Deep Neural Networks for Information Mining from Legal Texts

Ilias Chalkidis

Ph.D. Thesis

Department of Informatics
Athens University of Economics and Business

Supervisors:

Ion Androutsopoulos, Manolis Koubarakis
and Nikolaos Aletras

2021

Abstract

Legal text processing (Ashley, 2017) is a growing research area where Natural Language Processing (NLP) techniques are applied in the legal domain. There are several applications such as legal text segmentation (Mencia, 2009; Hasan et al., 2008), legal topic classification (Mencia and Fürnkranzand, 2007; Nallapati and Manning, 2008), legal judgment prediction and analysis (Wang et al., 2012; Aletras et al., 2016), legal information extraction (Kiyavitskaya et al., 2008; Dozier et al., 2010; Asooja et al., 2015), and legal question answering (Kim et al., 2015b, 2016b). These applications and relevant NLP techniques arise from three main sub-domains, i.e, legislation, court cases, and legal agreements (contracts). In all three sub-domains, documents are much longer than in most other modern NLP applications. They also have different characteristics concerning the use of language, the writing style, and their structuring, compared to non-legal text.

Given the rapid growth of deep learning technologies (Goodfellow et al., 2016; Goldberg, 2017), the goal of this thesis is to explore and advance deep learning methods for legal tasks, such as contract element and obligation extraction, legal judgment prediction, legal topic classification, and information retrieval, that have already been discussed in the literature (but not in the context of deep learning) or that were first addressed during the work of this thesis. In this direction, we aim to answer two main research questions: First and foremost on the adaptability of neural methods that have been proposed for related NLP tasks in other domains and how they are affected by legal language, writing, and structure; and second on providing explanations of neural models' decisions (predictions).

Considering the first research question we find and highlight several cases, where either legal language affects a model's performance or suitable modeling is needed to imitate the document structure. To this end, we pre-train and use in-domain word representations and neural language models, while we also propose new methods with state-of-the-art performance. With respect to model explainability, we initially experiment with saliency (attention) heat-maps and highlight their limitations as a means for the explanation of the model's decisions, especially in the most challenging task of legal judgment prediction, where it is most important. To overcome these limitations we further study rationale

extraction techniques as a prominent methodology towards model explainability.

In lack of publicly available annotated datasets in order to experiment with deep learning methods, we curate and publish five datasets for various legal tasks (contract element extraction, legal topic classification, legal judgment prediction and rationale extraction, and legal information retrieval), while we also publish legal word embeddings and a legal pre-trained language model to assist legal text processing research and development.

We consider our work, a first, fundamental, step among other recent efforts, towards improving legal natural language understanding using state-of-the-art deep learning techniques, which further promotes the adaptation of new technologies and sheds light on the emerging field of legal text processing.

Acknowledgments

First of all, I would like to thank my supervision committee that supported me in this journey. First and foremost, Ion Androutsopoulos, who offered me this opportunity, supported me and guided me. I would also like to thank him for pushing me to aim always higher and not compromise with mediocre work, while he also taught me how to be an honest, humble and decent researcher. Secondly, I would like to thank Manolis Koubarakis, who introduced me in the exciting field of Artificial Intelligence. But mostly because he believed in me, when I probably didn't believe in myself, and offered me the opportunity to work and do research on his side, while he was also supportive in my move to Athens University of Economics and Business. Last but not least, I would like to thank Nikos Aletras, who accepted to join this committee and offered valuable insights with respect to Natural Language Processing and his expertise in legal applications.

I would especially like to thank my colleagues, Makis Malakasiotis, who acted as a fourth supervisor in this thesis, and Manos Fergadiotis. Thank you both for your advice, encouragement, tolerance, and endless time of working on many of the projects that are part of this thesis. This thesis would not have reached the same extent and the same level of detail without you and our common passion for research, my friends.

I would like to thank my whole family (my mother, my father, my brother, Mandana, and Eleftheria), who believed in me and supported me unconditionally. You taught me to be kind but honest with people, to care for and love my work, to dream but be grounded, to be fearless and have self-esteem, to care for, listen to, and love people. All of these helped me to move forward, meet my goals and set new and bigger ones.

I would also like to thank all of those who supported me financially all these years. My former employer and colleague, Vasilis Tsolis on behalf of Cognitiv+, and my current manager and colleague, George Paliouras on behalf of the Institute of Informatics & Telecommunications at NCSR "Demokritos". This work would not have been possible without their support and tolerance, both very much needed for a PhD student.

I would also like to thank the following colleagues in no particular order: Achilleas Michos, Hamid Motahari, Dimitris Tsarapatsanis, Enrico Francesconi, George Leledakis, Nikolaos Manginas, Eva Katakalou, Giannos Koutsikakis, and Iosif Angelidis.

Contents

List of Figures

List of Tables

List of Thesis Publications

Relevant to Chapter 1: Introduction

- I. Chalkidis and D. Kampas. Deep Learning in Law: Early Adaptation and Legal Word Embeddings Trained on Large Corpora. *Artificial Intelligence and Law*, 27 (2):171–198, June 2019. ISSN 0924-8463

Relevant to Chapter 2: Information Extraction for legal documents

- I. Chalkidis, I. Androutsopoulos, and A. Michos. Extracting contract elements. In *Proceedings of the 16th International Conference on Artificial Intelligence and Law (ICAIL)*, pages 19–28, London, UK, 2017

- I. Chalkidis and I. Androutsopoulos. A deep learning approach to contract element extraction. In *Proceedings of the 30th International Conference on Legal Knowledge and Information Systems (JURIX)*, pages 155–164, Luxembourg, 2017

- I. Chalkidis, I. Androutsopoulos, and A. Michos. Obligation and Prohibition Extraction Using Hierarchical RNNs. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (ACL)*, Melbourne, Australia, 2018

- I. Chalkidis, M. Fergadiotis, P. Malakasiotis, and I. Androutsopoulos. Neural Contract Element Extraction Revisited. In *Proceedings of the Document Intelligence Workshop of the 33rd Conference on Neural Information Processing Systems (NeurIPS)*, Vancouver, Canada, 2019d

Relevant to Chapter 3: LMTC for legal documents

- I. Chalkidis, M. Fergadiotis, P. Malakasiotis, N. Aletras, and I. Androutsopoulos. Extreme Multi-Label Legal Text Classification: A case study in EU Legislation. In *Proceedings of the Annual Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT)*, Minneapolis, USA, 2019b

- I. Chalkidis, M. Fergadiotis, P. Malakasiotis, and I. Androutsopoulos. Large-Scale Multi-Label Text Classification on EU Legislation. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics (ACL)*, Florence, Italy, 2019c

- I. Chalkidis, M. Fergadiotis, S. Kotitsas, P. Malakasiotis, N. Aletras, and I. Androutsopoulos. An empirical study on large-scale multi-label text classification including few and zero-shot labels. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 7503–7515, Online, 2020a

Relevant to Chapter 4: Legal Judgment Prediction and Explainability

- I. Chalkidis, I. Androutsopoulos, and N. Aletras. Neural Legal Judgment Prediction in English. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics (ACL)*, Florence, Italy, 2019a

- I. Chalkidis, D. Tsarapatsanis, M. Fergadiotis, P. Malakasiotis, N. Aletras, and I. Androutsopoulos. Paragraph-level Rationale Extraction through Regularization: A case study on European Court of Human Rights Case. In *Proceedings of the Annual Conference of the North American Chapter of the Association for Computational Linguistics (NAACL 2021)*, Online, 2021b

Relevant to Chapter 5: Legal Document to Document Information Retrieval

- I. Chalkidis, M. Fergadiotis, N. Manginas, E. Katakalou, and P. Malakasiotis. Regulatory Compliance through Doc2Doc Information Retrieval: A case study in EU/UK legislation where text similarity has limitations. In *16th Conference of the European Chapter of the Association for Computational Linguistics (EACL 2021)*, Online, 2021a

Other relevant work

- I. Chalkidis, M. Fergadiotis, P. Malakasiotis, N. Aletras, and I. Androutsopoulos. LEGAL-BERT: The muppets straight out of law school. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 2898–2904, Online, 2020b

- N. Manginas, I. Chalkidis, and P. Malakasiotis. Layer-wise guided training for BERT: Learning incrementally refined document representations. In *Proceedings of the Fourth Workshop on Structured Prediction for NLP*, pages 53–61, Online, 2020

List of Thesis Resources

Annotated Datasets

- US-CONTRACTS-NER: US Contracts annotated with contract elements. 1,000 contracts with gold annotations for clause headings (CONTRACT STRUCTURE), 2,500 contracts with gold annotations for contract elements of CONTRACT HEADER, CONTRACT TERMINATION and APPLICABLE LAW.
  Link: `http://nlp.cs.aueb.gr/software_and_datasets/CONTRACTS_ICAIL2017`
- EURLEX57K: 57,000 European legislation annotated with EUROVOC concepts.
  Link: `http://nlp.cs.aueb.gr/software_and_datasets/EURLEX57K`
- ECtHR v1: 11,000 ECtHR cases, in English, annotated with violations of ECHR articles.
  Link: `https://archive.org/details/ECHR-ACL2019`
- ECtHR v2: 11,000 ECtHR cases, in English, annotated with alleged violations and silver and gold rationales.
  Link: `https://archive.org/details/ECHR-NAACL2021`
- REGULATORY COMPLIANCE IR: EU2UK and UK2EU datasets, contain legislation in English, based on transpositions of EU legislation by UK legislation.
  Link: `https://archive.org/details/eacl2021_regir_datasets`

Word Embeddings

- LAW2VEC: 200-dimensional English legal word embeddings trained using WORD2VEC (skip-gram model) with more than 100,000 legal documents from various public domains (e.g., legislation from EU, UK, Canada, Australia and US, etc.)
  Link: `https://archive.org/details/Law2Vec`

Pre-trained Language Models

- LEGAL-BERT: A family of English BERT models for the legal domain.
  Link: `https://huggingface.co/nlpaueb/legal-bert-base-uncased`

Chapter 1

Introduction

1.1 Overview of this thesis

Legal text processing (Ashley, 2017) is a growing research area where Natural Language Processing (NLP) techniques are applied in the legal domain. There are several applications, such as legal text segmentation (Mencia, 2009; Hasan et al., 2008), legal topic classification (Mencia and Fürnkranzand, 2007; Nallapati and Manning, 2008), legal judgment prediction and analysis (Wang et al., 2012; Aletras et al., 2016), legal information extraction (Dozier et al., 2010), and legal question answering (Kim et al., 2015b, 2016b), which mainly target three sub-domains: legislation, court cases, and legal agreements. A large number of companies, including hundreds of start-ups, operate in the emerging legal-tech and reg-tech industries in order to provide text analytics,[1] targeting a wide variety of use cases that are currently poorly handled due to the excessive amount of data (documents), which is difficult to be analyzed by humans. The interest for public access to all sorts of legal documents and the use of intelligent legal services is growing rapidly, leading to disputes between established legal service providers and newcomers (start-ups), who aim to build new services and platforms.[2] These tectonic changes in the law industry may have contributed to the introduction of new measures for the simplification of legal language, i.e., the Plain Writing Act 2010 (U.S.A.) and the Plain Language Act 2020 (Ireland),[3] and public open access to law, i.e., Open Courts Act 2020 (U.S.A.).[4]

