases. All these cases were heard between the years 2011 and 2016. Four of these cases were heard in the Eastern Cape, while the rest were heard in the Free State and Western Cape respectively. All theses cases had a similarity score of above 0.8, which suggests that the model predicted that the selected model is strongly associated with the predicted similar cases.

All these cases were identified to be fraud cases. Two of these fraud cases received a verdict of the case being dismissed, with one of the cases being dismissed due to a state error. These cases also had a longer day till judgement as compared to the other cases. Two other cases were charged for fraud, where one case received a reduced sentencing of two years ,while the other received a three year sentence.

The income tax act of 1962 and the value added tax act 89 of 1995 were used in one of the cases.

Case No	Case Summary	Judgement Year	Province	Days till judgment	Similarity score

CA&R 73/2016	"Criminal law. Fraud. Sentence. Thirty.six.year.old first offender, married with two minor children. Convicted of fraud involving R15 000. Money not repaid to complainant ́96 Overemphasis on deterrence by court a quo warranting interference with sentence. Sentence duly reduced from 3 years to 2 years."	2016	Eastern Cape	0	0.96
CA& R 25/14	"Criminal law. Sentence. Fraud. Community care funds diverted to enrich another. Accused not profiting herself. Husband having alcohol problem and not able to take care of the young children. Three years' correctional supervision."	2014	Eastern Cape	6	0.94

CA& R 299/2010	"Criminal law. Fraud. Elements of. Prejudice. Whether court a quo erred in finding that State failed to prove element of actual or potential prejudice arising from respondent having furnished appellant with a falsified matriculation certificate¬†. Legal position and requirements for establishing actual or potential prejudice discussed. State failed to prove that matric qualification required for placement in position applied and therefore potential or actual prejudice of appellant not established. State not adducing any evidence that respondents' dishonesty resulted in prejudice. Element of actual or potential prejudice not established, appeal dismissed."	2011	Eastern Cape	112	0.94
CA&R 9/2009	"Criminal law. Fraud. Tampering with blood samples to exclude paternity finding in DNA test. Chain of evidence examined. Appeal against conviction dismissed for first appellant, succeeding for second appellant."	2011	Eastern Cape	92	0.93

A 60/2011 ZAFSHC 114	"Fraud.Prejudice.What constitutes.Tender fraud.Appellants failing to disclose in tender application their connection to persons employed by state.Such constituting prejudice.Prejudice not only proprietary."	2012	Free State	0	0.90
A129/11	"Criminal law. Tax fraud. VAT. Failing to register and failing to submit returns. Sales of meat by business entity. Search and seizure. Whether documents seized falling within ambit of warrant. Whether warrant issued for one business entity also extending to related entity. Warrant issued under Income Tax Act 58 of 1962 while search aimed at contravention of VAT Act. Examination of invoices and other evidence. Value Added Tax Act 89 of 1991, s 58(c)."	2011	Western Cape	0	0.84

Table 3.5: Similar predicted cases

Case No	Crime_Type	Sentence	Law	Section	Judge	Court

-	'fraud'	'sentence', 'reduced from 3 years', 'to 2 years'	-	-	-	-
-	'fraud'	'sentence', 'three years'	-	-	-	-
-	'fraud'	'erred', 'dismissed'	-	-	-	-
-	'fraud'	'conviction', 'dismissed'	-	-	-	-
-	'fraud'	-	-	-	-	-
-	'fraud'	'income tax act 58 of 1962', 'value added tax act 89', 'of 1991'	-	-	-	-

Table 3.6: NER labels of similar predicted cases

3.3 Summary

We discuss the methodology that was used for this dissertation.
The NER model achieved an overall test F1 score of 74.84%, which is a good result considering that only 300 cases,which had 31 different crime type categories, were used to train the model.
The NER with a PV-DM model produced the best results compared to the four models.

It produced a much better similarity scores compared to the other models. We demonstrate how we can use the NER with a PV-DM model to extract insights of similar cases. We illustrated this by using the model's best predicted case and extracted insights of this case and the top six cases that the model predicted to be similar.We found that all these cases were identified to be fraud cases. Two of these fraud cases received a verdict of the case being dismissed, with one of the cases being dismissed due to a state error. These cases also had a longer day till judgement as compared to the other cases.

Chapter 4

Conclusions

Section 4.1 provides the final conclusion of this thesis and Section 4.2 provides the future work.

4.1 Summary of Conclusions

The NER entities used for this research were appropriate for clustering criminal cases. Using only the entities as an input for the PV-DM model, we saw an average increase of 65% in similarity score across our test set.
The NER model performed well in classifying the labels , however we believe that by training more data the model's accuracy would increase. The NER with a PV-DM model clearly out performed the other models, while the the Tfidf model performed the worst. This suggests that by using entities as inputs we are able to better cluster documents.

We demonstrated how we could use the NER with a PV-DM model to extract insights and cluster similar cases. Not only were we able to correctly cluster similar cases, we were able to exact insights of all the similar cases. These insights can be used by scholars and legal professionals to better analyse these cases.
The NER with a PV-DM model address the two main challenges found in SCM, it uses the legal entities extracted from documents as input. These entities are also used as insights to interpret the model.

