

To Aggregate or Not to Aggregate. That is the Question: A Case Study on Annotation Subjectivity in Span Prediction

Kemal Kurniawan¹ Meladel Mistica¹ Timothy Baldwin^{1,2} Jey Han Lau¹

¹The University of Melbourne ²MBZUAI
{kurniawan.k,misticam,laujh}@unimelb.edu.au
tb@ldwin.net

Abstract

This paper explores the task of automatic prediction of text spans in a legal problem description that support a legal area label. We use a corpus of problem descriptions written by laypeople in English that is annotated by practising lawyers. Inherent subjectivity exists in our task because legal area categorisation is a complex task, and lawyers often have different views on a problem, especially in the face of legally-imprecise descriptions of issues. Experiments show that training on majority-voted spans outperforms training on disaggregated ones.¹

1 Introduction

Automatic categorisation of lay descriptions of problems into relevant legal areas is of critical importance for providers of free legal assistance (Mistica et al., 2021). In our case, we have access to a dataset where a legal problem description is annotated by multiple lawyers who first perform document-level annotation by choosing relevant legal areas,² and then, for each legal area chosen, the lawyers highlight text spans that support their decision. These spans not only help justify the selected areas of law but also improve the interpretability of their decision.

The categorisation of legal areas is a complex problem, and lawyers sometimes have different views on a problem, especially when the task is performed on legally-imprecise descriptions of the personal circumstances of an individual. Therefore, subjectivity is inherent in our task. This subjectivity leads to annotation disagreements, both at the document- and the span-level. While such disagreements are often seen as noise that needs

¹Code is available at <https://github.com/kmkurn/wassa2024>.

²There are 32 possible legal areas including NEIGHBOURHOOD DISPUTES, ELDER LAW, and HOUSING AND RESIDENTIAL TENANCIES.

Area of law	Annotated description
ELDER LAW	I was admitted in a Public Hospital. I want the right to go home, NOT aged care!
GUARDIANSHIP AND ADMINISTRATION	I was admitted in a Public Hospital. I want the right to go home, NOT aged care!

Table 1: Examples of a description annotated with spans for two different areas of law.

to be eliminated in data annotation (Plank, 2022), here they are signal because they are produced by subject-matter experts.

In this paper, we explore the task of automatic span prediction using our expert-annotated dataset, as illustrated in Table 1. Given a problem description (which is a short document) and an area of law, the task aims to predict text spans in the description that support the area of law label. We describe the development of machine learning models for the task that are trained on a corpus containing legal problem descriptions written by laypeople in English. Across various evaluation scenarios, we find that aggregating training span annotations outperforms keeping them disaggregated.

2 Problem Statement

Given a text expressed as a sequence of N words $\mathbf{x} = x_1x_2 \dots x_N$ and a label l , the goal is to predict a set of non-overlapping spans $S = \{(b_i, e_i)\}_{i=1}^M$ where $1 \leq b_i \leq e_i \leq N$ such that the text segments $\{x_{b_i}x_{b_i+1} \dots x_{e_i}\}_{i=1}^M$ explain the reason for assigning l to \mathbf{x} . In other words, b_i and e_i respectively denote the beginning and the end indices of the i -th span supporting the assignment of l to \mathbf{x} . We cast the problem as sequence tagging by modelling the probability of S given \mathbf{x} and l as

$$P(S | \mathbf{x}, l) \propto \exp f(\mathbf{x}, \mathbf{y}, l) \quad (1)$$

where $\mathbf{y} = y_1 y_2 \dots y_N$ is a sequence of N tags representing the spans in S , each y_i corresponds to x_i , and f is a real-valued function that measures the relevance of \mathbf{y} in supporting the assignment of l to \mathbf{x} . To get \mathbf{y} from S , we use an encoding where y_i takes one of 5 possibilities depending on the position of i in a span (Sekine et al., 1998):

1. singleton, if $\exists(b, e) \in S$ where $b = e = i$;
2. beginning, if $\exists(b, e) \in S$ where $b = i < e$;
3. end, if $\exists(b, e) \in S$ where $b < i = e$;
4. inside, if $\exists(b, e) \in S$ where $b < i < e$; and
5. outside, otherwise.

The span prediction problem is then equivalent to finding the highest scoring sequence

$$\mathbf{y}^* = \arg \max_{\mathbf{y}} f(\mathbf{x}, \mathbf{y}, l).$$

The sequence \mathbf{y}^* is then decoded to get the final predicted spans.

