Tackling the Backlog: Support for Completing and Validating Forms

Katie Atkinson [1] David Bareham [1] Trevor Bench-Capon [1] Joe Collenette [2]
Jack Mumford [1]
[a] *Department of Computer Science, University of Liverpool, UK*
[b] *Department of Computer Science, University of Manchester, UK*

Abstract.
In a wide variety of jurisdictions and courts, an accumulation is building up of legal cases waiting to be processed. New technologies offer promising solutions for tackling the backlogs and reducing the build up of cases, but the technologies must be targeted at specific bottlenecks. In this paper we describe work completed to meet these aims in two ways: 1) a tool aimed at reducing the number of incorrectly completed forms submitted; 2) a decision support tool for the identification of inadmissible claims submitted to the court. Both tools were developed for the setting of cases submitted to the European Court of Human Rights. We provide an overview of both these tools and their evaluation in initial trials, with the aim of enabling claimants to get improved access to justice by providing a level of automation for key decision points in the application process.

Keywords. Backlog, Forms, Admissibility

1. Introduction

Access to Justice involves a number of things. First, people must be aware of their rights. Second, they must be able to claim them. Third, they must be granted them. And this needs to be done reasonably promptly, for "Justice delayed is justice denied.[1]"

AI has the potential to address all of these aspects. The first has been tackled by providing consultative expert systems for the relevant domain so that potential claimants may provide their individual details and receive advice as to the likelihood of their claim succeeding. A recent example is [19]. An alternative is to make the advice leaflets issued more accessible by customising them to the individual and directing focus to the relevant parts [18]. To address the second point, assistance can be provided with completing the claim form, to help people to make valid claims (e.g. [11], [12]). For the third point, officials are targeted,

[1] W.E. Gladstone said this in a House of Commons debate on 16th March 1868, possibly alluding to one of the provisions of the 1215 Magna Carta "To no one will we sell, to no one will we refuse or delay, right or justice" (clause 40).

by offering support to those processing claims [6]. In this paper we want to look at one particular problem which affects access to justice by extending the time taken to process claims beyond what is reasonable: the large backlogs that exist in many cases. This involves support at both the second and third points.

One example area affected by backlogs is claims for asylum in the UK. The UK's asylum backlog, that is the number of asylum cases awaiting a Home Office decision, has grown substantially in recent years. On 31 December 2022, 132,182 main applicants were awaiting an initial decision, or 160,919 people if we include family members applying with them. In 2021, UK asylum applications took an average of around 20 months to receive an initial Home Office decision."[2]. Germany has an even larger backlog, although it is able to process claims more quickly.

This is a serious problem, especially since many of these claims will prove successful. People's lives are put on hold for many months, and they must be accommodated while they wait, resulting in excessive costs, and in some cases social problems with local communities.

A second example is the European Court of Human Rights (ECtHR). The Court's backlog is one of the key challenges that the ECtHR is facing, so more automation enabling speedier resolution of the applications is of crucial importance. Although the number of pending applications reduced since 2010 when it reached 150,000, it was still over 60,000 in 2021[3]. It has been estimated that the Court will need years to sort out its backlog even if the influx of new applications were to stop.

One particular aspect is the number of inadmissible applications received. The importance of admissibility should not be underestimated. On average about 90% of all applications submitted to the ECtHR are declared inadmissible. For instance, in 2019, 44,500 applications were submitted to the Court and in the same year 38,480 applications were declared inadmissible. At the same time, in 2019, the Court delivered only 2,187 meritorious judgements[4]. A large number of applications are declared inadmissible every year. Considering inadmissible applications takes a significant proportion of the Court's time and resources which could be re-allocated to the meritorious cases.