[1]"713% Growth: Legal Tech Set An Investment Record In 2018", V. Pivovarov, Forbes, 2019, https://www.forbes.com/sites/valentinpivovarov/2019/01/15/legaltechinvestment2018

[2]"ROSS Shuts Down Operations, Citing Financial Burden From Thomson Reuters Lawsuit", R. Dipshan, Law.com, 2020, https://www.law.com/legaltechnews/2020/12/11/ross-shuts-down-operations-citing-financial-burden-from-thomson-reuters-lawsuit

[3]"Cross-party support for new Irish Plain Language Bill", L. Austen-Gray, europa.eu, 2019, https://epale.ec.europa.eu/en/content/cross-party-support-new-irish-plain-language-bill

[4]"House Passes Bill for PACER", J.T., Law.com, 2020, https://www.law.com/nationallawjournal/2020/12/08/rejecting-opposition-from-judiciary-house-passes-bill-to-make-pacer-free

Considering the aforementioned reality and the rapid growth and adaptation of deep learning technologies in NLP (Goodfellow et al., 2016; Goldberg, 2017), the goal of this Ph.D. thesis is to explore and advance deep learning methods for various legal tasks across different fields of law (legislation, court cases, contracts), jurisdictions (EU, UK, and US) and NLP areas (information extraction, text classification, information retrieval), including tasks that have already been discussed in the literature or arisen in this thesis. More specifically this thesis seeks to answer two main research questions:

- How well do Deep Neural Networks (DNNs) that have been proposed for NLP tasks in other domains (e.g., named entity recognition in news articles, movie reviews sentiment classification, biomedical information retrieval) perform in the legal domain (e.g., when attempting to extract contractor names, start/end dates, amounts from contracts or classify and retrieve relevant legislation or case law)? Does the legal language and writing affect their performance? How can they be made more effective for the legal domain (e.g., capture legal language and document structure)?

- Can we provide explanations to support the decisions (predictions) of systems? Current DNNs are often worse than traditional rule-based approaches and systems with manually crafted features when it comes to producing explanations of their outputs. This is also a very hot research area in DNNs for NLP (Serrano and Smith, 2019; Jain and Wallace, 2019; Wiegreffe and Pinter, 2019), but it has not been yet explored in the legal domain, where it is of crucial importance in many cases (e.g., when predicting the outcome of a trial) given the sensitivity of the domain.

1.1.1 The three pillars of Law

The scope of Law can be divided into two domains. Public law concerns government and society, including constitutional law, administrative law, and criminal law. Private law deals with legal disputes between individuals and/or organizations in areas such as contracts, property, and commercial law. From a text processing point of view in this work, we mainly consider a more typical and fundamental distinction categorizing legal documents in three major collections. In general, legal documents, also referred to as legal instruments, means any formally executed written document that can be formally attributed to its author (e.g., legislature, court officials, or any other person or organization), which records and expresses a legally enforceable act, right or obligation. The term "legal documents" may refer, but is not limited to any document filed by legislatures, courts, or any other authority (quasi-judicial body) that has the right to interpret the law. It can also refer to any other deed, mortgage, or contract. In this work, we experiment with datasets derived from three major categories of legal documents:

Legislation: Legislation is law that has been promulgated by a legislature. A legislature is a body with the authority to make laws, commonly parliaments or congresses. In this thesis, we consider United Kingdom (UK) and European Union (EU) legislation in Chapters 3-5. In UK, legislation can be introduced mainly by the parliaments of England and Scotland and the national assemblies of Wales and Northern Ireland.[5] In EU, legislation is usually proposed by the European Commission and approved by the Council of the European Union and European Parliament to become law.[6]

Court documents: Court documents are a broad collection of documents relevant to legal cases and proceedings. Briefly we can categorize court documents following typical judicial processes into: (a) A pleading, which is a written presentation by a litigant in a lawsuit setting forth the facts upon which the litigant claims legal relief or challenges the claims of the defendant. They are usually followed by hearings. (b) A hearing which is a formal examination before a judge according to the laws of a particular jurisdiction. Considering parties' positions and hearings, the court (judges) usually proceed in a formal judgment. (c) A judgment, which is a decision of a court regarding the rights and liabilities of parties in a legal action or proceeding. In most cases, the aforementioned documents are not publicly available as a whole. In this project, we consider cases from the European Court of Human Rights (ECtHR) in Chapter 4. ECtHR judges discuss and rule on the violation, or not, of specific articles of the European Convention of Human Rights (ECHR) as being alleged by the applicants (civilians or organizations) against the defendants (EU states).

Legal agreements (contracts): A legal agreement (contract) is a "legally binding document between at least two parties that defines and governs the rights and duties of the parties to an agreement" (Fergus, 2006). A contract typically involves the exchange of goods (e.g., purchase agreements), services (e.g., service agreements), money (e.g., loan and credit agreements), property (e.g., lease agreements) among others (e.g., employment agreements, NDAs etc.). The formation of a contract requires offer and acceptance between parties. In practice, contracts are drafted and negotiated until the parties accept the terms as a whole. In this theiss, we consider U.S. Contracts in Chapter 2 that have been publicly filed in the U.S. Securities and Exchange Commission (SEC).

[5]The entire collection of UK legislation is publicly available at `https://www.legislation.gov.uk`. For additional details on the law-making procedure refer to `https://www.legislation.gov.uk/understanding-legislation#Howlegislationworks`.

[6]EU legislation is publicly available at `https://eur-lex.europa.eu`. For additional details on the law-making procedure refer to `https://europa.eu/european-union/law`.

1.1.2 Why is legal language any different?

As with other specialized domains (e.g., biomedical, finance), legal text (e.g., legislation, court documents, contracts) has distinct characteristics compared to generic corpora, such as specialized vocabulary, peculiar syntax, semantics based on extensive domain-specific knowledge, etc., to the extent that legal language is often classified as a 'sublanguage' (Tiersma, 1999; Williams, 2007; Haigh, 2018). In fact, there are different kinds (genres) of legal writing: for example, academic legal writing as in law journals, juridical legal writing as in court judgments, or legislative legal writing as in laws, regulations, and contracts (Bhatia, 1994). Our work focuses on English, so here we describe how legal English differs from standard English. In terms of the vocabulary, the most important differences are the following:

Legal terminology. Similarly to the language used in other domains (finance, medical, etc.), legal English includes technical terminology that is not common for the general population (e.g., 'waiver', 'restrictive covenant', 'promissory estoppel', 'tort', 'fee simple', 'arbitration', 'novation').

In-domain use of words. There are also ordinary words that are being used with special meanings in the legal sector. For example 'consideration' is usually mentioned in contracts to denote a promise from one party to another. The word 'action' usually refers to a 'lawsuit', 'sentence' refers to 'punishment', 'executed' is used in contract meaning that an agreement is signed and effective, and of course, the word 'party' refers to a legal entity (person or organization) engaged in a legal agreement.

Use of French and Latin. Legal English borrows words and phrases from French (e.g., 'estoppel', 'laches', and 'voir dire') and Latin ('certiorari', 'habeas corpus', 'inter alia', 'sub judice'), which are considered foreign words in standard English.

Use of pronominal adverbs. Legal scholars use pronominal adverbs dating from the 16th century that are not used in standard English nowadays. These words are usually formed with the words 'here', 'there' and 'where' as a prefix and have derivatives by including '-at', '-in', '-after', '-before', '-with', '-by', '-above', '-on', '-upon' as a suffix (e.g., 'herein', 'hereto', hereby, 'whereby', and 'wherefore'). Their use in legal English is primarily to avoid repeating names or phrases, such as "the parties hereto" instead of "the parties to this contract".

Moreover with respect to syntax and document structure, there are the following notable differences:

Use of long sentences. In common English, the guidelines recommend an average of 15–20 words per sentence (Cutts, 2009), which is usually followed, as smaller and clearer

sentences improve text readability. However, the complexity of legal text often leads to extensive sentences, which are 15-70 words on average. The difference is more widespread in contracts, where the average sentence is 20-70 words, comparing to legislation and court documents, where the average sentence is 15-45 words.

Unusual word order. In many cases, there is a notable difference in the word order used compared to standard English, especially in contracts. For example, *"the provisions for termination hereinafter appearing or will at the cost of the borrower forthwith comply with the same"* and *"In cases in which a claimant receives reimbursement under this subpart for expenses that also will or may be reimbursed from another source, the claimant shall subrogate the United States to the claim for payment from the collateral source up to the amount for which the claimant was reimbursed under this subpart."*.

Extensive use of bullet lists. In legal writing, especially contract and legislative drafting, bullet lists are extensively used to break down complicated information, i.e., sub-cases and exceptions with respect to an event, or combination of terms to fulfil an action. They are usually structured as conditional expressions, similar to those used in programming languages (nested if-then-else statements). For example, *"In the event of X, Seller shall (a) If Y1 is the case, do A1; (b) If Y2 is the case, do A2."*

To further highlight the importance of the distinction, in the past there was also a formal degree associated with legal English. The International Legal English Certificate (ILEC)[7] was considered a high-level English language qualification for lawyers. The ILEC exam was discontinued in December 2016. The discontinuation can be viewed on par with a general public interest to simplify legal writing, also referred to as *legalese*. In this direction, in 1978 U.S. President Carter issued executive orders intended to make government regulations "cost-effective and easy-to-understand by those who were required to comply with them", which later became a federal requirement with the "Plain Writing Act 2010" under President Obama's administration.[8] Similar concerns and measures to simplify legal writing were part of the public discourse in the UK, as early as in 1950, and has been intensified after the foundation of the Plain English Campaign in 1979. Also, a new bill named "Plain Language Act", was drafted in Ireland in 2019, and will possibly get enacted soon after COVID-19 pandemic with cross-party support.[9][10][11] All the aforementioned events highlight a growing trend towards simpler language in official

[7] https://www.britishcouncil.es/en/exam/ilec-international-legal-english-certificate

[8] https://www.plainlanguage.gov/about/history/

[9] http://www.ourcivilisation.com/smartboard/shop/gowerse/complete/chap1.htm

[10] http://www.plainenglish.co.uk

[11] https://epale.ec.europa.eu/en/content/cross-party-support-new-irish-plain-language-bill

and legal documents (e.g., legislation, regulations, etc.).