3 Corpus

The corpus was collected by Justice Connect,³ an Australian public benevolent institution⁴ that connects laypeople seeking legal assistance with pro bono lawyers. On its website, Justice Connect allows help-seekers to describe their problem in free text format in English. After anonymising identifiable information, problem descriptions collected from July 2020 to early December 2023 were presented to a pool of lawyers to be annotated. Each annotator selected one or more out of the 32 areas of law that applied to the problem (thus it is a *multi-label* classification problem), representing the different law specialisations the case relates to. On average, a problem description is labelled with 3 areas of law. For each document-level area of law selected, the annotator then select spans of words⁵ that support their decision. On average, each problem description is annotated by 5 lawyers. This whole annotation process was carried out by Justice Connect. In other words, we do not perform any additional annotation and simply use the annotated corpus.

Relating to the problem statement in Section 2, the description and the area of law form the inputs \mathbf{x} and l respectively, while the spans make up the output S . Together, the input and the output form

³<https://justiceconnect.org.au>

⁴As defined by the Australian government: <https://www.acnc.gov.au/charity/charities/4a24f21a-38af-e811-a95e-000d3ad24c60/profile>

⁵The number of words must be at least three.

a labelled example of the task. Following prior work on a similar corpus (Mistica et al., 2021), we employ 20-fold cross validation to create the training and the test sets and randomly take 10% of the training set to form the development set. Over the 20 folds we have a total of 35K unique problem description and legal area pairs, with a total of 3.8M words in the problem descriptions.

4 Method

4.1 Subjectivity-Aware Evaluation

Because of the inherent subjectivity of the labelling task, a test input (consisting of a problem description and an area of law) can have multiple valid span annotations whose boundaries may not match exactly. Specifically for a given problem description, the same area of law can be supported by different spans. Similarly, the same span can support different areas of law. To deal with this mismatched boundaries issue, we adopt both span- and word-level evaluation. To address the issue of multiple valid spans, we experiment with 2 types of gold spans: majority-voted and best-matched. With these strategies, we have a total of 4 combinations of evaluation setup.

4.1.1 Span- and Word-Level Evaluation

In span-level evaluation, a predicted span is considered correct if it starts from and ends at the same positions as a gold span. In other words, their span boundaries must match exactly to be considered equal.

In contrast, word-level evaluation considers a word in a predicted span as correct if it is also a word in a gold span. Put simply, this evaluation gives a positive score to two overlapping spans whose boundaries do not match exactly.

We use precision, recall, and F₁ scores as evaluation metrics. We use the evaluation script⁶ of CoNLL-2000 chunking shared task (Tjong Kim Sang and Buchholz, 2000) to perform both types of evaluation.⁷

4.1.2 Majority-Voted and Best-Matched Gold Spans

We perform strict majority voting to get the majority-voted gold spans for evaluation. For example, if there are 2 annotators with the following span annotations:

⁶Downloadable from <https://www.cnts.ua.ac.be/conll2000/chunking/output.html>.

⁷Word-level evaluation is achieved by passing `-r` as option.

1. *[I was fired from work] because of [my complaint against my boss] months ago,*
2. *I was [fired from work] because of my [complaint against my boss months ago]*

where square brackets denote a span, then the gold spans are *fired from work* and *complaint against my boss*. In other words, only words voted by more than 50% of the annotators are included. In particular, words voted by exactly 50% of the annotators are *not* included.

Another type of gold spans we experiment with is the best-matched spans. Given an input and its predicted spans, best-matched spans of that input are its span annotations against which the predicted spans result in the highest F_1 score when evaluated. These span annotations must come from a single annotator. For instance, if (a) there are 2 annotators with the same span annotations as before, (b) the predicted span is only *fired from work*, and (c) span-level F_1 is used, then the best-matched spans are the spans given by the second annotator. A similar approach has been used in automatic text summarisation (Lin, 2004).

4.2 Model

We parameterise the function f in Equation (1) with a neural sequence tagger. The tagger uses a pretrained language model to provide contextual word representations and a bidirectional LSTM (Hochreiter and Schmidhuber, 1997) with a CRF output layer (Collobert et al., 2011) as the classifier similar to previous work (Lample et al., 2016). We use the implementation provided by the open-source NLP library FLAIR (Akbik et al., 2019).⁸

Following prior work on a similar corpus (Mistica et al., 2021), we use the base and uncased version of BERT (Devlin et al., 2019) as the pretrained language model. The problem description and the area of law are joined and given as a single text input to BERT. For example, if the problem description is *My landlord kicked me out without reason* and the area of law is HOUSING AND RESIDENTIAL TENANCIES then the input is *My landlord kicked me out without reason* <sep> HOUSING AND RESIDENTIAL TENANCIES where <sep> marks the end of the problem description. Both <sep> and succeeding input words corresponding to the area of law are excluded from evaluation.