4.2 Future Work

We believe by adding more training data should help improve the NER model. The model would be able to achieve a 95% overall F1 score. Thus, would also improving the NER with a PV-DM model similarity score as well.
We would also like to test this approach on other fields like medical or financial fields. This would help us to determine whether this approach can be applied across several fields.

Although the the model can be improved, we believe the model in its current state can be used by legal professionals to assist them in court cases, since judicial decisions in South Africa are made according to similar and representative cases in the past (Common Law) and by scholars to assist them in their research with regards to the South African criminal justice system.

Bibliography

[1] Charu C Aggarwal. *Data mining: the textbook*. Springer, 2015.

[2] Max Bramer. *Principles of data mining*, volume 180. Springer, 2007.

[3] Jose Camacho-Collados and Mohammad Taher Pilehvar. From word to sense embeddings: A survey on vector representations of meaning. *Journal of Artificial Intelligence Research*, 63:743–788, 2018.

[4] Davide Chicco and Giuseppe Jurman. The advantages of the matthews correlation coefficient (mcc) over f1 score and accuracy in binary classification evaluation. *BMC genomics*, 21(1):1–13, 2020.

[5] National Planning Commission et al. National development plan 2030 – our future – make it work, 2012.

[6] Black Sash Education and Traning unit. Legal dictionary, 2021.

[7] Vijay Garla, Vincent Lo Re III, Zachariah Dorey-Stein, Farah Kidwai, Matthew Scotch, Julie Womack, Amy Justice, and Cynthia Brandt. The yale ctakes extensions for document classification: architecture and application. *Journal of the American Medical Informatics Association*, 18(5):614–620, 2011.

[8] South African Government. Acts, 2021.

[9] Ben Green and Yiling Chen. Disparate interactions: An algorithm-in-the-loop analysis of fairness in risk assessments. In *Proceedings of the Conference on Fairness, Accountability, and Transparency*, pages 90–99, 2019.

[10] Zachary Hamilton, Melanie-Angela Neuilly, Stephen Lee, and Robert Barnoski. Isolating modeling effects in offender risk assessment. *Journal of Experimental Criminology*, 11(2):299–318, 2015.

[11] Matthew Honnibal, Ines Montani, Sofie Van Landeghem, and Adriane Boyd. spaCy: Industrial-strength Natural Language Processing in Python, 2020.

[12] Jerrold Soh Tsin Howe, Lim How Khang, and Ian Ernst Chai. Legal area classification: A comparative study of text classifiers on singapore supreme court judgments. *arXiv preprint arXiv:1904.06470*, 2019.

[13] Anna Huang et al. Similarity measures for text document clustering. In *Proceedings of the sixth new zealand computer science research student conference (NZCSRSC2008), Christchurch, New Zealand*, volume 4, pages 9–56, 2008.

[14] Laura Igual and Santi Seguí. Introduction to data science. In *Introduction to Data Science*, pages 1–4. Springer, 2017.

[15] Wahab Khan, Ali Daud, Jamal A Nasir, and Tehmina Amjad. A survey on the state-of-the-art machine learning models in the context of nlp. *Kuwait journal of Science*, 43(4), 2016.

[16] Quoc Le and Tomas Mikolov. Distributed representations of sentences and documents. In *International conference on machine learning*, pages 1188–1196. PMLR, 2014.

[17] Elena Leitner, Georg Rehm, and Julián Moreno-Schneider. A dataset of german legal documents for named entity recognition. *arXiv preprint arXiv:2003.13016*, 2020.

[18] Jiamin Li, Xingbo Liu, Xiushan Nie, Lele Ma, Peng Li, Kai Zhang, and Yilong Yin. Weighted-attribute triplet hashing for large-scale similar judicial case matching. *Computational Intelligence and Neuroscience*, 2021, 2021.

[19] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*, 2013.

[20] Prakash M Nadkarni, Lucila Ohno-Machado, and Wendy W Chapman. Natural language processing: an introduction. *Journal of the American Medical Informatics Association*, 18(5):544–551, 2011.

[21] Department of Justice and Constitutional Development. The south african judicial system, 2020.

[22] Nita Patil, Ajay S Patil, and BV Pawar. Survey of named entity recognition systems with respect to indian and foreign languages. *International Journal of Computer Applications*, 134(16), 2016.

[23] 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. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

[24] Juan Ramos et al. Using tf-idf to determine word relevance in document queries. In *Proceedings of the first instructional conference on machine learning*, volume 242, pages 29–48. Citeseer, 2003.

[25] Nihar Ranjan, Kaushal Mundada, Kunal Phaltane, and Saim Ahmad. A survey on techniques in nlp. *International Journal of Computer Applications*, 134(8):6–9, 2016.

[26] SAPS. Saps crimestats, 2020.

[27] Khaled Shaalan. A survey of arabic named entity recognition and classification. *Computational Linguistics*, 40(2):469–510, 2014.