Given the fact that such active measures were only recently adopted, if at all, we suspect that legal language affects the performance of generic NLP models to a greater or lesser degree. Modern neural NLP methods typically rely on pre-trained word embeddings or pre-trained language models. Hence, we suspect that pre-training with in-domain corpora in order to better capture the aforementioned characteristics may provide in-domain knowledge that is missing from generic corpora that are usually used to pre-train generic NLP models. Furthermore, structural differences, such as long sentences and bullet lists can also affect models' performance, in case long-distance relationships and document's structure play an important role in specific legal tasks.

1.1.3 Early Adoption of Deep Learning in the Legal Domain

Recently, Deep Learning (Goodfellow et al., 2016; Goldberg, 2017; Charniak, 2019) has gained significant attention in the Natural Language Processing (NLP) research community, like in many areas of Artificial Intelligence, as a promising family of techniques. Deep Neural Networks (DNNs) have been rapidly replacing rule-based approaches, dictionary-based models, and traditional machine learning techniques (i.e., linear models, decision trees), which in their majority require intensive manual feature engineering. Manual feature engineering and traditional bag-of-words representations cannot effectively capture polysemy, synonyms, and semantics in general, contrary to neural continuous vector representations (Mikolov et al., 2013b,c) used by DNNs. The above-said techniques fall short in capturing language semantics and linguistic structures along with their long-distance relationships (Bengio et al., 2003; Mikolov et al., 2010; Kalchbrenner and Blunsom, 2013).

On the other hand, DNNs mainly rely on two characteristic technical features, multi-layering and non-linear activations, i.e., stacking layers that typically perform matrix multiplications followed by non-linear activations to learn and enhance feature representations with respect to the network's objectives. Thus, we usually refer to DNNs as feature learners or extractors. Moreover, the more sophisticated recursive (RNNs) (Hochreiter and Schmidhuber, 1997; Gers and Schmidhuber, 2001) and convolutional (CNNs) (LeCun and Bengio, 1998; Kim, 2014) neural networks, as well as TRANSFORMERs (Vaswani et al., 2017), better capture natural language. Recently, all these variants have extended their analytical and processing capacity to better capture language syntax and semantics; relying on gating and attention mechanisms (Bahdanau et al., 2015; Vaswani et al., 2017), while also exploiting massive pre-training considering language modeling and sentence-ordering classification tasks (Devlin et al., 2019; Liu et al., 2019b; Raffel et al., 2020). These models usually perform close or better to human performance in several downstream NLP tasks on generic benchmark datasets, such as GLUE (Wang et al., 2018),

SUPER-GLUE (Wang et al., 2019), and SQUAD (Rajpurkar et al., 2016).

DNNs have also been gradually introduced in the legal domain. Traditionally, researchers employed manually crafted knowledge bases and patterns to capture legal concepts, terms of interest, and synonyms that were defined beforehand. Previous work relied on the structure of the legal documents to be able to segment and process text. Nevertheless, the presumed structure was not consistent across different laws and legislation. Furthermore, the legal concepts evolve and the maintenance of legal knowledge bases is tedious and expensive. Also, ontological representations that model legal knowledge as a whole are too generic and often lack details to be useful in practice. Similarly, taxonomies crafted to fit a particular task may become outdated rapidly.

Early work employing DNNs in legal question answering using CNNs (Kim, 2014) and obligation extraction using RNNs (O'Neill et al., 2017) seems primitive, comparing to today standards, but also quite hasty with respect to studying domain peculiarities (how legal language, writing and structure affect performance). While the examined DNNs outperform traditional methods, they do not closely follow the state-of-the-art in the NLP literature. More importantly, in-domain pre-training was ignored alongside proper modeling of document structure. In this work, we aim to meet and surpass the most up-to-date standards in deep learning literature, extensively compare neural methods alongside non-neural methods, and most importantly study and highlight how domain peculiarities affect the performance of the examined models.

1.2 Contributions

The contributions of this thesis can be summarized as follows:

New legal NLP datasets: We compile and publish five new annotated datasets for various legal tasks (contract element extraction, legislation classification, legal judgment prediction, and legal information retrieval) for academic research.

Domain-specific word embeddings and pretrained models: We introduce new domain-specific English word embeddings trained on legal corpora that better capture legal language, compared to generic word embeddings (Pennington et al., 2014; Bojanowski et al., 2016) trained on data derived from the web. Furthermore, given the excessive use of numbers in the legal text, we use number normalization, i.e., all digits are considered to be the same, which is important to generalize across similar numeric-based entities, i.e., dates, amounts, postcodes, etc., and avoid out-of-vocabulary (OOV) issues. Following the most recent advances in NLP, we also introduce a family of English BERT language models (Devlin et al., 2019) pre-trained on legal corpora.

In the case of contract element extraction (Chapter 2), we find that domain-specific word embeddings and in-domain pre-training of neural language models improves the performance in two out of three entity groups. The positive effect of domain adaptation is minimal in the rest of the tasks, which consider EU legislation (Chapter 3) and ECtHR cases (Chapter 4); with the exception of multi-label classification on ECtHR.

LSTMs beat BERT in contract element extraction: Considering the challenging task of contract element extraction, we show that LSTM-based methods (Graves et al., 2013; Huang et al., 2015) outperform pre-trained TRANSFORMER-based models (Devlin et al., 2019) that currently dominate in NLP, even compared with the legal BERT model (Chapter 2). This finding is mostly correlated with the use of context-sensitive entities and to a lesser degree with occasional severe fragmentation of tokens by TRANSFORMER-based models.

Hierarchical LSTM and BERT models to cope with document structure: To successfully tackle the task of obligation extraction from contracts (Chapter 2), we propose a slightly modified version of the Hierarchical Attention Network (Yang et al., 2016) for sentence tagging (classification) to capture inter-sentence relations and relations between sentences and listed clauses in a document. The new method has state-of-the-art performance, out-performing flat methods that do not consider neighboring sentences or consider them in a non-hierarchical manner. We have similar findings for legal judgment predictions, where we employ the Hierarchical Attention Network (HAN) to capture the document structure of ECHR cases (Chapter 4). Furthermore, we propose a hierarchical variation of BERT (HIER-BERT). While both methods have been recently used by several researchers in various classification tasks, we were among the first who proposed and used hierarchical neural networks, especially for sentence tagging.

New state-of-the-art methods for legal topic classification: With respect to legal topic classification (Chapter 3), a Large-Scale Multi-label Text Classification (LMTC) application, we study the effect of using RNN-based, PLT-based (Prabhu et al., 2018) and TRANSFORMER-based (Vaswani et al., 2017) methods. We show that RNN-based baselines (BIGRU-ATT, HAN) outperform current state-of-the-art Label-Wise Attention Networks (LWANs) (Mullenbach et al., 2018) that rely on CNN encoders (CNN-LWAN), while we propose a new BIGRU-based LWAN which is even better. Moreover, we show that newly introduced pre-trained TRANSFORMER-based methods (BERT, ROBERTA) further improve results, without relying on the label-wise attention mechanism, thus we propose a new state-of-the-art method that combines BERT with LWAN (BERT-LWAN) achieving the best results overall. In a more general perspective, we validate our findings in other non-legal LMTC benchmark datasets (AMAZON13K, MIMIC-III). We have similar findings for

AMAZON13K, while we interestingly show that TRANSFORMER-based methods underperform in MIMIC-III and study the possible reasons for this result.

Moreover, few-shot and zero-shot learning are vastly understudied in Large-scale Multi-label Text Classification (LMTC). Following the work of Rios and Kavuluru (2018) for few and zero-shot learning in the biomedical domain, we investigate the use of structural information from the label hierarchy in LWAN. We propose new zero-shot capable LWAN-based models with improved performance in these settings.

New task and model for explainability in legal judgment prediction: We study explainability in the legal domain and find the use of saliency (attention) maps problematic in legal judgment prediction (Chapter 4). To this end, we introduce a new legal prediction task accompanied by the extraction of paragraph-level rationales. While we adopt and compare various rationale constraints (regularizers) from the literature in the new paragraph-level setting, we also propose new constraints (regularizers) that produce state-of-the-art results by improving rationale quality, while also improving faithfulness.

New task, dataset, and models for legal document-to-document IR: We introduce regulatory information retrieval, an application of *document to document* IR, which is a new family of IR tasks, where both queries and documents are long, typically containing thousands of words (Chapter 5). We show that fine-tuning BERT on an in-domain classification task produces the best document representations with respect to IR and improves pre-fetching results compared to various methods including the traditional BM_{25} and generic BERT models (BERT-BASE of Devlin et al. (2019), S-BERT of Reimers and Gurevych (2019)).

1.3 Outline of the remainder of this thesis

The remainder of this thesis is organized as follows: Chapter 2 presents our research on information extraction from legal documents, specifically contracts, targeting two applications (contract element extraction and obligation extraction); Chapter 3 presents our research on Large-scale Multi-label Text Classification (LMTC) for legal documents targeting the application of classifying EU legislation with EUROVOC concepts; Chapter 4 presents our research on Legal Judgment Prediction and Explainability for cases of the European Court of Human Rights (ECtHR); Chapter 5 presents our research in Legal Document to Document Information Retrieval targeting the application of Regulatory Compliance between UK and EU legislation; and lastly Chapter 6 concludes, highlights limitations and proposes directions for future research.

Chapter 2

Information Extraction for legal documents

2.1 Introduction

We focus on two legal information extraction tasks for contracts. The first application is *contract element extraction*, an entity extraction task, where we aim to extract contract elements that are common and particularly useful in legal text analytics applications. The second application is *obligation extraction*, where we aim to identify obligations and prohibitions, two major types of agreed terms between contract parties. The automation of these NLP applications, among others could benefit two major use cases:

Due Diligence/Contract Negotiations: Thousands of agreements are signed every day between businesses and individuals. There is an excessive need for checks by all interested parties during negotiations in order to control the terms and the obligations that will govern the implementation of each agreement, where each party comes with its own playbook (agenda) to protect their interests.

Contract Management/Vendor Management: Most established companies need to maintain a large portfolio of legal agreements for various business processes (employment, services/vendors, loans, investments, confidentiality). All of them include details on different contract terms, pay rates, rights for termination, etc. Using intelligent services in order to organize and review these agreements is vital in order to prevent several issues such as missing payments, revenue leakage, invalid contracts, etc.

So far, there is not much work for information extraction from legal documents. There are three main limitations in the relevant literature described in Section 2.2: (a) tasks are usually being mishandled or oversimplified; (b) there are no publicly available datasets and the ones used often contain a small number of instances (e.g., less than 1,000); (c)

the methods used are usually out-dated, limited to lookup tables, rule-based systems and linear classifiers (SVMs) with bag-of-word representations and hand-crafted features.

In our work, we aim to develop and evaluate neural methods following the relevant NLP literature, while also considering some crucial in-domain aspects. Legal documents, like contracts in our study, are usually very long, comprising thousands of words and structured (e.g., with sections, bullet/numbered lists), which had not been considered before in information extraction. We show that modeling structure is particularly important in one of the two tasks that we study. Both characteristics (excessive length and structure of the documents) are particularly problematic for current neural methods.

