LDSI S2021 Course Project Report

Name: Quazi Fahim Faisal Dhruba (03722194)
Gloss ID: ldsi_s21_13

Summary

The project addresses the task of automating the classification of sentence types within BVA cases. An analysis of sentence segmenters is done, and subsequently reaches the conclusion that Law-specific sentence segmenter such as LUIMA does markedly better than others for the same development effort. A word embedding model was trained and analyzed, which was seen to reflect the domain text. Several classifiers were than trained based on both TFIDF featurization and Word Embedding featurization, and their performance observed. It was concluded that a Radial Kernel SVM model based on Word Embedding Featurization performed the best with an accuracy of 0.84.

Dataset Splitting

Documents in the test set: '0934845.txt', '1623087.txt', '1114333.txt', '1317440.txt', '1333883.txt', '0824445.txt', '1007840.txt', '0829439.txt', '1635686.txt'.
Documents in the dev set: '1631703.txt', '0601461.txt', '1522066.txt', '0806464.txt', '1311391.txt', '1235794.txt', '0805869.txt', '1414169.txt', '1624269.txt'.
The rest of the documents formed the training set.

Sentence Segmentation

Standard Segmentation Analysis
We used the standard sentence segmenter provided by spacy [1] (version: 3.0.6) without any extensions or exceptions on the entire training set. The documents with the some of the lowest Precision and Recall were the following: '*0843259.txt*', '*1638605.txt*' and '*1222019.txt*'.

File Name	Precision	Recall	F1 Score
'*0843259.txt'*	0.152	0.296	0.201
'*1638605.txt*'	0.160	0.267	0.200
'*1222019.txt*'	0.191	0.439	0.267

Examination of over-segmented and under-segmented parts:

- Initial parts containing Citations, Date, Docket No., etc. are always segmented into several sentences. This is expected as the segmenter has difficulty ignoring empty spaces.
- Sentences terminating without a punctuation often not segmented, for example capitalized Headers or single sentences under headers.
- Empty space between parts of the document have been segmented as sentences.
- Periods in abbreviations, for example, “Washington, DC.” often triggers a sentence to be segmented.
- Correctly segmented sentences often not within the range specified, due to redundant space before or after sentence.
- Continuous citations within the document often segmented into several sentences.

- There is a general trend of over-segmentation in the beginning and end of the document due to citations and footers.

Improved Segmentation Analysis

Several rules and exceptions were added to the spacy pipeline. For example, special characters ("\n","\t","\r"," ") and common parts causing over-segmentation, such as case headers. This led to the following change in the error metrics for the worst 3 documents:

File Name	Precision	Recall	F1 Score
'*0843259.txt'*	0.195	0.333	0.247
'*1638605.txt*'	0.200	0.367	0.259
'*1222019.txt*'	0.225	0.439	0.298

Overall, there were the following changes to the error metrics: precision increased by 0.061 recall increased by 0.012 and the F1 score increased 0.051.

Law-specific sentence segmenter (LUIMA [2]**)**

LUIMA sentence segmenter works impressively better on the overall dataset. Achieving the following scores in the error metrics: precision = 0.701, recall: 0.853, F1 Score: 0.770. Additionally, the following scores were reached for the 3 worst performing documents:

File Name	Precision	Recall	F1 Score
'0843259*.txt'*	0.292	0.519	0.373
'*0942105.txt*'	0.377	0.604	0.464
'*0820506.txt*'	0.405	0.652	0.500

To be noted, all segmenters have file, '0843259*.txt',* among the ones they worst performed in. Upon closer examination, it's a rather short, remanded decision with numbered instructions, which pose difficulties for the segmenters.

Error Analysis:

- Over-segments in locations where Capitalized words appear, this is not prevalent in the segmentation by spacy.
- LUIMA impressively handles headers within the document compared to spacy.
- The line indicating space for signatures in the case footer is entirely missed and not segmented by LUIMA.
- There is very strange splitting around numbers, for example, "1.", "2.".
- Both spacy and LUIMA struggle with CaseHeaders and CaseFooters. LUIMA fails to segment the CaseHeader as a whole.

Decision

LUIMA sentence segmenter performs the best among the segmenters analyzed, thus the decision was taken to use it. It performs with much greater accuracy compared to the other segmenters observed. The parts where it under performs are the same parts where the other segmenters perform even worse.