Two ways of reducing these backlogs are to reduce the number of incorrectly completed forms, and secondly to support the identification of inadmissible claims submitted to the court. In this paper we discuss tools designed for both purposes. Section 2 describes the underlying model we use to support both applications. Section 3 describes a web-based tool to assist claimants in completing their forms correctly [6], describes tools to support admissibility checking, and reports a small survey carried out with lawyers working for the European Convention on Human Rights (ECHR) [9]. Section 4 offers discussion of future work that is planned to extend on these foundations by incorporating state-of-the-art NLP techniques. Finally, Section 5 presents concluding remarks.

[2]https://migrationobservatory.ox.ac.uk/resources/briefings/the-uks-asylum-backlog/

[3]European Court of Human Rights, Analysis of statistics, 2022. https://www.echr.coe.int/Documents/Stats_analysis_2021_ENG.pdf.

[4]European Court of Human Rights, Analysis of statistics. (2019). https://www.echr.coe.int/Documents/Stats_analysis_2019_ENG.pdf.

2. Modelling the domain with Angelic

Both the applications we describe are founded on a domain model constructed using the Angelic Methodology ([1] and [4]). Angelic is based on several ideas developed in leading work on AI and Law.

It produces an Angelic Design Model (ADM), which corresponds to the factor hierarchy used in CATO [3] and IBP [8], which has the case *outcome* as root, followed by layers of *issues*, then layers of *abstract factors* and has *base level factors* as its leaf nodes.

Issues are the points that must be established if the plaintiff is to win their case. The issues are normally found in the legislation, but in common law domains can be established by a precedent. *Factors* are the aspects of the case that need to be considered when resolving issues. Base level factors are ascribed to a case on the basis of its facts, and are stereotypical patterns of fact that provide a reason to decide an issue for one party or another. *Abstract factors* are used to group base level factors together and provide information for explanation, and for assessing whether a distinction between two cases is significant (see [17].

This structure is augmented by providing each of the nodes with a set of acceptance conditions, derived from statute or precedent cases using the techniques of [16]. These acceptance conditions take the form of a set of rules, each with either *accept* or *reject* as head, and a body comprising only children of the node. The rules are prioritised to reflect preferences shown in precedents, and the final condition is a default, reflecting the burden of proof. This structure corresponds to a particular form of Abstract Dialectical Framework [7].

This structure is further augmented by a set of questions to be posed to the user. The acceptance conditions of the base level factors are defined in terms of responses to these questions. ADMs for the ECHR are given in Appendix A of [9]. The questions are tailored to the target user: in [9] the target application was designed to support the assessment of admissibility and so the questions are targeted at legal trained employees of the ECtHR. The same questions were used in the pilot of the form completion application, the assumption being that the claimant would complete the form with legal advice. For an application usable by claimants alone, some adaptation of the questions to express them in lay terms would be required.

The Angelic methodology has been used and refined in a series of application, both academic and commercial. Among the academic domains are US Trade Secrets ([1] and [4]), the wild animals property cases beginning with *Pierson v Post* ([1]) and the Automobile Exception to the US 4th Amendment ([1]). Several commercial domains were developed for the law firm Weightmans, including claims for Noise Induced Hearing Loss [2].

Angelic has been implemented using a variety of languages and tools: Prolog [1], Java and Javascript [9], with a front end from Kira Systems [2], and a platform designed for distribution on mobile phones, Logiak [5].