Another important aspect, which we face in one out of the two applications, is the amount of domain-specific tags (labels), which is also not common in information extraction. Moreover, depending on their types, entities (meaning entity types) occur in different particular sections (or other zones) of the documents, where they may co-occur with entities of particular other types, contrary to the entity types used in generic NER (e.g., persons, organizations, locations), which can be present across documents. The entity types are also more refined, including many similar ones, e.g., types of dates/locations/organizations, instead of universal types describing broader concepts. In this case, broader context is more crucial to determine the correct entity type, for example, to discriminate one type of date from another, as the entity text itself and local context may possibly not provide the necessary information.

2.2 Related Work

Curtotti and Mccreath (2010) classified lines (separated by line breaks) of Australian contracts into 32 classes (e.g., 'titleline', 'clausehead', 'recitalhead', 'recitalline', 'partyline', 'datemadeline', 'contactofficer', 'emailline', etc.). Although many of the aforementioned classes (elements) correspond to entities (multi-word expressions), Curtotti and Mccreath (2010) addressed the task as text (line) classification instead of sequence labeling. The authors experimented with several machine learning algorithms (e.g., SVMs, decision trees), using 40 hand-crafted features. They obtained their best results (83.48% accuracy) with a single multi-class classifier that combined machine learning (Random Forest) and manually written tagging rules (which provided additional features to the Random Forest). They experimented, however, with only 30 contracts from a corpus of 256.

Indukuri and Krishna (2010) employed SVMs and n-gram features to classify contract sentences as clauses or non-clauses, and classify clauses as payment terms or not, experimenting with only 73 sentences. Gao et al. (2012) used 2,647 contracts, but experimented only with manually crafted patterns to detect exception clauses (e.g., "in case of defect").

In the broader context of legal text analytics, Mencia (2009) used SVMs and hand-crafted features to segment French laws (e.g., identify titles, articles), experimenting with 181 texts (1,146 articles). Hasan et al. (2008) relied on heuristics to segment Spanish legislative bulletins into their components (e.g., articles), assuming that each bulletin includes a table of contents, and experimenting with 50 texts. Biagioli et al. (2005) used an SVM with bag-of-words features to detect paragraphs of Italian laws with particular types of information (e.g., obligation, sanction). Then they employed pattern matching to fill in type-specific slots (e.g., entity sanctioned), experimenting with 582 paragraphs.

Dozier et al. (2010) identified judges, attorneys, companies, jurisdictions, and courts in US trial documents. A Conditional Random Field (CRF) (Lafferty et al., 2001) with n-gram, positional, and punctuation features was used to segment each document into zones. Then lists of known entities (e.g., courts) and hand-crafted patterns were used to extract named entities from particular zones. Manually constructed rules were employed to map each extracted entity to a record (e.g., containing fields for the first name and surname of an extracted attorney name, along with city names that occurred near the attorney name) and retrieve candidate matching records from authority files (e.g., records of known attorneys). SVMs with field-specific similarity measures as features were subsequently used to select the 'best' authority file record per extracted named entity record.

Kiyavitskaya et al. (2008) used grammars, word lists, and heuristics to extract rights, obligations, exceptions, and other constraints from US and Italian regulations. Asooja et al. (2015) employed SVMs with n-gram and manually crafted features to extract paragraphs of money laundering regulations into five classes (e.g., enforcement, monitoring, reporting), experimenting with 212 paragraphs.

More recently, O'Neill et al. (2017) extracted contractual terms from financial legislation, classified in three categories (obligations, prohibitions, and permissions), experimenting with 2,000 pre-splitted sentences. Similarly, Waltl et al. (2017) classified statements from German tenancy law into 22 classes (including prohibition, permission, consequence), using active learning with Naive Bayes, Logistic Regression (LR), and MLP classifiers, experimenting with 504 sentences.

2.3 Contributions

- We compile two new annotated datasets extracted from contracts including thousands of instances, for two legal information extraction tasks: Contract Element Extraction and Obligation Extraction. While both datasets adhere to IPR, we publish the contract element extraction dataset for academic research in an encoded form, i.e., replace words with identifiers, to preserve the interests of the owner.

- We show that in contract element extraction, LSTM-based methods (Graves et al., 2013; Huang et al., 2015) outperform pre-trained TRANSFORMER-based models (Devlin et al., 2019) that currently dominate in NLP, even compared with a new TRANSFORMER-based model pre-trained on legal corpora. This finding is mostly correlated with the use of context-sensitive entities and to a lesser degree with occasional severe fragmentation of tokens by TRANSFORMER-based models.

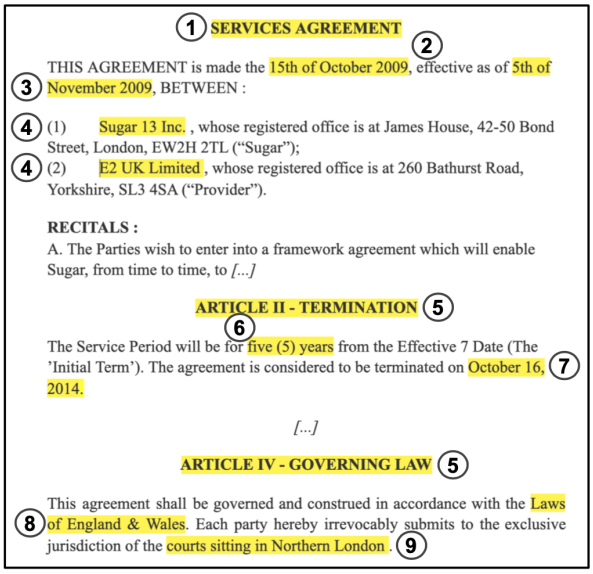
- We propose a slightly modified version of the Hierarchical Attention Network (Yang et al., 2016) for sentence tagging (classification) to capture inter-sentence relations and relations between sentences and listed clauses in a document. The new method has state-of-the-art performance, outperforming flat methods that do not consider neighboring sentences or consider them in a non-hierarchical manner. The new method has state-of-the-art performance in obligation extraction, out-performing flat methods that do not consider or consider neighboring sentences in a non-hierarchical (structured) manner. While this method has been recently used by several researchers in various classification tasks, we were among the first who proposed and used hierarchical neural networks, especially for sentence tagging.

2.4 Contract Element Extraction

Extracting information from contracts and other legal agreements is an important part of daily business worldwide. Thousands of agreements are written up every day, resulting in a huge volume of legal documents relating to several business processes, such as employment, services/vendors, loans, leases, investments, to name a few. These documents contain crucial information (e.g., contract terms, pay rates, termination rights). More importantly, when negotiating or revising agreements, the parties involved need to scrutinize all the terms of the agreements as recorded in the corresponding documents. From another point of view, taxation authorities may need to focus on contracts involving particular parties and large payments. Many of these tasks can be automated by extracting particular contract elements (e.g., termination dates, legislation references, contracting parties, agreed payments). Contract element extraction, however, is currently performed mostly manually, which is tedious and costly.

2.4.1 Contract Structure and Elements

Contracts typically start with a *preamble*, which contains the contract title (Figure 2.1a, point 1) and specifies the start or effective date (points 2, 3) and contracting parties (points 4). It is also common to use a *cover page* with the same information followed



1 SERVICES AGREEMENT 2

THIS AGREEMENT is made the 15th of October 2009, effective as of 5th of November 2009, BETWEEN : 3

4 (1) Sugar 13 Inc. , whose registered office is at James House, 42-50 Bond Street, London, EW2H 2TL ("Sugar");

4 (2) E2 UK Limited , whose registered office is at 260 Bathurst Road, Yorkshire, SL3 4SA ("Provider").

RECITALS :

A. The Parties wish to enter into a framework agreement which will enable Sugar, from time to time, to *[...]*

ARTICLE II - TERMINATION 5

6

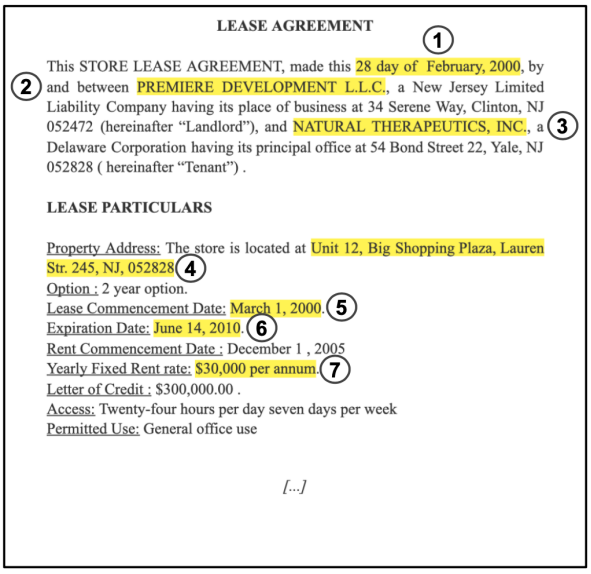
The Service Period will be for five (5) years from the Effective 7 Date (The 'Initial Term'). The agreement is considered to be terminated on October 16, 7 2014.

[...]

ARTICLE IV - GOVERNING LAW 5

8 This agreement shall be governed and construed in accordance with the Laws of England & Wales. Each party hereby irrevocably submits to the exclusive jurisdiction of the courts sitting in Northern London . 9

(a) Typical structure of a contract, with contract elements highlighted.



LEASE AGREEMENT 1

2 This STORE LEASE AGREEMENT, made this 28 day of February, 2000, by and between PREMIERE DEVELOPMENT L.L.C., a New Jersey Limited Liability Company having its place of business at 34 Serene Way, Clinton, NJ 052472 (hereinafter "Landlord"), and NATURAL THERAPEUTICS, INC., a 3 Delaware Corporation having its principal office at 54 Bond Street 22, Yale, NJ 052828 (hereinafter "Tenant") .

LEASE PARTICULARS

Property Address: The store is located at Unit 12, Big Shopping Plaza, Lauren Str. 245, NJ, 052828 4

Option : 2 year option.

Lease Commencement Date: March 1, 2000. 5

Expiration Date: June 14, 2010. 6

Rent Commencement Date : December 1 , 2005

Yearly Fixed Rent rate: $30,000 per annum. 7

Letter of Credit : $300,000.00 .

Access: Twenty-four hours per day seven days per week

Permitted Use: General office use

[...]

(b) Typical structure of a lease header, with contract elements highlighted.