[28] Haoxi Zhong, Chaojun Xiao, Cunchao Tu, Tianyang Zhang, Zhiyuan Liu, and Maosong Sun. How does nlp benefit legal system: A summary of legal artificial intelligence. *arXiv preprint arXiv:2004.12158*, 2020.

Appendix A

Crime Category descriptions

Common law offences still applicable within the South African legal system are defined below.

Abduction

Abduction consists in unlawfully taking a minor out of the control of his or her custodian with the intention of enabling someone to marry or have sexual intercourse with that minor.

Arson

Arson is the unlawful and intentional setting fire to an immovable property belonging to another.

Assault

Assault consists of unlawfully and intentionally applying force to the person of another; inspiring a belief in another person that force is immediately to be applied to him or her; Assault with intent to cause grievous bodily harm. This is another form of assault, however, committed with the intention to cause serious bodily injury.

Bestiality

Bestiality consist in unlawful intentional sexual intercourse between a human being and an animal.

Bigamy

It consists of unlawfully and intentionally entering into what purports to be a lawful marriage ceremony with one person while lawfully married to another.

Contempt of court

Contempt of court consists in unlawfully and intentionally - violating the dignity, repute or authority of a judicial body or a judicial officer in his/her judicial capacity; or publishing information or comment concerning a pending judicial proceeding which has the tendency to influence the outcome of the proceeding or to interfere with the administration of justice in that proceeding.

Crimen Injuria

Crimen injuria consist of unlawfully and intentionally impairing the dignity or privacy of another person.

Culpable Homicide

Culpable homicide is the unlawful negligent killing of another human being.

Defamation

Defamation consists of the unlawful and intentional publication of matter that impairs another person's reputation.

Defeating or obstructing the course of justice

The crime of defeating or obstructing the course of justice consists of unlawfully and intentionally engaging in conduct which defeats or obstructs the course or administration

of justice.

Exposing an infant

This crime consists of unlawful and intentional exposure and abandonment of an infant in such a place or in such circumstance that its death from exposure is likely to result.

Extortion

It consists of taking from another some patrimonial or non-patrimonial advantage by intentionally and unlawfully subjecting that person to pressure which induces him or her to submit to the taking.

Forgery and uttering

Forgery consists of unlawfully and intentionally making a false document to the actual or potential prejudice of another. Uttering consists of unlawfully and intentionally passing off a false document (forged) to the actual or potential prejudice of another.

Fraud

It is the unlawful and intentional making of a misrepresentation which causes actual prejudice or which is potentially prejudicial to another.

High treason

It consists of any conduct unlawfully committed by a person owing allegiance to a state with the intention of:

- overthrowing the government of the Republic;
- coercing the government by violence into any action or inaction;
- violating, threatening or endangering the existence, independence or security of the Republic;
- changing the constitutional structure of the Republic.

Housebreaking with intent to commit a crime

Housebreaking with intent to commit a crime consists of unlawfully and intentionally breaking into and entering a building or structure with the intention of committing some crime in it.

Incest

Incest is unlawful and intentional sexual intercourse between male and female persons who are prohibited from marrying each other because they are related within the prohibited degrees of consanguinity, affinity or adoptive relationship.

Indecent assault

Indecent assault consists of unlawfully and intentionally assaulting, touching or holding another in circumstances in which either the act itself or the intention with which it is committed is indecent.

Kidnapping

This crime consists of unlawfully and intentionally depriving a person of his or her freedom of movement and/or, if such person is a child, the custodians of their control over the child.

Malicious injury to property

It consists of unlawfully and intentionally damaging the property of another.

Murder

Murder is the unlawful and intentional killing of a human being.

Perjury

Perjury consists in the unlawful and intentional making of a false statement in the course of a judicial proceeding by a person who has taken the oath or made an affirmation before,

or who has been admonished by somebody competent to administer or accept the oath, affirmation or admonition.

Poisoning or administering poison or other noxious substance

This crime consists of unlawfully and intentionally administering poison or other noxious (harmful) substance to another.

Public indecency

This crime consists of unlawfully, intentionally and publicly engaging in conduct which tends to deprave the morals of others, or which outrages the public's sense of decency.

Public violence

It consists of the unlawful and intentional commission, together with a number of people, of an act/s which assume serious dimensions and which are intended forcibly to disturb public peace and tranquillity or to invade the rights of others.

Rape

Rape consists of intentional unlawful sexual intercourse with a woman without her consent.

Receiving stolen property

The crime of receiving consists of unlawfully receiving possession of stolen property knowing it to have been stolen.

Robbery

It consists of the theft of property by intentionally using violence or threats of violence to induce submission to the taking of it from another.

Sedition

It consists of unlawfully and intentionally taking part in a concourse of people violently or by threats of violence challenging, defying or resisting the authority of the State; or causing such a concourse.

Theft

It consists of the unlawful appropriation of moveable corporeal property belonging to another with intent to deprive the owner permanently of the property.

Violating a corpse

It consists of unlawfully and intentionally violating a corpse.

Violating a grave

It consists of unlawfully and intentionally damaging a human grave.

A.1 Summary

This section provides a description of the crime categories that were used to create the models.