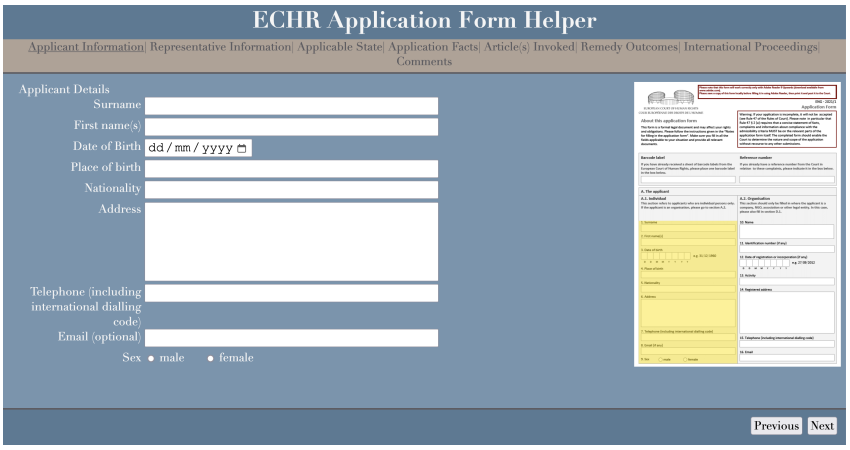

Figure 1. Example of the form support tool showing an information entry screen. The LHS is an amplification of the yellow box on the RHS.

3. Support for Completing Forms

Applications to the ECtHR are not trivial. The form is a complex document which encompasses sections that may or may not need to be completed, along with all the supporting documentation that the applicant should send. We have developed a tool which helps guide an applicant through the application form, generates a filled-in form which requires signing, and then gives a recommendation on whether to submit the application based on answers the applicant gives after they have filled in the form. For the user it is important that the tool not only gives the recommendation but gives an explanation on why that recommendation has been given, so they can if possible, make modifications and produce an admissible application. Ease of use was a key aim in our development. To ensure that the tool can be used by users with minimal computer experience we developed the tool as a web-based form using standard website development tools, HTML, CSS, and Javascript. The look of the website and an example of an information entry screen is shown in Figure 1. In addition the website also highlights the part of the application form they are currently working on.

Once the applicant has filled in all the information, they are able to get a copy of the completed application filled in with this information as a PDF. In addition the applicant is able to get a recommendation on whether to submit the application to the ECtHR. An example page from the generated application and the finishing screen giving the recommendation are shown in Figure 2. To generate a recommendation for the user, the tool uses an ADM with 87 nodes to represent the necessary arguments. There are 74 possible questions that can be posed to the user. The user will only be required to answer the questions relevant to them, as determined by their answers. Some questions overlap as they are based on previous answers, and the tool can answer some of the questions for the applicant by taking information the applicant has already given. An example