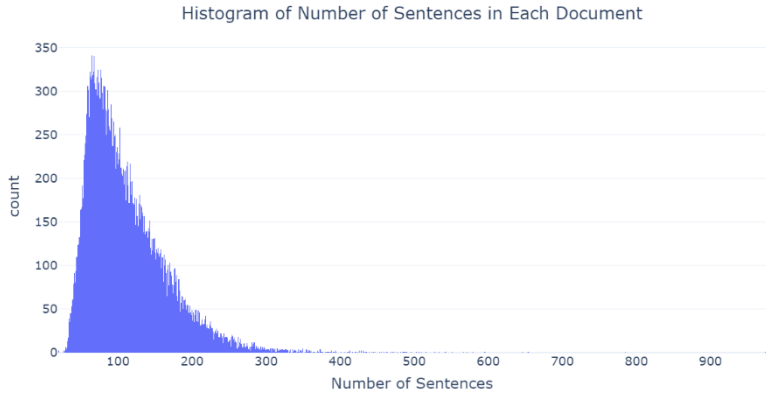
Preprocessing and Tokenization

The entire unlabeled corpus was segmented into sentences with the LUIMA segmenter which resulted in **3,360,513** individual sentences. The histogram below (bin width=1) illustrates the distribution of the number of sentences across all unlabeled decisions:

The sentences were then tokenized using Spacy's tokenizer, with the following extensions and exceptions:

- 'Vet. App.' and 'Fed. Cir.' Were kept as whole tokens. As these words often occur together.

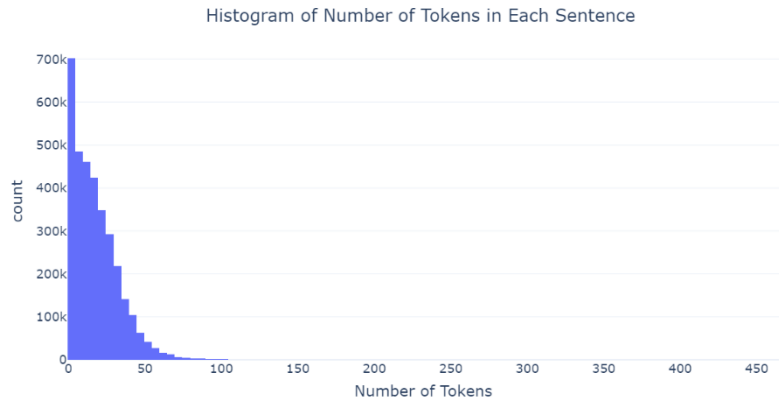
- Possessive nouns such as ‘veteran’s’ or ‘veterans’’ had the apostrophe removed and tokenized including the ‘s’. This was separately handled as possessive implications were being lost otherwise.
- Only Alphanumeric characters were preserved. Symbols and special characters provide little meaning.



- Numbers were simplified to be denoted as <NUM*number_of_digits*>
- Punctuations were not tokenized
- Everything was lowercased

The performance of the tokenizer was observed on the sentences highlighted in the Classifier Workshop in addition to hand picked sentences from the corpus, and the above extensions were iteratively added to achieve a satisfactorily consistent tokenizer behavior.

The tokenizer was then used to tokenize all segmented sentences of the unlabeled data. The histogram below (bin width=5) illustrates the distribution of the number of tokens in the sentences of the unlabeled data:



A single file, with each line of which consisting of a sentence's tokens, separated by a single whitespace was assembled, where the order of the sentences was randomized. Prior to generating this file sentences with 5 or less tokens were dropped. This reduced the total number of individual sentences from **3,360,513** to **2,558,769**.

Custom Embeddings

FastText was used to train an unsupervised word embedding model for 10 epochs with a minimum word occurrence count of 20. The total vocabulary size of the model was **12,257** words.

The nearest neighbors of 14 words, including the 8 given in the instructions were observed. Observations for a few notable words’ nearest neighbors are given below:

Word	Nearest neighbors									
Veteran	He	Appellant	Have	She	His	Additionally	Adamantly	Subsequently	Furthermore	That

- The words “He”, “She” and “His” all denote the veterans and their gender.
- “Appellant”, is the most relevant to the word “Veteran” as the two words denote the same individual. This neighbor seems to be most meaningful.

- “Additionally”, ”Subsequently”, “Furthermore” and “That”, are all recurring generic words, that most likely does not provide additional meaning.