Figure 2.1: Contract samples annotated with contract elements.

by a *table of contents*, before the preamble. The preamble is usually followed by the *recitals* (Figure 2.1a), which provide background information. The remainder of the contract is organized in *clauses*, often called 'chapters', 'articles', 'sections' etc. to reflect a hierarchical structure. Clauses have headings (Figure 2.1a, points 5), which indicate their topics (e.g., 'Definitions', 'Termination', 'Payments'). In this paper, we focus on extracting the following types of *contract elements*, when present. More contract element types can be defined, but the types we focus on are common and particularly useful in analytics applications like the ones highlighted in Section 2.1.

Contract Header: The first set of contract elements are present in the contract header, i.e., the cover page and/or introduction.

- **Contract Title** (Figure 2.1a, point 1). The title usually indicates the type of the contract (e.g., services, employment, loan) and often also the version of the contract (e.g., 'second amendment').

- **Contracting Parties** (Figure 2.1a, points 4). *Contract Parties* are all the legal entities (persons or organizations) that are engaged in a contract, i.e., service provider and client, employer and employee, lender and borrower, landlord and tenant, etc.

- **Start, Effective Dates:** The *start date* (Figure 2.1a, point 2) is when the contract was signed. The *effective date* (points 3) specifies when the contract becomes effective. In many cases, the contract is immediately effective after signing, which

means that the start and effective date are the same, thus effective date is usually not referred in these cases. Contrary, if the effective date differs, it is usually referred alongside the start date in the cover and/or introduction.

Contract Termination: The second set of contract elements is relevant to the termination of the contract. The *termination date* (Figure 2.1a, points 7) specify when the contract terminates, while in other cases the contract specifies the *contract period* (point 6), which is the number of working or calendar days the contract will be effective for. One out of two is referred to in the contract, as both convey the same information, e.g., one can compute the contract period, if he/she is aware of both the effective and termination date.

Applicable Law: The third set of contract elements is relevant to the judicial aspect of the contracts. The *governing law* (Figure 2.1a, point 8) specifies the country or state whose laws apply. The *jurisdiction* (point 9) specifies the courts responsible to resolve disputes. Both elements are particularly important as the legislation and case law are different across countries or even states/provinces in the same country.

Lease Header: The last set of contract elements is present in the header of lease agreements, i.e., the cover page and/or introduction and/or specific sections named as 'Lease Details' or 'Lease Header' (Figure 2.1b), and contain the following information in free text or tabular format:

- **Property**. *Property* is the location of the leased property (e.g., building, apartment, store, etc.) (Figure 2.1b, point 4). This is usually described with the formal address (floor, street number, city, postcode) of the location.

- **Landlord and Tenant**. *Landlord* (point 2) is the contracting party who owns the leased property, while the *tenant* (point 3) is the one who rents (leases) this property. Those can be either persons or organizations.

- **Rent amount**. *Rent amount* (point 7) is the monetary amount provided by the tenant to the landlord, usually monthly or yearly, that both parties agreed upon.

- **Start, Effective, Termination Dates, and Period**. These contract elements are usually referred to in 'Lease Header' sections and have the same meaning, as described above.

Inferring Contract Structure from Headings: *Clause Headings* (Figure 2.1a, points 5) are needed to automatically build a table of contents (for contracts that do not include one) and split the text of each contract after the recitals into clauses (from clause heading to clause heading). We also note that most clause headings contain typical words or phrases that indicate their topics. For example, many contracts include a clause whose heading contains phrases like 'Governing Law' or 'Applicable Law'; the governing law contract element (e.g., 'Laws of England & Wales' in Figure 2.1a, point 8) is then found in that clause. Similarly, many contracts include a clause whose heading contains words or phrases like 'Termination' or 'Termination of Agreement'; the termination date (e.g., 'October 16, 2014' in Figure 2.1a, point 7) is then found in that clause. Alternatively, other contracts include a more general clause whose heading contains words or phrases like 'Term', 'Period', 'Term of Agreement'; clauses of this kind typically include the termination date and/or the contract period. Hence, identifying clause headings (and then looking for indicative words or phrases in the headings) is also a first step towards identifying clauses where other contract elements are expected to be found.

2.4.2 Relation to Named Entity Recognition

Generic named entity recognizers (NERs) (Bikel et al., 1999; Nadeau and Sekine, 2007), which typically recognize persons, organizations, locations, dates, amounts, etc., are not directly applicable to contract element extraction without retraining them on contracts and possibly modifying their feature sets.[1] For example, a generic NER may recognize dates, but without distinguishing between start, effective, termination, and other dates (e.g., payment or delivery dates, which we do not aim to extract). Note that several of these date types may occur in the same extraction zones; for example, start and effective dates typically occur both on the cover page and in the preamble. Hence, the date types we aim to extract cannot be distinguished simply by observing the extraction zones they occur in. Similarly, a generic NER may recognize persons and organizations, but not all persons and organizations mentioned in a contract are contracting parties; for example, a law firm that prepared the contract or a third-party service provider may be mentioned (sometimes in the same extraction zones as the contracting parties), without being contracting parties. Moreover, in the case of lease agreements, we want to discriminate contract parties in landlords and tenants. Similar comments apply to the property, governing law, and jurisdiction elements, which are not simply locations. Furthermore, contract titles and clause headings are not supported by generic NERs.

[1] Consider, for example, the NERs of Stanford University (`http://nlp.stanford.edu/software/CRF-NER.shtml`) and spaCy (`http://spacy.io/docs/usage/entity-recognition`).

Contract Element Type	Instances	Tokens	Contract Element Type	Instances	Tokens
Contract Header			Lease Header		
Contract Title	4,486	17,324	Property	2,552	19,271
Contracting Parties	8,030	35,413	Landlord	2,828	12,287
Start Date	2,503	10,950	Tenant	2,629	11,778
Effective Date	679	2,993	Start Date	1,804	8,693
Contract Termination			Effective Date	1,219	4,991
Termination Date	534	2,140	Termination Date	1,037	4,054
Contract Period	421	1,628	Contract Period	1,065	5,103
Applicable Law			Rent Amount	2,233	4,570
Governing Law	2,369	14,561	Contract Structure		
Jurisdiction	1,474	11,627	Clause Headings	38,269	∼183K

Table 2.1: Statistics of the US-CONTRACTS-NER labeled dataset.

2.4.3 Dataset

The benchmark datasets are produced by approximately 1,000 contracts annotated with clause headings (Contract Structure), 2,500 contracts annotated with contract elements of Contract Header, Contract Termination and Applicable Law and 2,000 contracts (lease agreements) annotated with contract elements in Lease Header. Table 2.1 shows the number of contract elements (instances) and tokens per contract element type in the labeled datasets; a contract element may consist of multiple tokens, which is why we also report the overall number of tokens.

The gold contract element annotations of the benchmark datasets were provided by 10 law students following specific guidelines curated in collaboration with legal experts. Each contract was annotated by one student. Before the final annotation, however, we used three rounds of preliminary experiments and two pairs of annotators (the same pairs in all rounds) to measure the inter-annotator agreement and improve the annotation guidelines. The average agreement reached 0.80 in the third round as the guidelines were improved. All the contracts of the benchmark datasets are in English.[2]

The *test* contracts of the benchmark datasets include gold (correct) *extraction zones* (contract sections) per contract element type (Contract Header, Contract Termination and Applicable Law and Lease Header), as already discussed (Section 2.4.1). For each test contract and each contract element type (e.g., contracting

[2]All datasets are publicly available at `http://nlp.cs.aueb.gr/software_and_datasets/CONTRACTS_ICAIL2017/index.html`. We cannot reveal the actual texts, due to privacy and IPR issues, but we provide them in an encoded form, where each vocabulary word has been replaced by a unique integer (e.g, 'agreement' is mapped to '[TOKEN_146]'). We also provide hand-crafted features, word embeddings, POS tag, and token shape embeddings per token, further discussed below.

parties), the tokens (word occurrences) of the corresponding extraction zones that are parts of contract elements of that type (e.g., the tokens of contracting parties, as indicated by the gold contract element annotations) are treated as *positive* test instances (e.g., tokens that should be classified as contracting parties), whereas the other tokens of the extraction zones are treated as *negative* test instances (e.g., tokens that should not be classified as contracting parties). To reduce the manual annotation effort that was required to produce the labeled dataset, the *training* contracts of the dataset do not contain extraction zones. Instead, we produce automatically generated *pseudo-extraction zones* of the training contracts. For each contract element type (e.g., contracting parties), the pseudo-extraction zones of a training contract contain the tokens (word occurrences) of the contract elements of that type (e.g., the tokens of the contracting parties, as indicated by the gold contract element annotations of the training contract) and up to 50 tokens before and after each contract element of that type in the training contract (e.g., 50 tokens before and after each contracting party).

Our assumption is that these pseudo-extraction zones used only during training time, approximate the gold extraction zones, i.e., a paragraph including co-occurring contract elements. At the same time, they reduce the class imbalance of negative vs. positive tokens during training. In preliminary experiments, we trained contract extractors considering the full text of contracts. Given the huge class imbalance in favor of negative tokens, i.e., thousands of words except a few positive instances, extractors had terrible performance. Especially for clause headings, the pseudo-extraction zones are entirely located in the text after the recitals of each contract, with each zone starting up to 20 tokens before and ending up to 20 tokens after a line break ('\n').

2.4.4 Methods

Word, Part-of-Speech tag and Shape Embeddings: We applied WORD2VEC (skip-gram model) (Mikolov et al., 2013b) to approximately 750,000 English contracts (approx. 9 billion tokens). We produced 200-dimensional *word embeddings*, one for each vocabulary word.[3] The vocabulary size of WORD2VEC was approximately 500,000. Out of vocabulary words are mapped to random embeddings. To generalize across numbers with similar patterns and tokens that differ only in the use of upper and lower case, the training corpora were pre-processed to lower-case its tokens and replace all digits by 'D'. For example, 'Agreement' became 'agreement', 'October 16, 2014' became 'october DD, DDDD', and 'CO2' became 'coD', respectively.

[3]We used Gensim's implementation of WORD2VEC (`http://radimrehurek.com/gensim/`), with 10 minimum occurrences per word and default values for other parameters.

We also use 25-dimensional *Part-Of-Speech (POS) tag embeddings*, which were obtained by applying WORD2VEC (again, skip-gram model, same other settings) to approximately 50,0000 contracts, after replacing the words by their POS tags, again as predicted by a generic POS tagger. We empirically found that SPACY's POS tagger provided better results than NLTK's.[4] We use fewer dimensions in the POS tag embeddings compared to the word embeddings (25 instead of 200), because the POS tag embeddings need to represent only 45 points (POS tags) in their vector space, whereas the word embeddings need to represent the entire vocabulary.

We also use 25-dimensional *token shape embeddings* reflecting the corresponding token shape. We built a vocabulary of the 5,000 most frequent token shapes by replacing all uppercase letters with 'C', lowercase letters with 'c', digits with 'd' and retained all punctuation marks, i.e., 'Google' became 'Ccccc', 'P2P' became 'CdC', and '2,500' became 'd,ddd'. Again, we obtained the token shape embeddings by applying WORD2VEC.