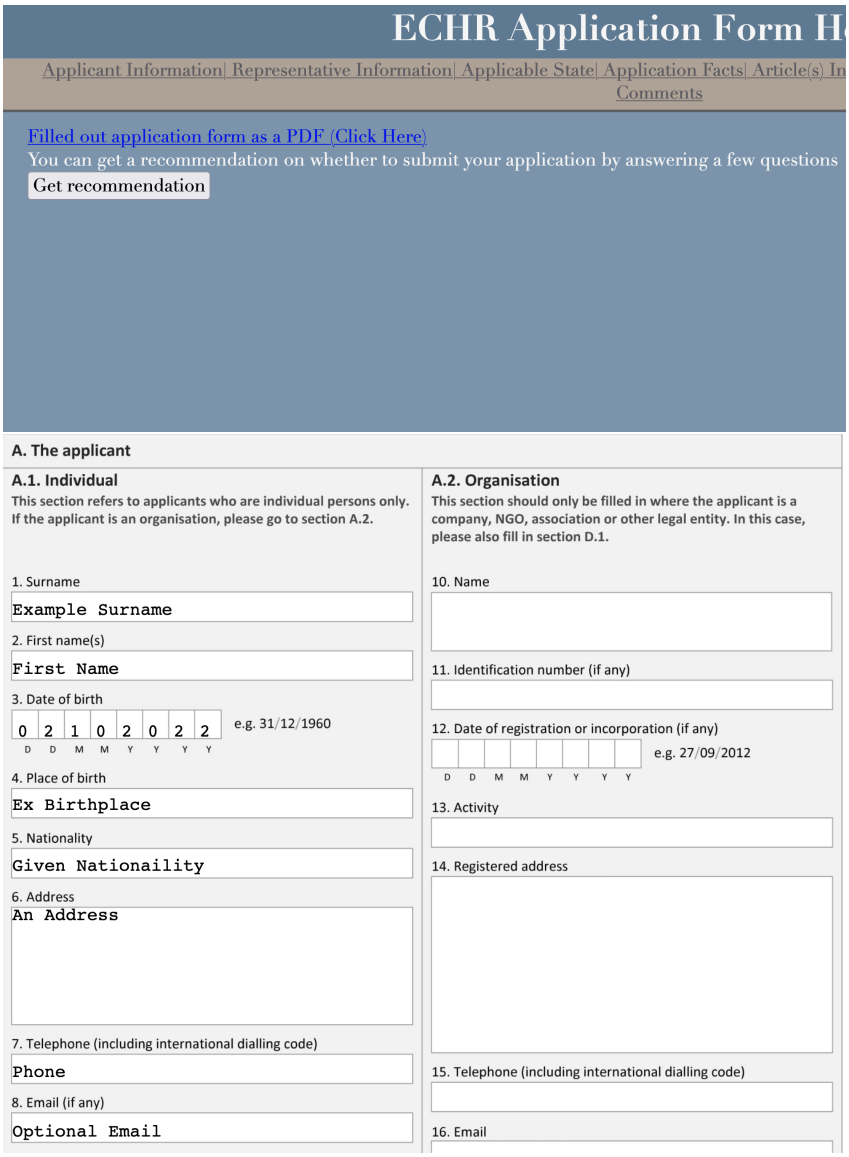

Figure 2. The final page of the tool whether the user is able to get a recommendation (Top) and the first page of a filled in application form that is generated by the tool (Bottom)

of a recommendation is shown in Figure 3. All the questions posed to the user are yes/no questions, a recommendation is given as soon as the ADM is able to resolve from the information it has been given.

3.1. Explanation Generation

Explanations are generated when a recommendation has been reached. The nodes in the ADM generate a human-readable sentence which explains the current node

Figure 3. Recommendation not to submit to the ECtHR as the applicant is not a victim as they have had adequate redress.

to the user. All the individual explanations from each of the nodes are concatenated with some connecting text which generates the full explanation to the user. Not all questions will have been asked, therefore not all the nodes will be able to resolve. Therefore when generating the explanation we only include the nodes which can resolve.

When generating an explanation for a recommendation not to submit the application we take a more targeted approach to generating the explanation. The user will want the explanation to reflect why the tool has given this particular recommendation. To achieve this we only generate explanations from the nodes that are relevant to the node which causes the ADM to recommend not to submit the application. As we know that the support form ends as soon as an application is recommended not to be submitted there will only be one factor which is relevant.

The following nodes will be included in generating the targeted explanation: The node which generates the negative recommendation, all parent nodes of the negative node, and all resolvable child nodes of the parent of the negative node. Using the example of Figure 3, the negative node is "The applicant has lost victim status through redress" and the parent node is "The applicant does not have victim status" but we also include the node "The applicant is a direct victim" as this node is also a child of the node "The applicant does not have victim status" and is relevant to answering why "The applicant does not have victim status".

Once an application form has been completed we have developed a tool directed at supporting those parties responsible for determining admissibility. This tool is very similar to the form completion tool. It is based on the same ADM, and the process is the same as that which provides the recommendation as to whether to submit.

From previous experience with lawyers in the ECtHR, it is difficult to provide a program that can be run on their systems as they are locked down and the lawyers may not have the computer skills to ensure that the program runs on their home systems. We have therefore delivered the application on a standard web browser. The flow of the tool is that once the user has started the process, they are asked a number of questions one by one then a recommendation is given with an explanation.