Word	Nearest neighbors									
Service	Connection	Disease	Disability	Incur	Active	Military	Inservice	During	Injury	Disorder

- In general, the nearest neighbors of “Service” seems one of the most meaningful.
- All the nearest neighbors can be categorized to have a relation with the word “Service” as the following:
  - Chronological Relation: “Incur”, “Inservice” and “During”. E.g., “…he had psychiatric treatment **during service**…”
  - State while in Service: “Disease”, “Disability”, “Injury” and “Disorder”. E.g., “…**Service connection** may be established for a **disability** resulting from **disease** or **injury incurred**…”
  - Service Type: “Connection”, “Military” and “Active”. E.g., “…evidence to be etiologically related to his **active service**…”

Word	Nearest neighbors									
Letter	Correspondence	Letters	Send	Inform	Advise	Postdate	Unsigned	Notification	Mail	211a

- Nearest neighbors of “Letter” are also quite meaningful. Except the neighbor “211a”, this seems to be an anomaly, as even going through the segmented sentences yielded no results, where “211a” occurs.
- Upon further observation it was found to be an abbreviated number which was not simplified when tokenized.
- All the other words (except “211a”), impressively, relate directly to the word “Letter”

Word	Nearest neighbors									
Records	Nprc	Nara	Jsrcc	Curr	Archives	Usascrur	Usascurr	Rmc	Services	Jssrc

- At first glance the neighbors may seem nonsensical but noting that all abbreviations were simplified when we tokenized, by removing the periods, shows that all nearest neighbors denote organizations which keep or generates records.

Word	Nearest neighbors									
Caused	Cause	Confused	Unlikely	Definitely	Causes	Patients	Causative	Quite	Slow	Contributor

- Most of the neighbors of “Caused” seem linguistically related.
- Specialized relation to the corpus is not particularly demonstrated, except maybe “patients”, as most BVA cases have mention of patients of some kind.

Training & Optimizing Classifiers

All models specified in the instructions were trained and observed, first with TFIDF featurization and then with the word embedding featurization. Parameters that have not been explicitly stated below have been set to their default values.

TFIDF Featurization model performance on Dev Set

Model	**Parameters / Hyperparameters**	**F1-Score**	**Worst Performing Classes**	**Notes**
Linear Models				
Linear SVM	Random state= 0,	0.82	EvidenceBasedReasoning, LegalPolicy, PolicyBasedReasoning	
Logistic Regression	Random state= 0,	**0.83**	EvidenceBasedReasoning, LegalPolicy, PolicyBasedReasoning	**Best Model**
Non-Linear Models				

Model	Parameters / Hyperparameters	F1-Score	Worst Performing Classes	Notes
Non-Linear SVM	Kernel = “radial basis function”, Random state= 0	0.82	EvidenceBasedReasoning, LegalPolicy, PolicyBasedReasoning	
	Kernel = “polynomial function”, Random state= 0	0.76	EvidenceBasedReasoning, LegalPolicy, PolicyBasedReasoning	
	Kernel = “sigmoid function”, Random state= 0	0.60	EvidenceBased/Intermediate Finding, LegalPolicy, PolicyBasedReasoning	
Decision Tree	Max Depth = None, Random state = 0	0.67	EvidenceBasedReasoning, LegalPolicy, PolicyBasedReasoning	Exhibits overfitting behavior, since accuracy on the training set as 1.0
	Max Depth = 20, Random state = 0	0.71	EvidenceBasedReasoning, LegalPolicy, PolicyBasedReasoning	Around the optimal depth
	Max Depth = 13, Random state = 0	0.69	EvidenceBasedReasoning, LegalPolicy, PolicyBasedReasoning	Exhibits Underfitting Behavior
Random Forest	Max Depth = 30, Number of Trees = 80, Random state = 0	0.82	EvidenceBasedReasoning, LegalPolicy, PolicyBasedReasoning	Around the optimal Parameters
	Max Depth = 30, Number of Trees = 100, Random state = 0	0.81	EvidenceBasedReasoning, LegalPolicy, PolicyBasedReasoning	Exhibits Overfitting Behavior
	Max Depth = 20, Number of Trees = 50, Random state = 0	0.78	EvidenceBasedReasoning, LegalPolicy, PolicyBasedReasoning	Exhibits Underfitting Behavior