⁸Version 0.13.

4.3 Training

We experiment with two approaches to dealing with subjectivity in model training. The first approach (MV) aggregates span annotations with majority voting similar to how the majority-voted gold spans are constructed (Section 4.1). This approach resolves subjectivity by only including spans on which the majority of annotators agree.

The second approach is repeated labelling (REL) which treats multiple annotations of the same input as separate labelled examples (Sheng et al., 2008). In other words, annotations in the training set are left as they are without any attempt to aggregate them. This approach embraces subjectivity by treating all annotations equally.

While REL may seem counterintuitive because the same input can be presented with different annotations, these annotations may have consistent patterns. Spans that are often (resp. rarely) annotated give a strong signal of the presence (resp. absence) of a true span. We expect that models can learn the correct spans from these signals.

For both approaches, the tagger is trained for 10 epochs to maximise the probability of the sequence of tags in the training set. Both learning rate and batch size are tuned on the development set. The word-level F_1 score against majority-voted spans is used as the hyperparameter tuning objective.

4.4 Comparisons

Baseline We employ a model that predicts spans randomly as a baseline (RANDOM) which reflects a model that does not perform any learning from data. The model tags each word in the input description with one out of 3 possibilities uniformly at random: start of a span, continuation of a span, or outside of any span. This sequence of tags is then decoded into a set of spans as the output.

Expert performance The majority-voted gold spans in Section 4.1 may not resemble spans produced by a real annotator. Therefore, even an expert annotator may not achieve perfect performance when evaluated against the majority-voted gold spans. We compute this expert performance to serve as a more realistic upper bound of model performance on our dataset. We estimate this performance by evaluating the performance of the best annotator of each test input, where best is defined as resulting in the highest F_1 score against the majority-voted gold spans. Note that this is different from the best-matched spans mentioned in

Section 4.1 because here the gold spans are fixed to the majority-voted spans while the predicted spans come from the best annotator. While there are limitations to this estimation (see Limitations), we argue that the estimate is still useful as a point of reference.

5 Results

Table 2 shows that both MV and REL perform substantially better than RANDOM in terms of F_1 scores for all 4 evaluation setups, indicating the potential of both methods. Comparing MV and REL across both types of gold spans, while the former is on par with the latter in word-level evaluation, MV outperforms REL substantially in span-level evaluation. This finding is consistent across precision and recall, and thus demonstrates that MV is overall superior to REL. However, the table also shows that RANDOM outperforms both MV and REL in terms of word-level recall across both types of gold spans, which points to an area for improvement.

While the performance numbers with majority-voted gold spans are lower than the best-matched counterparts, the patterns of model performance are consistent across both types of gold spans. This result suggest that both types of gold spans are equally acceptable for handling subjectivity in span annotations. However, using the majority-voted gold spans has the advantage of time efficiency because the gold spans do not need to be recomputed when evaluating different models.

For majority-voted spans, Table 2a shows that model performance is still far behind expert performance, suggesting that there is still plenty of room for improvement. Furthermore, the expert performance is moderately high in span-level evaluation and approaches perfect performance in the word-level counterpart. This finding demonstrates that the majority-voted spans are realistic as they show a high degree of similarity to span annotations given by experts.

5.1 Experiments with Other Pretrained Language Models

We also experiment with an improved version of BERT known as DeBERTaV3 (He et al., 2023). Key differences include a more complex model architecture, a simpler pretraining objective, and a larger amount of pretraining data. We use the base version of DeBERTaV3 which has the same number of layers, attention heads, and hidden units

but four times the vocabulary size of the base version of BERT, as used in the previous experiment. We evaluate only against the majority-voted gold spans based on the previous findings. Due to time constraints, we use the hyperparameters (learning rate and batch size) tuned on the first fold (out of 20) for all the folds of the dataset.

Table 3 shows that both MV and REL outperform RANDOM substantially on both span- and word-level evaluations across all metrics except for word-level recall where RANDOM achieves the best score. This finding agrees with that of the BERT-based models. Looking at F_1 scores, the table shows that REL is on par with MV in span-level evaluation and marginally outperforms MV in the word-level counterpart. This finding contradicts the results for BERT-based models, suggesting the effectiveness of REL with improved language models.

Furthermore, the table shows that for span-level evaluation, REL outperforms MV in precision but performs worse than MV in recall. In contrast, for word-level evaluation, MV outperforms REL in precision but performs worse than REL in recall. These findings suggest that with stronger language models, the best method depends not only on whether span- or word-level evaluation is prioritised but also on whether precision or recall is more crucial. These patterns of performance again contradict those of the BERT-based models, suggesting that the choice of pretrained language models is important. We leave the analysis on the possible reasons behind these findings and the evaluation on best-matched gold spans for future work.