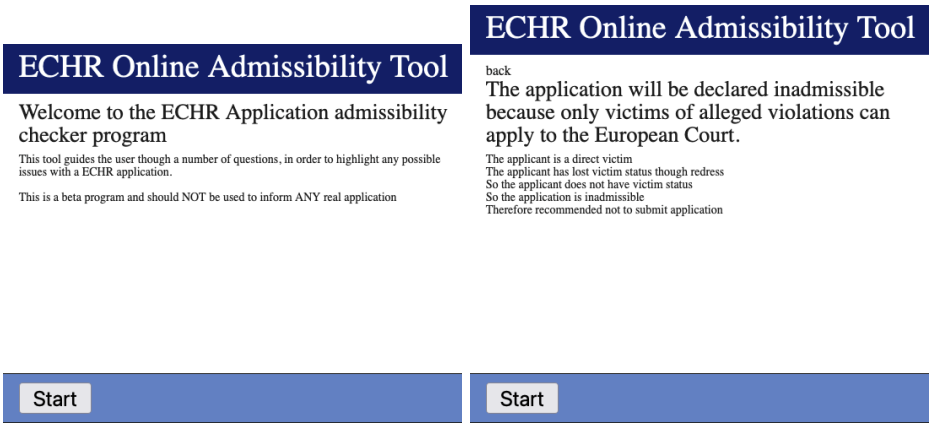

Figure 4. Example of the admissibility tool showing a question page (LHS), and the outcome page (RHS)

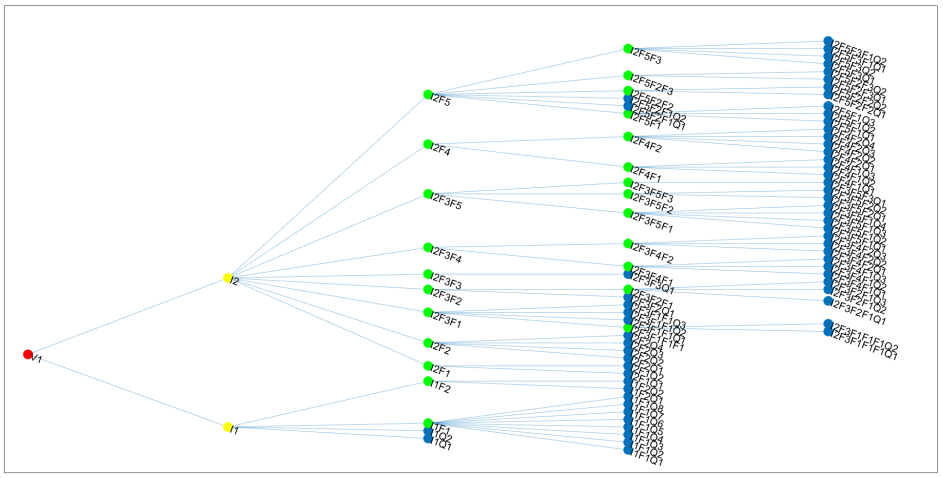

Figure 5. Full tree representation of the ADM used in the support tool, along with IDs for each node. The connecting paths show a parent child relation with the parents on the left

Figure 4 shows an example of additional information which is given to the user to help answer the question "Is the applicant a direct victim?" The full ADM tree is shown in Figure 5 along with the IDs of each node.

3.2. Usability Survey

A key aim in the design of this project was to create a tool which practitioners within the ECtHR would find useful. This aim provided the motivation to conduct a usability survey for the web based tool using a sample from the intended user audience (a small group of ECtHR lawyers).

The survey comprised of questions covering 5 different aspects of the tool: functionality, usability, explainability, usefulness, and feedback on the questions

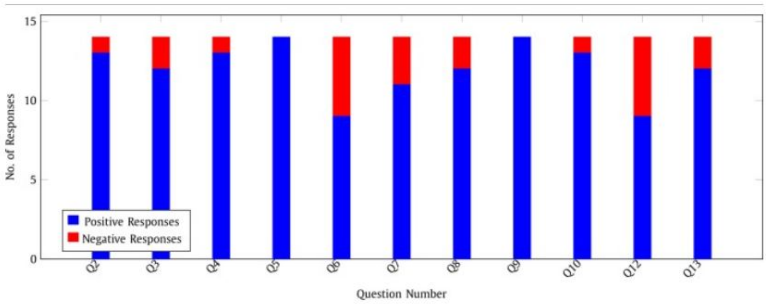

Figure 6. Summary of results from second survey showing split between positive and negative responses

used in the tool. Each question offered four responses, one very positive, one moderately positive, one moderately negative and one very negative.