Word Embedding Featurization model performance on Dev Set

Model	Parameters / Hyperparameters	F1-Score	Worst Performing Classes	Notes
Linear Models				
Linear SVM	Random state= 0	0.84	EvidenceBasedReasoning, LegalPolicy, PolicyBasedReasoning	
Logistic Regression	Random state= 0	0.83	EvidenceBasedReasoning, LegalPolicy, PolicyBasedReasoning	
Non-Linear Models				
Non-Linear SVM	Kernel = “radial basis function”, Random state= 0	**0.84**	EvidenceBasedReasoning, EvidenceBased/Intermediate Finding, PolicyBasedReasoning	**Best Model**

Model	Parameters / Hyperparameters	F1-Score	Worst Performing Classes	Notes
	Kernel = “polynomial function”, Random state= 0	0.82	EvidenceBasedReasoning, LegalPolicy, PolicyBasedReasoning	
	Kernel = “sigmoid function”, Random state= 0	0.68	EvidenceBased/Intermediate Finding, LegalPolicy, PolicyBasedReasoning	
Decision Tree	Max Depth = None, Random state = 0	0.67	EvidenceBasedReasoning, LegalPolicy, PolicyBasedReasoning	Exhibits Overfitting Behavior, since accuracy on the training set as 1.0
	Max Depth = 20, Random state = 0	0.70	EvidenceBasedReasoning, LegalPolicy, PolicyBasedReasoning	Exhibits Underfitting Behavior
	Max Depth = 13, Random state = 0	0.71	EvidenceBasedReasoning, LegalPolicy, PolicyBasedReasoning	Around Optimal Parameters
Random Forest	Max Depth = 50, Number of Trees = 100, Random state = 0	0.81	EvidenceBasedReasoning, LegalPolicy, PolicyBasedReasoning	Exhibits Overfitting Behavior
	Max Depth = 18, Number of Trees = 80, Random state = 0	0.82	EvidenceBasedReasoning, LegalPolicy, PolicyBasedReasoning	Around Optimal Parameters
	Max Depth = 10, Number of Trees = 20, Random state = 0	0.79	EvidenceBasedReasoning, LegalPolicy, PolicyBasedReasoning	Exhibits Underfitting Behavior

Test Set Evaluation

Best TFIDF Featurization model: Logistic Regression

TRAIN:

	precision	recall	f1-score	support
CaseFooter	0.95	0.98	0.97	85
CaseHeader	0.98	0.96	0.97	83
CaseIssue	0.93	0.95	0.94	78
Citation	0.97	0.98	0.97	1057
ConclusionOfLaw	0.89	0.75	0.81	163
Evidence	0.80	0.95	0.87	1926
EvidenceBased/Intermediate Finding	0.75	0.66	0.70	673
EvidenceBasedReasoning	0.73	0.44	0.55	552
Header	0.98	0.99	0.98	726
LegalPolicy	0.70	0.16	0.26	98
LegalRule	0.81	0.88	0.84	844
PolicyBasedReasoning	0.00	0.00	0.00	21
Procedure	0.89	0.89	0.89	822
RemandInstructions	0.88	0.81	0.84	336
accuracy			0.86	7464
macro avg	0.80	0.74	0.76	7464
weighted avg	0.85	0.86	0.85	7464

Best Word Embedding Featurization model: Radial Kernel SVM

TRAIN:

	precision	recall	f1-score	support
CaseFooter	0.95	0.95	0.95	85
CaseHeader	0.97	0.94	0.96	83
CaseIssue	0.91	0.96	0.94	78
Citation	0.97	0.99	0.98	1057
ConclusionOfLaw	0.82	0.73	0.77	163
Evidence	0.76	0.93	0.84	1926
EvidenceBased/Intermediate Finding	0.58	0.55	0.56	673
EvidenceBasedReasoning	0.51	0.23	0.31	552
Header	0.99	0.99	0.99	726
LegalPolicy	0.75	0.18	0.30	98
LegalRule	0.78	0.86	0.82	844
PolicyBasedReasoning	0.00	0.00	0.00	21
Procedure	0.90	0.87	0.88	822
RemandInstructions	0.87	0.71	0.78	336
accuracy			0.82	7464
macro avg	0.77	0.71	0.72	7464
weighted avg	0.81	0.82	0.80	7464

