Henry Hirsch
Natural Language Processing

Final Project:
Automated Classification of Federal Rules Using BERT

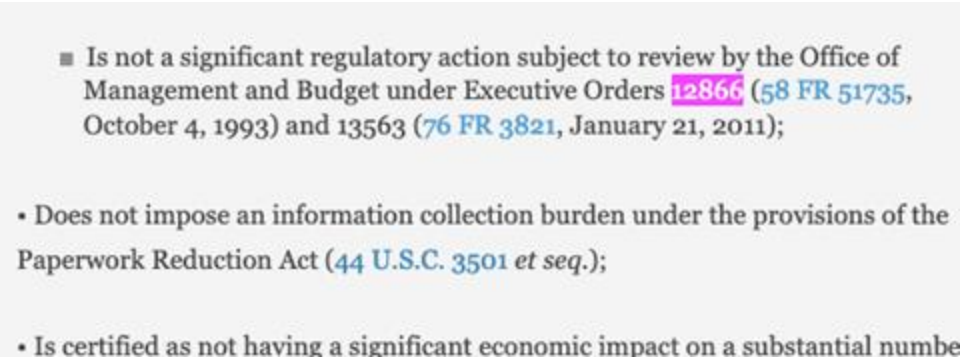
Introduction

I work at GW's Regulatory Studies Center. One of the RSC's main projects is tracking federal rulemaking activity and displaying the corresponding data visualizations on its RegStats page. Much of this data is gathered from the Federal Register and recorded in RSC's fr_tracking dataset. This dataset contains approximately 15,000 rows and has the following columns:

publication_date, effective_on, department, agency, independent_reg_agency, title, abstract, action, citation, document_number, regulation_id_number, docket_number, significant, econ_significant, 3(f)(1) significant, Major, html_url, Notes, and 5 U.S.C. 804(2).

Every week, a research assistant uses the web app to update the fr_tracking dataset with the federal rulemaking activity from the past week. Every column in the dataset is automatically populated with the exception of significant, econ_significant, 3(f)(1) significant, and Major. These are rule significance designations that cannot currently be updated by the web app because they require the research assistant to scan the text of the regulations in the Federal Register and look for certain textual patterns that indicate the corresponding level of significance. See the following two examples.

This is not a significant regulatory action under Executive Order 12866:

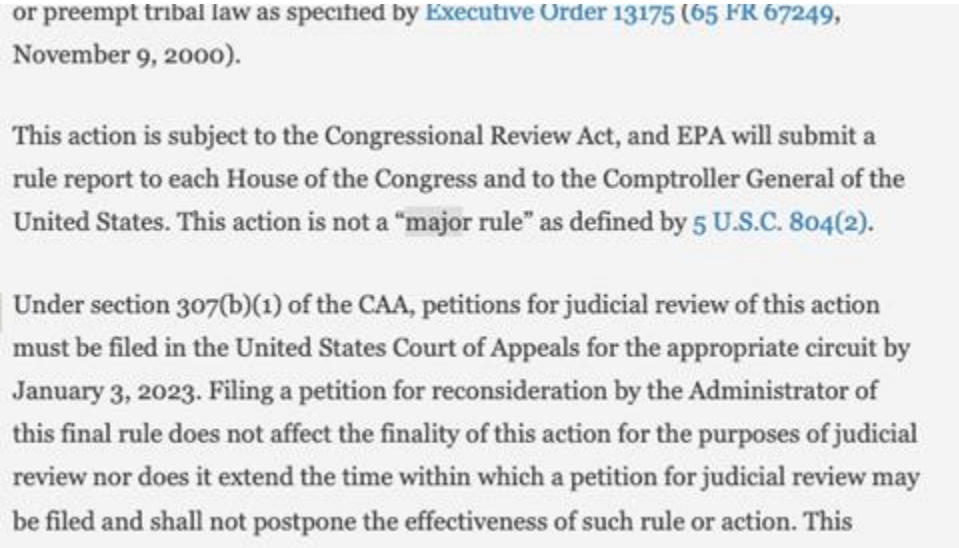


■ Is not a significant regulatory action subject to review by the Office of Management and Budget under Executive Orders 12866 (58 FR 51735, October 4, 1993) and 13563 (76 FR 3821, January 21, 2011);

• Does not impose an information collection burden under the provisions of the Paperwork Reduction Act (44 U.S.C. 3501 *et seq.*);

• Is certified as not having a significant economic impact on a substantial numbe

This is not a major rule under 5 U.S.C. 804(2):



or preempt tribal law as specified by Executive Order 13175 (65 FR 67249, November 9, 2000).