Initially a pilot survey was conducted with a very small sample of 3 ECtHR lawyers to provide a preliminary assessment as to the applicability, functionality and usability of the tool. The positive responses, reported by the lawyers involved in the pilot study, motivated the extension of this to a wider group of 14 ECtHR lawyers. The results from the wider group were also highly encouraging regarding the utility of the tool, with almost all participants finding the tool useful, and most agreeing that the generated explanations were easy to read and that the tool could save them time. The results from the second survey are shown in Figure 6.

The results from the usability study help to demonstrate the effectiveness of the reasoning models for the issue of admissibility and the willingness of stakeholders to make use of such a tool.

4. Incorporating NLP for Enhanced Recommendations

The use of ADMs to aid applicants in applying to the ECtHR has shown promise in enhancing access to justice and streamlining the Court's processing time. These forms simplify the submission process and help in making preliminary assessments regarding the admissibility of cases, garnering encouraging feedback from ECtHR legal practitioners.

While the factor-based approach of ADMs provides a high-level abstraction for recommendations, there is an opportunity to delve deeper. Current implementation of the forms focus on broader legal principles without assessing the nuanced, specific facts of each case – a process that would be intractable for manual symbolic encoding within the ADM. Therefore, future work aims to harness sophisticated NLP methods to process individual case facts and map them accurately to the ADM's leaf nodes.

Recent work ([14], [15]) has demonstrated the efficacy of state-of-the-art NLP models in ascribing factual case details to an ADM structure. Incorporating these NLP techniques into the user forms could allow for more detailed descriptions in be included, leading to more finely tuned recommendations. Moreover, the attention mechanisms ([10], [13]) inherent in advanced NLP models, can elucidate the

connection between an application's specific facts and the output recommendation. For instance, we saw in Figure 3 that the recommendation against submission was based on the applicant's loss of victim status, with the finest point of explanation corresponding to the leaf node in the ADM denoting that 'the applicant has lost victim status through redress'. Yet no explanation is available as to why this leaf node is rejected. Extending the form to include the attention mechanism could highlight the exact factual statements and relevant case law that led to this ascription. This level of detail is currently absent but could significantly enhance the transparency and usefulness of the recommendations provided.

Envisioning a hybrid ADM-NLP system holds substantial promise for the future. Such a system could refine the user forms to capture finer details of individual cases, aiding legal practitioners in quickly processing applications and providing applicants with a clear understanding of admissibility criteria. By offering pointed insights into how the factual nuances of their cases align with ECtHR requirements, this approach could become an invaluable tool in legal preparation and case submission processes.

5. Concluding Remarks

In this paper, we have presented a suite of user input forms designed to standardise and streamline the ECtHR case application process. Our aim is to enhance access to justice by simplifying the application experience for individuals and by accelerating the court's ability to filter out inadmissible cases – which constitute the majority. This approach not only augments the efficiency of the court system but also contributes to reducing the existing backlog and improving case turnaround times.

Feedback from ECtHR lawyers regarding our current form implementations has been overwhelmingly positive, underscoring the practical benefits of our work. Moreover, we have outlined our intention to enrich these forms by integrating advanced NLP techniques. This enhancement will enable us to provide more granular and effective recommendations based on detailed factual case descriptions submitted by users, alongside pertinent case law. Through this innovation, we anticipate offering even greater support to applicants and legal professionals, furthering the cause of accessible and swift justice.

References

[1] Latifa Al-Abdulkarim, Katie Atkinson, and Trevor Bench-Capon. A methodology for designing systems to reason with legal cases using abstract dialectical frameworks. *Artificial Intelligence and Law*, 24:1–49, 2016.