Lastly, comparing to Table 2a, we see that DeBERTa-based models outperform the BERT-based counterparts across the board. This finding is unsurprising because DeBERTa was developed as an improvement over BERT (He et al., 2021).

6 Related Work

Pruthi et al. (2020) have studied the span prediction problem under the name of evidence extraction. However, their model also performs classification jointly and is trained in a semi-supervised manner. More importantly, they did not consider subjectivity in the span annotations. In contrast, we focus only on predicting spans, supervised learning, and incorporating subjectivity in model training and evaluation.

Previous work has leveraged a similar dataset of

Method	Span			Word		
	P	R	F ₁	P	R	F ₁
RANDOM	0.2 ± 0.0	4.1 ± 0.1	0.4 ± 0.0	17.1 ± 0.0	66.6 ± 0.0	27.2 ± 0.0
MV	17.9 ± 1.9	18.5 ± 0.3	18.2 ± 1.1	58.2 ± 0.4	48.7 ± 0.1	53.0 ± 0.1
REL	11.2 ± 1.4	12.6 ± 0.3	11.8 ± 0.9	57.5 ± 0.7	48.9 ± 1.2	52.8 ± 0.5
Expert	80.2	67.5	73.3	91.0	97.5	94.2

(a) Majority-voted gold spans

Method	Span			Word		
	P	R	F ₁	P	R	F ₁
RANDOM	0.0 ± 0.0	1.2 ± 0.0	0.1 ± 0.0	31.0 ± 0.0	66.9 ± 0.0	42.4 ± 0.0
MV	20.9 ± 2.2	26.3 ± 0.4	23.3 ± 1.5	69.2 ± 0.5	48.6 ± 0.2	57.1 ± 0.1
REL	17.1 ± 2.2	24.1 ± 0.5	19.9 ± 1.7	69.6 ± 0.5	48.7 ± 1.2	57.3 ± 0.7

(b) Best-matched gold spans

Table 2: Span- and word-level precision, recall, and F₁ scores (in %) of the span prediction model against majority-voted and best-matched gold spans. Mean (± std) across 3 runs are reported except for Expert.

Method	Span			Word		
	P	R	F ₁	P	R	F ₁
RANDOM	0.2 ± 0.0	4.1 ± 0.1	0.4 ± 0.0	17.1 ± 0.0	66.6 ± 0.0	27.2 ± 0.0
MV	18.4 ± 1.6	19.7 ± 0.3	19.0 ± 1.0	61.3 ± 0.4	50.2 ± 0.3	55.2 ± 0.0
REL	23.7 ± 2.6	14.8 ± 0.1	18.2 ± 0.8	58.7 ± 0.3	53.0 ± 0.4	55.7 ± 0.2

Table 3: Span- and word-level precision, recall, and F₁ scores (in %) of the DeBERTaV3-based model against majority-voted gold spans. Mean (± std) across 3 runs are reported. RANDOM performance is copied from Table 2a.

legal problem descriptions (Mistica et al., 2021). They focussed on the text classification aspect where areas of law are assigned to a problem description. Different from their work, ours treats the area of law as given and focusses on predicting the spans that support the assignment of the area of law.

Our work falls within the broader theme of human label variation (Plank, 2022). Previous work has mainly focussed on text classification tasks (Leonardelli et al., 2023; Fornaciari et al., 2021; Nie et al., 2020, *inter alia*). In contrast, we focus on spans, which are still understudied in this area. Our work is also related to data perspectivism.⁹

7 Conclusion

We explore the task of automatically predicting text spans in a legal problem description that support the labelling of an area of law. We develop neural sequence taggers that deal with the inherent subjectivity of the task. Experiments across various subjectivity-aware evaluation setups show that

training on majority-voted annotations outperforms training on the disaggregated counterparts.

Limitations

The dataset we use in this work cannot be released publicly, which is a major limitation of our work in terms of reproducibility. This is because the topics discussed are sensitive, and more importantly, the help-seekers have not given their consent to share their data. Nevertheless, we believe our work still offers valuable scientific knowledge on handling subjectivity, especially in span annotation tasks.

For the evaluation using majority-voted gold spans, we estimate the expert performance by determining the best annotator of each test input. However, the majority-voted gold spans are a function of the best annotator’s spans. Thus, the estimated expert performance is dominated by test inputs that are annotated by fewer annotators. To mitigate this issue, a leave-one-annotator-out strategy can be employed, which we leave for future work.