This action is subject to the Congressional Review Act, and EPA will submit a rule report to each House of the Congress and to the Comptroller General of the United States. This action is not a "major rule" as defined by 5 U.S.C. 804(2).

Under section 307(b)(1) of the CAA, petitions for judicial review of this action must be filed in the United States Court of Appeals for the appropriate circuit by January 3, 2023. Filing a petition for reconsideration by the Administrator of this final rule does not affect the finality of this action for the purposes of judicial review nor does it extend the time within which a petition for judicial review may be filed and shall not postpone the effectiveness of such rule or action. This

If, during the scan of the text, the research assistant determines that the rule meets the criteria for significant, econ_significant, 3(f)(1) significant, or Major, he or she enters a "1" in the corresponding column in the fr_tracking dataset. If it's explicitly stated that the rule does not meet these criteria, then the research assistant enters a "0". If neither of these scenarios is true, the research assistant enters a ".".

Checking these significance designations each week is a time-intensive task. And despite there being similarities in the way that these significance designations are worded in the text of different rules, there is not a single absolute pattern that they follow. However, there are some key commonalities (e.g. significant rules mentioning Executive Order 12866 and major rules mentioning 5 U.S.C. 804(2)).

In this project, I used the Federal Register API to download the corresponding text of all of the rules currently in the fr_tracking dataset and then used the significance designations already entered into the fr_tracking dataset as targets for model training (e.g. "1" or "0"). The model was trained on the corresponding text of each rule.

My Work

Architecture:

- Base model: DistilBERT-base-uncased (66M parameters, 6 layers)
- Three independent binary classifiers (not one multi-label model)
- Each classifier trained on category-specific extracted text
- Binary classification head: 768-dim → Dropout(0.1) → Linear(768→2) → Softmax

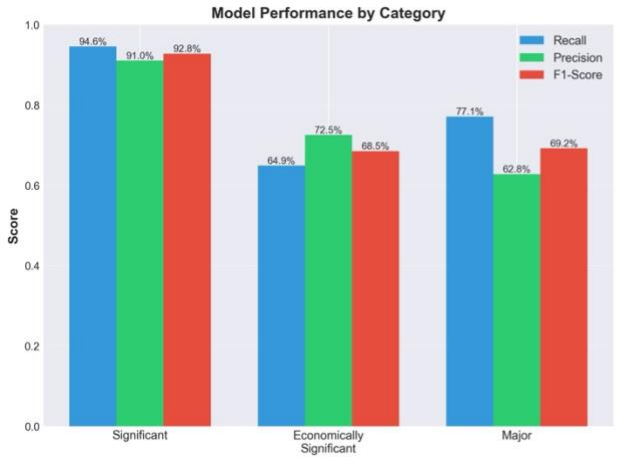
Training

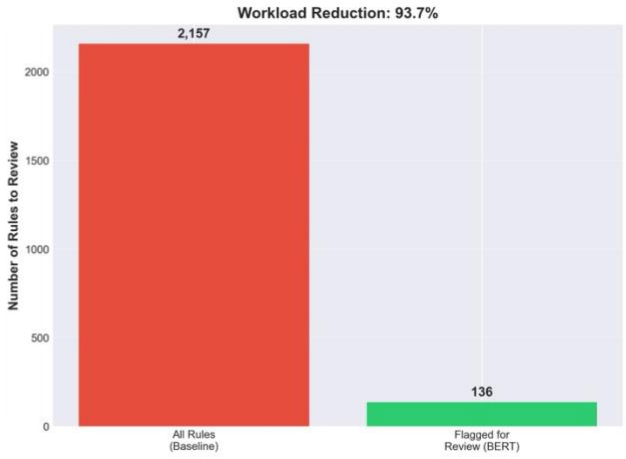
Configuration:

- Split: 70% train (10,066), 15% validation (2,157), 15% test (2,157)
- Stratified sampling to maintain class distributions
- Batch size: 8 (training), 16 (evaluation)
- Learning rate: 2e-5 with warmup (500 steps)
- Optimizer: AdamW (weight decay 0.01) Epochs: 3 with early stopping (patience=2)

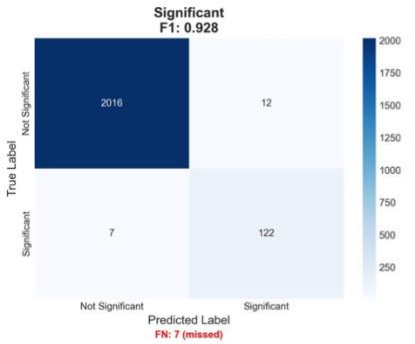
Results

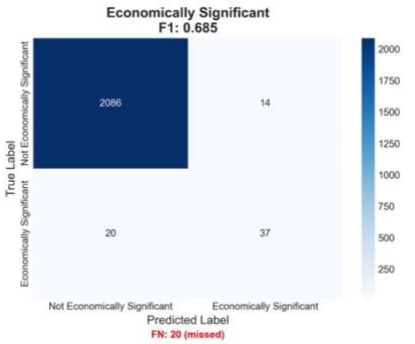
Category	Recall	Precision	F1-Score
Significant	94.6%	91.0%	0.928
Economically Significant	64.9%	72.5%	0.685
Major	77.1%	62.8%	0.692

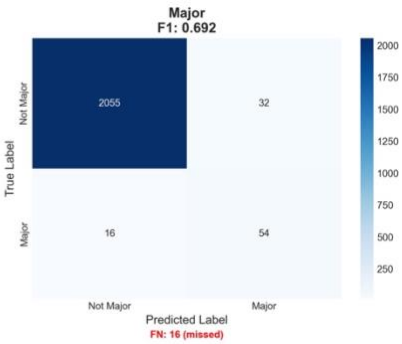




Significant Confusion Matrix	Predicted Negative	Predicted Positive
Actual Negative	2016 (TN)	12 (FP)
Actual Positive	7 (FN)	122 (TP)







I was fairly happy with how my model performed. The metrics for the significant class were very promising. The 94.6% recall rate was reassuring considering the high cost of false negatives (no human will ever review them since they aren't flagged in the first place). Still the fact that there were 7 false negatives in the test set is somewhat discouraging. Ideally, I would have 100% recall. Considering that I intended for a human to review all of the flagged rules, I'm less worried about misclassification (the precision metric) than failing to classify rules that should have been classified all together. The results for the economically significant and the major category were somewhat worse, but still trending in a promising direction. Still, for this tool to actually provide my workplace with value, the recall would have to be 100%.

Summary and Conclusion

I probably need to tweak the architecture of the model in order to get better results. Major is a distinct class and so a separate BERT model makes sense to use on it (it has its own distinct vocabulary). However, the economically significant subclass seems to have too much overlap with the significant class to warrant its own model. Perhaps a single hierarchical classification model would work better for the different types of significant rules. Taking even more drastic steps to account for the severely imbalanced data would also help performance. Finally, I noticed that some of the rules are misclassified in the early years of the fr_tracking dataset. This almost certainly mislead my model during training (always a risk of unverified manually labelled datasets). If I had the time, I'd go back and check all of the manually input classifications in the fr_tracking CSV.

In retrospect, these classifications could perhaps have been hardcoded with some very complex regex work. This is possibly the only way to achieve 100% recall. A statistical model is almost certainly going to always have some false negatives (due to having to optimize along certain metrics simultaneously). I may try this in a subsequent attempt.

Code

All of the code was written by me (with AI assistance).

References

fr_tracking.csv dataset
Federal Register