[2] Latifa Al-Abdulkarim, Katie Atkinson, Trevor Bench-Capon, Stuart Whittle, Rob Williams, and Catriona Wolfenden. Noise induced hearing loss: Building an application using the angelic methodology. *Argument and Computation*, 10(1):5–22, 2019.

[3] Vincent Aleven. Using background knowledge in case-based legal reasoning: a computational model and an intelligent learning environment. *Artificial Intelligence*, 150(1-2):183–237, 2003.

[4] Katie Atkinson and Trevor Bench-Capon. Angelic II: An improved methodology for representing legal domain knowledge. In *Proceedings of the 19th International Conference on Artificial Intelligence and Law*, pages 12–21, 2023.

[5] Katie Atkinson, Trevor Bench-Capon, Tom Routen, Alejandro Sánchez, Stuart Whittle, Rob Williams, and Catriona Wolfenden. Realising ANGELIC designs using Logiak. 2019.

[6] Katie Atkinson, Joe Collenette, Trevor Bench-Capon, and Kanstantsin Dzehtsiarou. Practical tools from formal models: the echr as a case study. In *Proceedings of the 18th International Conference on Artificial Intelligence and Law*, pages 170–174, 2021.

[7] G. Brewka and S. Woltran. Abstract dialectical frameworks. In *Principles of Knowledge Representation and Reasoning: Proceedings of the 12th International Conference*, pages 102–111. AAAI Press, 2010.

[8] Stephanie Brüninghaus and Kevin Ashley. Predicting outcomes of case based legal arguments. In *Proceedings of the 9th International Conference on Artificial Intelligence and Law*, pages 233–242, 2003.

[9] Joe Collenette, Katie Atkinson, and Trevor Bench-Capon. Explainable AI tools for legal reasoning about cases: A study on the European Court of Human Rights. *Artificial Intelligence*, 317:103861, 2023.

[10] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

[11] David Frohlich, L Crossfield, and Nigel Gilbert. Requirements for an intelligent form-filling interface. In *People and computers: Designing the interface*, pages 102–117. Cambridge University Press, 1985.

[12] Nigel Gilbert. The claimant advice systems. In Trevor Bench-Capon, editor, *Knowledge-Based Systems and Legal Applications*, pages 183–198. Academic Press, 1991.

[13] Jinghui Lu, Maeve Henchion, Ivan Bacher, and Brian Mac Namee. A sentence-level hierarchical bert model for document classification with limited labelled data. In *Proceedings of DS 2021*, pages 231–241. Springer, 2021.

[14] Jack Mumford, Katie Atkinson, and Trevor Bench-Capon. Reasoning with legal cases: A hybrid ADF-ML approach. In *Proceedings of JURIX 2022*, pages 93–102, 2022.

[15] Jack Mumford, Katie Atkinson, and Trevor Bench-Capon. Combining a legal knowledge model with machine learning for reasoning with legal cases. In *Proceedings of the 19th ICAIL*, pages 167–176, 2023.

[16] Henry Prakken and Giovanni Sartor. Modelling reasoning with precedents in a formal dialogue game. *Artificial Intelligence and Law*, 6:231–287, 1998.

[17] Henry Prakken, Adam Wyner, Trevor Bench-Capon, and Katie Atkinson. A formalization of argumentation schemes for legal case-based reasoning in aspic+. *Journal of Logic and Computation*, 25(5):1141–1166, 2015.

[18] Paul Soper and Trevor Bench-Capon. Coupling hypertext and knowledge based systems: two applications in the legal domain. *Artificial Intelligence and Law*, 2(4):293–314, 1993.

[19] Hannes Westermann and Karim Benyekhlef. Justicebot: A methodology for building augmented intelligence tools for laypeople to increase access to justice. In *Proceedings of the 19th International Conference on Artificial Intelligence and Law*, pages 351–360, 2023.