DEV:

	precision	recall	f1-score	support
CaseFooter	1.00	0.91	0.95	11
CaseHeader	1.00	1.00	1.00	9
CaseIssue	1.00	1.00	1.00	9
Citation	0.97	0.96	0.97	81
ConclusionOfLaw	0.73	0.55	0.63	20
Evidence	0.83	0.96	0.89	370
EvidenceBased/Intermediate Finding	0.47	0.24	0.32	67
EvidenceBasedReasoning	0.15	0.12	0.13	42
Header	0.96	1.00	0.98	75
LegalPolicy	0.00	0.00	0.00	4
LegalRule	0.83	0.75	0.79	73
PolicyBasedReasoning	0.00	0.00	0.00	3
Procedure	0.84	0.91	0.87	95
RemandInstructions	0.92	0.80	0.85	44
accuracy			0.83	903
macro avg	0.69	0.66	0.67	903
weighted avg	0.80	0.83	0.81	903

DEV:

	precision	recall	f1-score	support
CaseFooter	1.00	0.91	0.95	11
CaseHeader	1.00	1.00	1.00	9
CaseIssue	1.00	1.00	1.00	9
Citation	1.00	0.98	0.99	81
ConclusionOfLaw	0.81	0.65	0.72	20
Evidence	0.83	0.96	0.89	370
EvidenceBased/Intermediate Finding	0.38	0.27	0.32	67
EvidenceBasedReasoning	0.22	0.12	0.15	42
Header	1.00	1.00	1.00	75
LegalPolicy	1.00	0.25	0.40	4
LegalRule	0.83	0.81	0.82	73
PolicyBasedReasoning	0.00	0.00	0.00	3
Procedure	0.90	0.91	0.90	95
RemandInstructions	0.89	0.77	0.83	44
accuracy			0.84	903
macro avg	0.78	0.69	0.71	903
weighted avg	0.81	0.84	0.82	903

TEST:

	precision	recall	f1-score	support
CaseFooter	1.00	1.00	1.00	10
CaseHeader	1.00	1.00	1.00	9
CaseIssue	0.67	1.00	0.80	8
Citation	0.94	1.00	0.97	117
ConclusionOfLaw	0.88	0.71	0.79	21
Evidence	0.75	0.91	0.82	229
EvidenceBased/Intermediate Finding	0.49	0.53	0.51	58
EvidenceBasedReasoning	0.42	0.19	0.26	69
Header	0.95	1.00	0.97	76
LegalPolicy	1.00	0.11	0.20	9
LegalRule	0.78	0.75	0.76	111
PolicyBasedReasoning	0.00	0.00	0.00	4
Procedure	0.87	0.83	0.85	108
RemandInstructions	0.95	0.80	0.86	44
accuracy			0.80	873
macro avg	0.76	0.70	0.70	873
weighted avg	0.78	0.80	0.78	873

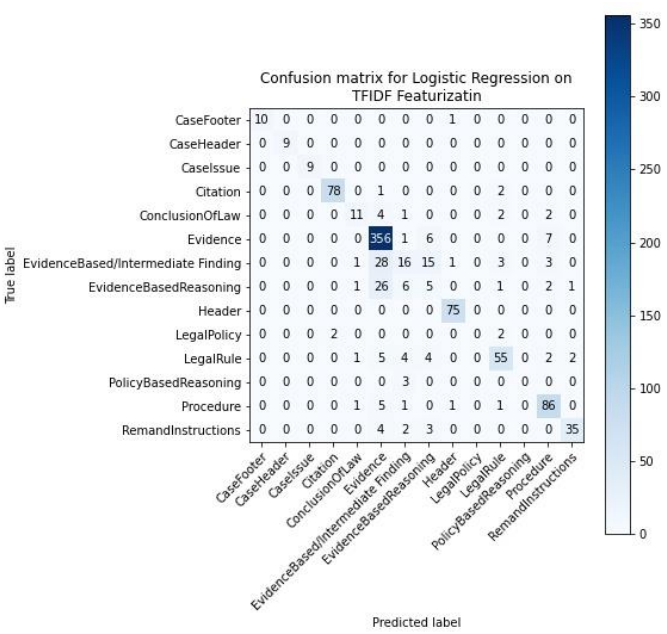
TEST:

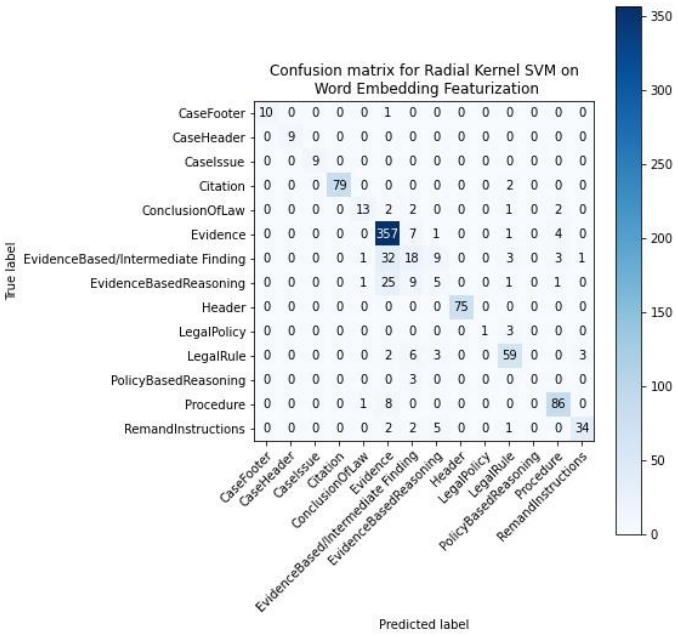
	precision	recall	f1-score	support
CaseFooter	1.00	1.00	1.00	10
CaseHeader	1.00	1.00	1.00	9
CaseIssue	0.73	1.00	0.84	8
Citation	0.93	0.97	0.95	117
ConclusionOfLaw	0.79	0.71	0.75	21
Evidence	0.75	0.93	0.83	229
EvidenceBased/Intermediate Finding	0.45	0.52	0.48	58
EvidenceBasedReasoning	0.62	0.23	0.34	69
Header	0.97	0.97	0.97	76
LegalPolicy	0.50	0.11	0.18	9
LegalRule	0.80	0.75	0.77	111
PolicyBasedReasoning	0.00	0.00	0.00	4
Procedure	0.91	0.85	0.88	108
RemandInstructions	0.85	0.77	0.81	44
accuracy			0.80	873
macro avg	0.74	0.70	0.70	873
weighted avg	0.79	0.80	0.79	873

- The overall F1-Score was focused on to choose the best model. When comparing performance based on individual classes the class specific F1-Score was focused on.
- Where obvious overfitting was observed (by comparing difference in training and dev accuracy), considerable effort was spent to find optimal parameters for each non-linear model.
- When comparing the best models for the two kinds of featurization, the model based on Word Embedding Featurization does comparably better on all classes except “Remand Instructions”, leading to a slightly better performing model overall (accuracy 0.83 vs 0.84 on dev set)
- The model based on Word Embedding Featurization did markedly well compared to the one based on TFIDF Featurization on the classes, ConclusionOfLaw and LegalPolicy.

Error Analysis

Confusion Matrices for the Dev Set





Best Model Overall: Radial Kernel SVM on Word Embedding Featurization

Error analysis of the three most difficult types

- EvidenceBasedReasoning
  - Some sentences do appear to be a wrong annotation, for example, “Although the Veteran is competent to report in-service acoustic trauma, he is not competent to render a probative (persuasive) opinion on a medical matter.” has been labeled an Evidence where it probably is not.
  - There are some sentences which are true misclassifications, such as, “In summary, the evidence shows that the veteran's claimed in-service stressors have been corroborated by service records, and he also has a medical diagnosis of post-traumatic stress disorder based on those stressors.” which is clearly a Finding. Here the importance of the key word “shows” might not have been captured by the model.
- EvidenceBased/Intermediate Finding
  - Often the sentences seem truly ambiguous between Reasonings and Findings, thus it is difficult to conclude what information may be missing that leads to a misclassification.
  - Here too, the importance of key words seems to be missing, such as the word “satisfied” in the following sentence, “The Board is satisfied VA has made reasonable efforts to obtain relevant records and evidence.” which clearly denotes a form of reasoning rather than a finding.
- PolicyBasedReasoning
  - Support for this class was the lowest, therefore the low performance can be attributed to a severe lack of data.
  - Must be noted that no sentence had a predicted label of “PolicyBasedReasoning”, supporting the notion that there is a severe lack of data. The analysis below was, therefore, done on False Negatives.
  - The generalized nature of a PolicyBasedReasoning statement could not be derived, also, possibly, due to lack of data.