Hand-Crafted Features: In early work, we also used hand-crafted features with sliding window linear classifiers. The values of these features are automatically computed; by 'hand-crafted' we mean that the particular feature sets and the meaning of each feature (what each feature stands for) were chosen by ourselves; by contrast, the components of word embedding vectors cannot be directly mapped to human-interpretable concepts. The first 14 hand-crafted features are the same regardless of the type of contract element being detected: 4 binary features for all upper, all lower, mixed case tokens, tokens containing numbers; 7 binary features indicating the length of the token (the first feature is true if the length of the token is 1-2 characters, the second feature is true if the length is 3-4 characters, and similarly for lengths 5-6, 7-8, 9-10, 11-12, >12); 3 binary features indicating if the token is numeric, a special character, or stop-word. The rest of hand-crafted features are also binary, but differ per contract element type. They indicate if the token is common inside or near elements of the particular type, or if it is matched by regular expressions that detect frequent parts of elements of the particular type (e.g., '1.1', '1.2.1' for clause headings, 'inc.', 'corp.' for contracting parties '2009', '13th' for dates, 'laws' and 'courts' for governing law and jurisdiction, etc.).

Methods

Sliding window SVMs classifier: Our first method (SW-SVMs) uses a separate linear SVM classifier (Vapnik, 1995; Cristianini and Shawe-Taylor, 2000) per contract element type (11 classifiers).[5] For each token t being classified, each classifier considers a slid-

[4] Available at `https://spacy.io` and `https://www.nltk.org`

[5] We employ the SCIKIT-LEARN implementation of SVM (`http://scikit-learn.org/`).

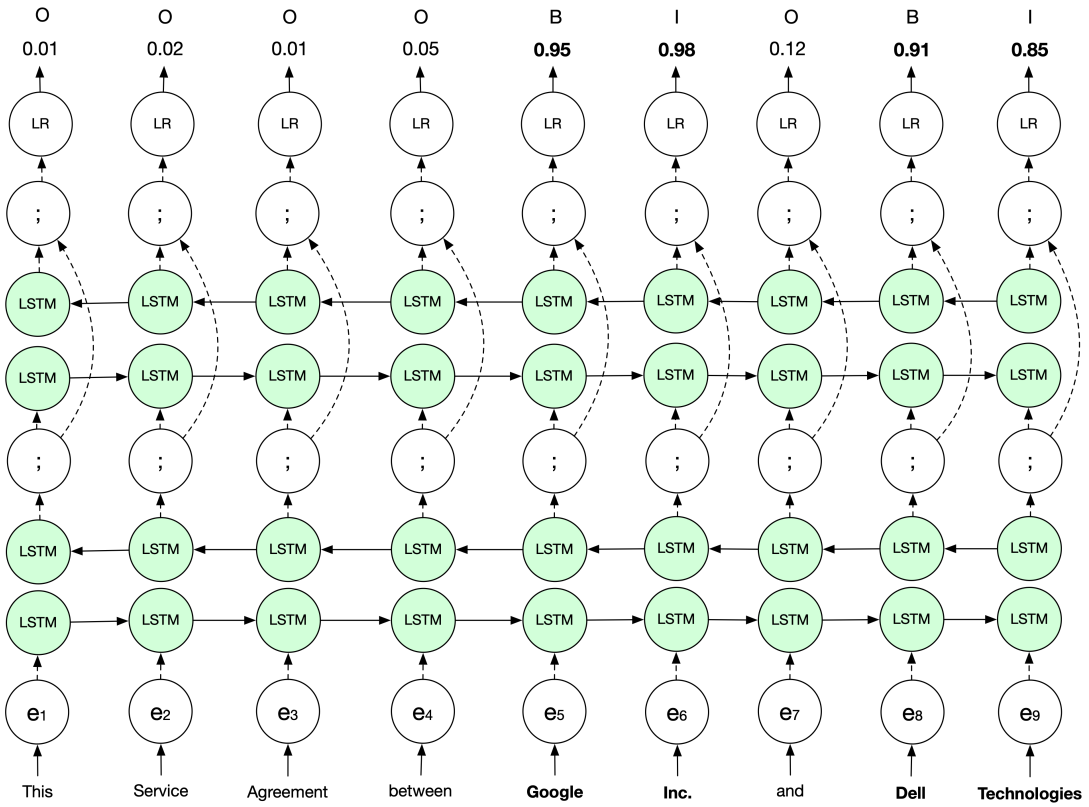


Figure 2.2: BILSTMs extractor for a particular contract element type.

ing window of 5-6 tokens around t (11-13 tokens); the exact size of the window varies, depending on the type of contract elements that each classifier extracts. The window is turned into a feature vector containing the concatenated word embeddings, POS tag embeddings, and hand-crafted features of all the tokens in the sliding window (e.g., 11 tokens $\times$ (200 + 25 + 17) = 2,662 features). We call this method (SW-SVM-ALL). In early experiments, we found that using either the pre-trained embeddings or hand-crafted features harms performance in development data. We also experimented with a similar method with Logistic Regression (LR) instead of SVMs, with comparable results. We omit the results for these experiments (partial feature set and use of LR) for brevity.

BILSTMs: In the first neural method we consider (Figure 2.2), we use stacked bidirectional LSTMs (Graves et al., 2013) to convert the concatenated word, POS tag, and token shape embeddings of each token to context-aware token embeddings, as follows:

$$\overrightarrow{h}_t = \text{LSTM}\left(\overrightarrow{h}_{t-1}, e_t\right) \tag{2.1}$$

$$\overleftarrow{h}_t = \text{LSTM}\left(\overleftarrow{h}_{t+1}, e_t\right) \tag{2.2}$$

$$\overleftrightarrow{h_t} = [\overrightarrow{h}_t; \overleftarrow{h}_t] \tag{2.3}$$

where e_t is the concatenated word, POS tag, and token shape embeddings and $\overrightarrow{h}_t$, $\overleftarrow{h}_t$

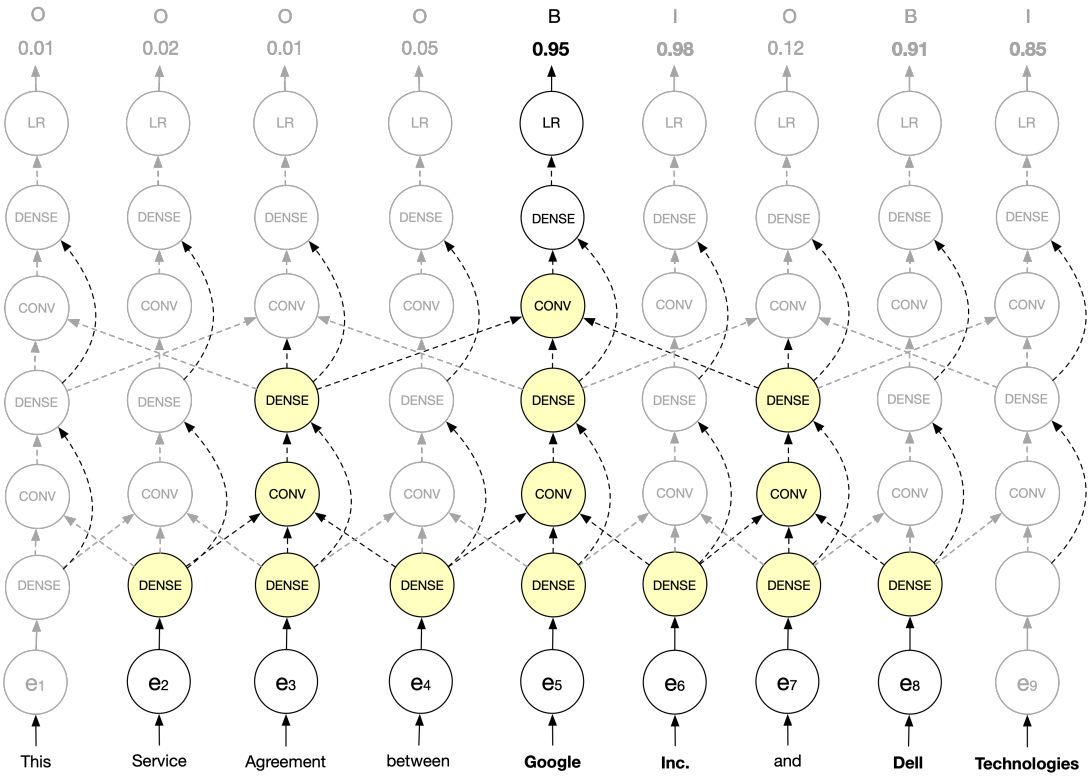


Figure 2.3: DILATED-CNNs extractor for a particular contract element type.

are the context-aware embeddings given left and right context, respectively. $\overleftrightarrow{h_t}$ is the concatenation of both $\overrightarrow{h}_t$ and $\overleftarrow{h}_t$, which may be forwarded to a second or more stacked BILSTMs. In order to avoid vanishing gradients, we use skip-connections, also known as residual connections (He et al., 2016), between each stacked BILSTM layer:

$$h_t = \left(W_h \cdot \overleftrightarrow{h_t}\right) + e_t \tag{2.4}$$

As vectors e_t and $\overleftrightarrow{h_t}$ may have different dimensionality ($D_h \neq D_e$) we have to transform $\overleftrightarrow{h_t}$ using a trainable matrix $W_h \in \mathbb{R}^{D_h \times D_e}$ before we add it to e_t. The final context-aware token embedding (h_t) is then passed on to a Logistic Regression (LR) layer followed by the softmax (or sigmoid) function to estimate the probability that the corresponding token is positive or negative with respect to the contract element type of the particular extractor.

DILATED-CNNs: In the second neural method (Figure 2.3), we use DILATED-CNNs (Kalchbrenner et al., 2016) instead of BILSTMs to produce context-aware token embeddings. When applied to text, CNNs with only one convolution layer Kim (2014) contextualize each word by considering only local context (the text span considered during each convolution). When using CNNs with multiple stacked convolution layers, the receptive field (the text span each context-aware word representation depends on) increases at each

layer, and this helps CNNs capture longer-distance dependencies. DILATED-CNNs are also stacked CNNs, but incrementally increase the gaps between the words each convolution considers as we move to higher layers (Figure 2.3), to more quickly increase the receptive field to grasp broader relationships with fewer stacked convolution. More concretely, DILATED-CNNs use stacked convolutional blocks.