The best-matched gold spans are likely to come from various annotators. Taken together, these spans may not reflect a realistic pattern of a single human annotator. A remedy is to evaluate against a

⁹<https://pdai.info/>

single best annotator. However, this approach is not straightforward in our case because an annotator may annotate only a subset of examples. We thus leave this approach for future work.

Acknowledgements

We thank the anonymous reviewers for their constructive feedback on the paper. This research is supported by the Australian Research Council Linkage Project (project number: LP210200917) and funded by the Australian Government. This research is done in collaboration with Justice Connect, an Australian public benevolent institution.¹⁰

References

- Alan Akbik, Tanja Bergmann, Duncan Blythe, Kashif Rasul, Stefan Schweter, and Roland Vollgraf. 2019. [FLAIR: An easy-to-use framework for state-of-the-art NLP](#). In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics (Demonstrations)*, pages 54–59.
- Ronan Collobert, Jason Weston, Léon Bottou, Michael Karlen, Koray Kavukcuoglu, and Pavel Kuksa. 2011. Natural Language Processing (almost) from Scratch. *Journal of Machine Learning Research*, 12(Aug):2493–2537.
- Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. [BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding](#). In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186.
- Tommaso Fornaciari, Alexandra Uma, Silviu Paun, Barbara Plank, Dirk Hovy, and Massimo Poesio. 2021. [Beyond Black & White: Leveraging Annotator Disagreement via Soft-Label Multi-Task Learning](#). In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 2591–2597.
- Pengcheng He, Jianfeng Gao, and Weizhu Chen. 2023. [DeBERTaV3: Improving DeBERTa using ELECTRA-style pre-training with gradient-disentangled embedding sharing](#). In *The Eleventh International Conference on Learning Representations*.
- Pengcheng He, Xiaodong Liu, Jianfeng Gao, and Weizhu Chen. 2021. [DeBERTa: Decoding-enhanced BERT with disentangled attention](#). In *9th International Conference on Learning Representations*.
- Sepp Hochreiter and Jürgen Schmidhuber. 1997. [Long short-term memory](#). *Neural Computation*, 9(8):1735–1780.
- Guillaume Lample, Miguel Ballesteros, Sandeep Subramanian, Kazuya Kawakami, and Chris Dyer. 2016. [Neural architectures for named entity recognition](#). In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 260–270.
- Elisa Leonardelli, Gavin Abercrombie, Dina Almanea, Valerio Basile, Tommaso Fornaciari, Barbara Plank, Verena Rieser, Alexandra Uma, and Massimo Poesio. 2023. [SemEval-2023 Task 11: Learning with Disagreements \(LeWiDi\)](#). In *Proceedings of the 17th International Workshop on Semantic Evaluation (SemEval-2023)*, pages 2304–2318.
- Chin-Yew Lin. 2004. Rouge: A package for automatic evaluation of summaries. In *Text Summarization Branches out: Proceedings of the ACL-04 Workshop*, volume 8.
- Meladel Mistica, Jey Han Lau, Brayden Merrifield, Kate Fazio, and Timothy Baldwin. 2021. [Semi-automatic Triage of Requests for Free Legal Assistance](#). In *Proceedings of the Natural Language Processing Workshop 2021*, pages 217–227.
- Yixin Nie, Xiang Zhou, and Mohit Bansal. 2020. [What can we learn from collective human opinions on natural language inference data?](#) In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 9131–9143.
- Barbara Plank. 2022. [The “Problem” of Human Label Variation: On Ground Truth in Data, Modeling and Evaluation](#). In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 10671–10682.
- Danish Pruthi, Bhuwan Dhingra, Graham Neubig, and Zachary C. Lipton. 2020. [Weakly- and Semi-supervised Evidence Extraction](#). In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 3965–3970.
- Satoshi Sekine, Ralph Grishman, and Hiroyuki Shinnou. 1998. A decision tree method for finding and classifying names in Japanese texts. In *Sixth Workshop on Very Large Corpora*.
- Victor S. Sheng, Foster Provost, and Panagiotis G. Ipeirotis. 2008. [Get another label? improving data quality and data mining using multiple, noisy labelers](#). In *Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD '08*, pages 614–622.

¹⁰As defined by the Australian government: <https://www.acnc.gov.au/charity/charities/4a24f21a-38af-e811-a95e-000d3ad24c60/profile>

Erik F. Tjong Kim Sang and Sabine Buchholz. 2000. Introduction to the CoNLL-2000 Shared Task Chunking. In *Fourth Conference on Computational Natural Language Learning and the Second Learning Language in Logic Workshop*.