Error analysis of some other types

- LegalRule
  - There seems to be a marked amount of misclassification of LegalPolicy as LegalRule, most probably due to significant similarity in structure and phrasing.
  - Some Citations, which have specific phrases from the cited source quoted verbatim, have often been classified as LegalRule, for example, “see also Dingess v. Nicholson, 19 Vet. App. 473, 484, 486 (2006) (holding that the VCAA notice requirements contained in 38 U.S.C.A. 5103(a) and 38 C.F.R. 3.159(b) *apply to all five elements of a service connection claim, which include: (1) veteran status; (2) existence of a disability; (3) a connection between the veteran's service and the disability; (4) degree of disability; and (5) effective date of the disability).*”
- LegalPolicy
  - Possible wrong annotation, “The Board has given consideration to the Veterans Claims Assistance Act of 2000 (VCAA).” given a true label of Evidence.
  - A lack of data also exists for the label LegalPolicy.
- RemandInstructions
  - LegalRule are often classified as RemandInstructions. This may be due to similar structure and phrasing. While the annotators are aware of whether a case was Remanded and thus can better annotate RemandInstructions by expecting them in the document, the model can only derive labels on the single sentence, and not on the entire document the sentence is taken from.
  - There are also some wrong annotations, such as, “If any action required by a remand is not undertaken, or is taken in a deficient manner, appropriate corrective action should be undertaken.”, which has been annotated as an EvidenceBasedReasoning.

Discussion

In order to achieve the objective of classifying sentences several elements are required. Firstly, a sentence segmenter needs to be developed, since we want to classify sentences, a whole document must be split into discrete sentences. To achieve this, domain specific segmenters such as the LUIMA segmenter performs the best according to the segmenters analyzed. But even these segmenters may not follow annotated documents fully, thus some development effort needs to be spent to improve these segmenters. An unsupervised word embedding model was also developed to capture the inter relation between the words specific to the corpus. Before the model could be trained the split sentences were tokenized. Spacy’s tokenization function was utilized, but here too considerable development effort may be spent to tailor it to the corpus. Upon observing the nearest neighbors of several words, it was concluded that the word embedding model sufficiently captures the essence of the corpus. Models based on Word Embedding Featurization performs slightly better. Performance between different kind of models is not markedly different. The project concludes that a Radial Kernel SVM model on Word Embedding Featurization performs that best.

Several pointers can be kept in mind when continuing with the project. The method proposed by Hannes Westermann et al [3]. as illustrated in the literature survey can be utilized for a more efficient annotation process. A small corpus should also be manually annotated on a sentence level, where annotators annotate single sentences as required by the classification models. This will help verify how much knowledge of the entire document is required to

correctly identify sentence types. Significant development effort should also be spent in developing a satisfactory sentence segmenter. Training a word embedding model on a larger corpus would also be beneficial. Deep learning approaches can also be explored, when developing classifiers, which was not done in this project.

Lessons Learned

Overall, the course project provided quite a robust opportunity to learn and explore the application of NLP in the legal domain. Certain parts, such as tokenization and sentence segmentation could be better explored if more time was provided. The code workshops during the semester were extremely useful, and an extended version might benefit future students of the course.

Code Instructions

The zip file contains 8 notebook files. Notebooks numbered 1 through 7 were used to code for the project. The *analyze* function can be accessed in the notebook named "**main**". Thus, all the cells in this notebook need to be executed sequentially, and the *analyze* function can be called at the end. No other notebooks need to be accessed to use the *analyze* function, they are only given so that the coding effort can be observed if required. There is also **no** *setup* function as simply executing the series of cells in the "main" notebook is required. The zip file also contains a "requirements.txt" file which lists the python packages required for the execution of the "main" notebook. It does not include other packages that may be required to fully execute through the numbered notebooks.

References

[1] "spaCy," [Online]. Available: https://spacy.io/.

[2] J. V. R. W. M. G. a. K. D. A. Savelka, "Sentence boundary detection in adjudicatory decisions in the united states.," 2017.

[3] J. S. Y. L. S. B. M. B. L. C. R. K. Hannes Westermann, "Sentence Embeddings and High-Speed Similarity Search for Fast Computer Assisted Annotation of Legal Documents," 2020.