Each block (Equations 2.6-2.7) includes convolutions, denoted CONV below, that operate on the representations of three words (when using trigram convolution filters, as in this work), either word embeddings or word representations built by lower blocks. The gaps between the three words (the dilation rate) are increasingly larger as we move to higher stacked blocks (Figure 2.3):

$$e'_t = W_e \cdot \text{RELU}(e_t) \tag{2.5}$$

$$\overleftrightarrow{h_t} = \text{RELU}(\text{CONV}\left(e'_{t-d}, e'_t, e'_{t+d}\right)) \tag{2.6}$$

$$h_t = \overleftrightarrow{h_t} + e_t \tag{2.7}$$

Here e_t is the representation of the t-th token built by the immediately lower block (or embeddings in the case of the lower block). e'_t a projection of e_t pre-activated with RELU activation. In Equation 2.6, d denotes the dilation rate controlling which tokens will be considered as neighbors of e'_t at each level. $d = 2^{l-1}$ at the l-th stacked convolution block in the model. For example, in our case where we use windows of tri-grams, in the second and fourth block a token has been conditioned by 6 and 30 neighbor tokens, respectively. Similarly to BILSTMs, we use residual connections in between convolution blocks (Equation 2.7). Strubell et al. (2017) showed that DILATED-CNNs have comparable performance to BILSTMs in generic NER datasets (CONLL-2003 and ONTO-NOTES).

TRANSFORMERs - BERT: Given the rise of pre-trained TRANSFORMER-based (Vaswani et al., 2017) language models and their success in generic NER tasks among other NLP tasks, we also explore the use of BERT (Devlin et al., 2019) as an alternative encoder. TRANSFORMERs (Figure 2.4) is originally an encoder-decoder architecture, which ingests an input sequence of tokens and produces an output sequence conditioned on the input. The encoder, similarly to the previous methods, represents each token with an embedding e_t. Since the model contains no recurrence and no convolution, it uses positional embeddings (p_t), in order to capture the order of the tokens in the input sequence, i.e., $e'_t = e_t + p_t$ to form a position-aware token embedding (e'_t). In the original work of TRANSFORMERs (Vaswani et al., 2017), sinusoidal positional embeddings were used, computed

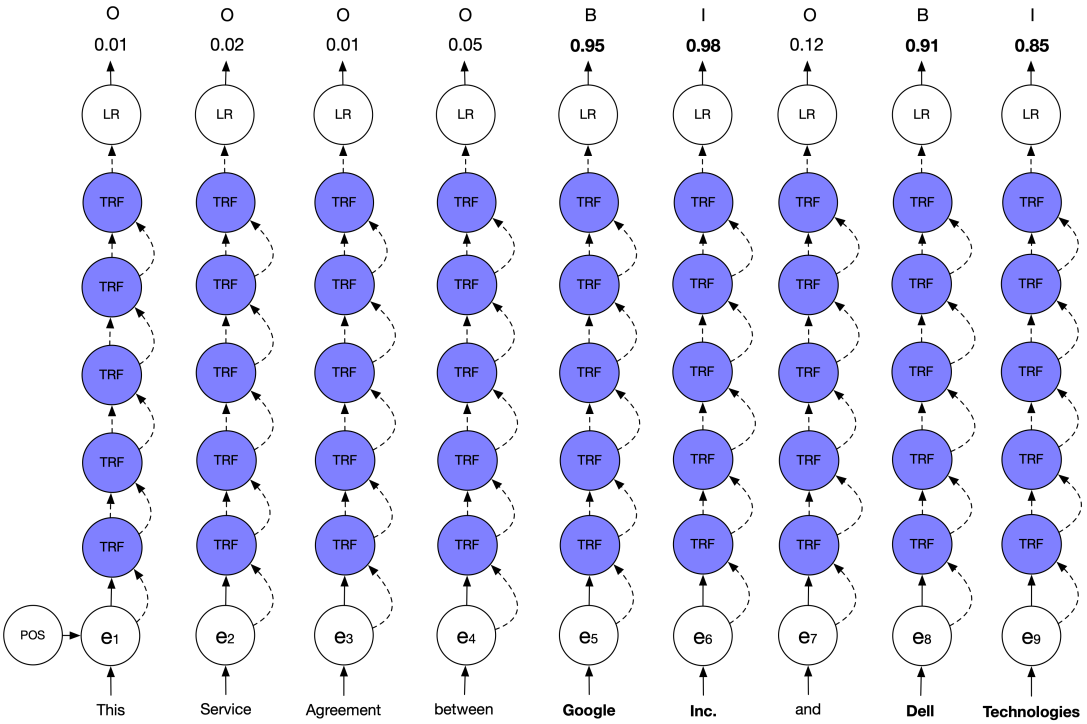


Figure 2.4: TRANSFORMERs extractor for a particular contract element type.

as follows:

$$p[t,i] = \begin{cases} \sin\left(\frac{i}{10000^{2i/d_{\text{model}}}}\right) & i \text{ even} \\ \cos\left(\frac{i}{10000^{2i/d_{\text{model}}}}\right) & i \text{ odd} \end{cases} \tag{2.8}$$

In later work (Devlin et al., 2019; Liu et al., 2019b; Raffel et al., 2020), the positional embeddings are casual trainable vectors, like any other set of embeddings. The position-aware token embeddings (e'_t) are forwarded to TRANSFORMER blocks. Each TRANSFORMER block comprises a multi-headed self-attention mechanism (Cheng et al., 2016) followed by a bottleneck feed-forward network. Both of these layers (multi-headed self-attention, feed-forward network) include a residual connection and layer normalization (Ba et al., 2016). For each head h_i in the multi-headed self-attention of the l_{th} TRANSFORMER block, we have the following for token t:

$$q_t^{h_i} = Q_{h_i} \cdot o_t^{(l-1)} \tag{2.9}$$

$$k_t^{h_i} = K_{h_i} \cdot o_t^{(l-1)} \tag{2.10}$$

$$v_t^{h_i} = V_{h_i} \cdot o_t^{(l-1)} \tag{2.11}$$

$$a_t^{h_i} = \text{softmax}\left(\frac{q_t^{h_i} \cdot {k_t^{h_i}}^\top}{\sqrt{d_k}}\right) v_t^{h_i} \tag{2.12}$$

where Q_{h_i}, K_{h_i}, V_{h_i}, are the "query", "key", and "value" projection matrices for head h_i, respectively. $o_t^{(l-1)}$ equals e'_t in the first TRANSFORMER block and represents the output of the previous $(l-1)$ TRANSFORMER block for the rest of the stacked blocks.

The self-attention outputs a_t^h for all H heads are then concatenated and projected with matrix O along with a residual connection and layer normalization as follows:

$$h_t = \text{LayerNorm}\left(\begin{bmatrix} a_t^1 \\ \vdots \\ a_t^H \end{bmatrix} O + e'_t\right) \tag{2.13}$$

The output of the multi-headed self-attention mechanism is then passed through a bottleneck feed-forward network that operates on each sequence element independently. Specifically, the feed-forward network consists of an up projection followed by a RELU activation, and another down projection as follows:

$$h'_t = W_{down} \cdot \text{RELU}(W_{up} \cdot h_t + b_{up}) + b_{down} \tag{2.14}$$

$$o_t = \text{LayerNorm}(h'_t + h_t) \tag{2.15}$$

The output h'_t, similarly to previous layers, passes through a residual connection followed by layer normalization. We omit the decoder, as it mainly targets sequence generation tasks and thus it not used in any experiment across the chapters of the thesis.

BERT (Devlin et al., 2019) is a TRANSFORMER-based encoder, pre-trained on large corpora, namely English Wikipedia and the Children Books Corpus (Hill et al., 2016) in order to acquire a general knowledge (syntax, semantics) of the English language. BERT is pre-trained in a multi-task setup that comprises two tasks: (a) mask language modelling (cloze task), where random tokens from the input sequence have been masked and the model predicts the missing tokens; and (b) next sentence prediction, where the model predicts if the input sequence provided comprises two originally sequential sentences or not (random sentences). BERT uses sub-word token embedding and trainable positional embeddings that both are tuned in the pre-training phase, as described.

Similarly to previous work, we use the token representation from the final BERT layer and add a Logistic Regression (LR) layer on top. Devlin et al. (2019) showed that BERT outperforms former state-of-the-art BILSTM-based methods in the aforementioned generic named entity recognition datasets. Additionally to the original BERT model, we experiment with a non-pre-trained TRANSFORMER-based model using the concatenated word, POS tag, token shape, and sinusoidal positional embeddings (Equation 2.8).

CRF-augmented methods: In all encoders, a dense layer with a softmax activation operates on the top-level representation of each token, providing a probability distribution over the labels, which is fed to the linear CRF layer. CRFs consider the global assignment of the sequence and provide categorical labels, instead of probabilities. The CRF-augmented models are named BILSTM-CRF, DILATED-CNNS-CRF, TRANSFORMERS-CRF and BERT-CRF, respectively.

Additional neural representations: We also experiment with models using ELMO (Peters et al., 2018) to obtain context-sensitive token embeddings, which we concatenate with the pre-trained word, POS tag, and token shape embeddings. ELMO is a pre-trained BILSTM-based language model. Further on, similarly to recent literature (Ma and Hovy, 2016; Peters et al., 2018; Akbik et al., 2019), we study the use of character-level encoders, in our case CNN-based ones. Character-level encoders consider tokens as sequences of characters represented by trainable character embeddings that are then passed through multi-filter CNNs, similar to those of Kim (2014). The character-level encoder finally applies max-pooling on top of the CNN context-aware character representations to produce a character-aware token embedding, which can be concatenated with the rest of the pre-trained embeddings (word, POS tag, token shape, and ELMO).

2.4.5 Experimental Setup

Across experiments, we used HYPEROPT and 5-fold Monte Carlo cross-validation on the training subset to tune hyper-parameters for all methods.[6] In the case of the sliding window SVM classifiers, we explore the following hyper-parameters on the training data with the following ranges: window size {5, 7, 9, 11, 13, 15} and regularizer C {0.01, 0.1, 1}. In the case of BILSTMs and DILATED-CNNs, we explore the following hyper-parameters on the training data with the following ranges: encoder output units {100, 150, 200, 250, 300}, encoder units per layer {1, 2, 3, 4}, batch size {8, 12, 16, 24, 32}, DROPOUT rate {0.2, 0.3, 0.4, 0.5, 0.6}, word DROPOUT rate {0.0, 0.05, 0.1}. We use the ADAM optimizer (Kingma and Ba, 2015) with initial learning rate 1e-3 and early stopping with patience for five epochs on validation loss. In the case of BERT, we grid-search for learning rate {2e-5, 3e-5, 4e-5, 5e-5}, as suggested by Devlin et al. (2019). All models were evaluated in precision, recall and F1-score per entity as defined in SemEval-2013 Task 9.1.[7]

2.4.6 Experiments

BILSTM-CRFs vs. sliding windows SVMs: In early experiments we compared a rule-based classifier (RULES), with SW-SVMs and BILSTM-CRF across all contract elements of CONTRACT HEADER, CONTRACT TERMINATION, APPLICABLE LAW, and CONTRACT STRUCTURE, which were initially available.[8] We trained a separate (binary) classifier for each contract element type. The results are presented in Table 2.2. On this batch of early experiments, scores are computed per entity with partial match.[9] As expected,

[6] We use the Tree of Parzen Estimators (TPE) algorithm (`http://hyperopt.github.io/hyperopt/`).

[7] For details on the evaluation rules, see `https://www.cs.york.ac.uk/semeval-2013/task9/data/uploads/semeval_2013-task-9_1-evaluation-metrics.pdf`

[8] LEASE HEADER dataset was annotated and used in a second phase of experiments.

[9] Partial match considers as true positives even extracted entities than partially match gold entities.

Contract Header									
	RULES			SW-SVMS-ALL			BILSTM-CRF		
	P	R	F1	P	R	F1	P	R	F1
Title	96.8	54.0	69.4	88.2	85.7	87.3	**97.3**	**94.7**	**95.3**
Party	96.1	39.2	55.7	89.7	81.6	86.2	**98.7**	**87.3**	**92.8**
S. Date	68.3	88.4	77.2	84.3	93.3	87.7	**92.6**	**96.3**	**93.7**
E. Date	84.1	94.7	88.6	44.2	53.2	48.4	**94.5**	**86.1**	**84.2**
MACRO-AVG	86.3	69.2	72.8	76.5	78.5	77.6	**95.6**	**91.2**	**91.5**
CONTRACT TERMINATION									
	RULES			SW-SVMS-ALL			BILSTM-CRF		
	P	R	F1	P	R	F1	P	R	F1
T. Date	66.3	97.6	79.2	49.1	85.3	61.8	**74.4**	**98.3**	**84.2**
Period	14.1	**82.2**	24.0	45.4	73.1	55.6	**62.3**	78.7	**70.5**
MACRO-AVG	40.2	74.3	51.6	47.3	79.2	58.7	**68.3**	**88.5**	**77.3**
Applicable Law									
	RULES			SW-SVMS-ALL			BILSTM-CRF		
	P	R	F1	P	R	F1	P	R	F1
Jurisdiction	**99.3**	59.4	74.2	65.3	84.2	74.4	93.4	**85.4**	**89.3**
Gov. Law	97.2	90.1	92.8	93.7	91.3	91.6	**99.2**	**96.2**	**97.2**
MACRO-AVG	**98.3**	74.7	83.5	79.5	87.7	83.0	96.3	**90.8**	**93.3**
Contract Structure									
	RULES			SW-SVMS-ALL			BILSTM-CRF		
	P	R	F1	P	R	F1	P	R	F1
Headings	79.8	86.2	83.1	60.3	70.9	64.9	**99.3**	**88.4**	**94.3**

Table 2.2: Precision (P), Recall (R), and F1-score across initially considered binary contract element extraction methods.

we find that BILSTM-CRFs outperform SW-SVM-ALL extractors across contract elements, usually by a large margin. In the rest of the experiments, we use a single (multi-class) classifier for all the contract elements that may reside in each of the following contract zones: Contract Header, Applicable Law, and Lease Header, which allows the classifier of each zone to generalize across the entity types.

Alternative neural encoders: Table 2.3 reports results with different sequence encoders, always followed by a CRF layer. With all encoders, a dense layer with a softmax activation operates on the top-level representation of each token, providing a probability distribution over the labels, which is fed to the CRF. In these experiments except for BERT-CRF, which uses its own sub-word embeddings, input representation of each token

	BILSTM-CRF			DILATED-CNNS-CRF			TRANSFORMERS-CRF			BERT-CRF		
Contract Header												
	P	R	F1	P	R	F1	P	R	F1	P	R	F1
Title	**96.0**	**96.4**	**96.2**	94.7	94.9	94.8	93.1	93.2	93.1	93.0	93.7	93.4
Party	**95.3**	**88.9**	**92.0**	93.7	86.2	89.8	88.4	79.4	83.6	89.4	87.2	88.3
S. Date	**96.8**	**97.4**	**97.1**	91.3	96.6	93.8	91.3	92.7	92.0	94.4	96.3	95.3
E. Date	94.6	**96.9**	95.7	**96.9**	95.1	**95.9**	92.0	88.5	90.1	86.9	91.3	89.0
MACRO-AVG	**95.7**	**94.9**	**95.2**	94.1	93.2	93.6	91.2	88.4	89.7	90.9	92.1	91.5
Applicable Law												
	BILSTM-CRF			DILATED-CNNS-CRF			TRANSFORMERS-CRF			BERT-CRF		
	P	R	F1	P	R	F1	P	R	F1	P	R	F1
Jurisdiction	**79.7**	**72.4**	**75.9**	69.6	67.6	68.4	73.6	58.5	65.0	74.7	66.8	70.5
Gov. Law	**98.1**	**96.3**	**97.2**	95.1	92.5	93.8	98.0	90.3	94.0	93.8	92.2	93.0
MACRO-AVG	**88.9**	**84.4**	**86.5**	82.3	80.0	81.1	85.8	74.4	79.5	84.3	79.5	81.8
Lease Header												
	BILSTM-CRF			DILATED-CNNS-CRF			TRANSFORMERS-CRF			BERT-CRF		
	P	R	F1	P	R	F1	P	R	F1	P	R	F1
Property	**67.0**	**65.8**	**66.2**	61.8	61.8	61.7	53.9	50.1	51.8	54.1	56.1	55.1
Landlord	**87.7**	**86.6**	**87.2**	83.4	83.8	83.6	76.5	68.7	72.3	80.6	81.9	81.2
Tenant	**90.7**	**90.9**	**90.8**	89.7	87.8	88.7	81.5	72.4	76.6	85.1	87.1	86.1
S. Date	**92.4**	**95.0**	**93.7**	91.7	93.4	92.5	88.2	90.5	89.3	89.8	93.2	91.5
E. Date	**88.7**	90.8	**89.7**	81.1	87.5	84.1	79.9	71.1	75.2	85.1	**91.9**	88.3
T. Date	**93.9**	85.4	**89.3**	91.3	84.2	87.6	73.2	67.7	70.2	86.3	**87.0**	86.6
Period	**86.6**	**89.1**	**87.8**	81.9	87.0	84.3	76.5	75.5	75.8	81.8	88.8	84.7
Rent	**86.5**	86.0	**86.2**	81.2	82.5	81.7	81.4	74.3	77.5	82.2	**88.2**	85.0
MACRO-AVG	**86.7**	**86.2**	**86.4**	82.8	83.5	83.0	76.4	71.3	73.6	80.6	84.2	82.3

Table 2.3: Precision (P), Recall (R), and F1-score across neural methods, now always using a CRF layer on top.

is the concatenation of its word, POS, and shape embeddings. Contrary to recent findings in sequence labeling (Strubell et al., 2017; Devlin et al., 2019), BILSTMs outperform DILATED-CNNs, stacked TRANSFORMERs, and BERT in all cases. Notice the particularly poor performance of TRANSFORMERS-CRF, which uses the same pre-trained WORD2VEC embeddings as BILSTM-CRF and no other pre-training. This observation highlights the superiority of BILSTMs over TRANSFORMERs, when pre-training is limited to word embeddings, in the tasks we consider. More precisely, comparing TRANSFORMER-based methods (TRANSFORMERS-CRF, BERT-CRF) to BILSTMs across entity types, we observe that the largest performance drop is in: *parties* (8.4 and 3.7 F1 decrease for TRANSFORMERS-CRF and BERT BERT-CRF, respectively), *jurisdiction* (10.9, 5.4), *property* (14.4, 11.1), *landlord* (14.9, 6), *tenant* (14.2, 4.7), and *period* (12.0, 3.1). This could be attributed to the

fact that, although TRANSFORMERs and BERT include positional embeddings and have large receptive fields, BILSTM-based models still cope better with *long-term dependencies* and *sequentiality*, which are important in legal documents. For example, to distinguish start and effective dates, or tenants, landlords and other parties (e.g., guarantors, etc.), or property address and other locations, one often has to consider a broader context than in generic named entity recognition. The particular order (sequentiality) of the context words is also important. For example, in the sentence *"This Service Agreement is signed on **February 26th, 2021**, and effective as of **May 1st, 2021**."*, the relative position of the words 'signed' and 'effective' is crucial to discriminate the two dates.

Considering these initial findings for the CRF-augmented neural methods, we conduct additional ablations studies to measure (a) the impact of CRFs; (b) the impact of alternative feature representations, such as GLOVE embeddings (Pennington et al., 2014), adding a character-level encoder, and ELMO embeddings (Peters et al., 2018); and (c) the impact of in-domain pre-training and lack of recurrency for BERT.

	Contract Header			Applicable Law			Lease Header		
	P	R	F1	P	R	F1	P	R	F1
BILSTMs	93.4	94.0	93.7	81.6	80.7	81.1	82.0	82.7	82.3
+ CRF	**95.7**	**94.9**	**95.2**	**88.9**	**84.4**	**86.5**	**86.7**	**86.2**	**86.4**
DILATED-CNNs	84.2	88.0	86.0	68.7	72.7	70.5	65.9	74.3	69.8
+ CRF	94.1	93.2	93.6	82.3	80.0	81.1	82.8	83.5	83.0
TRANSFORMERs	81.8	86.4	84.0	54.5	53.9	54.1	58.0	64.1	60.8
+ CRF	91.2	88.4	89.7	85.8	74.4	79.5	76.4	71.3	73.6
BERT	90.0	90.9	90.4	78.3	78.1	78.2	77.0	79.8	78.2
+ CRF	90.9	92.1	91.5	84.3	79.5	81.8	80.6	84.2	82.3

Table 2.4: Macro-averaged results with/without CRF layers.

Impact of CRFs: Table 2.4 compares the performance of all encoders with and without CRFs. In each dataset, results are macro-averaged over contract element types. Similarly to prior sequence labeling studies (Lample et al., 2016; Strubell et al., 2017), we find that CRFs always improve performance, especially for non-BILSTM encoders that lack recurrency (DILATED-CNNs, TRANSFORMERs). DILATED-CNNs stack convolutional layers with increasingly larger strides to quickly obtain a large receptive field, as already discussed. TRANSFORMERs solely rely on additive positional embeddings and are otherwise insensitive to word order.

Alternative Feature Representations: Table 2.5 compares the performance of BILSTM-CRF, the best encoder, with different input representations. Generic word embeddings (GLOVE) are vastly outperformed by domain-specific ones (W2V-WORD) in two out of

	Contract Header			Applicable Law			Lease Header		
	P	R	F1	P	R	F1	P	R	F1
GLOVE (generic)	90.2	89.6	89.9	88.7	84.3	86.4	66.1	65.8	65.9
W2V-WORD (domain-specific)	95.7	**95.1**	**95.4**	89.0	84.1	86.5	87.0	86.2	86.6
W2V-ALL (incl. POS, shape)	95.7	94.9	95.2	88.9	**84.4**	86.5	86.7	86.2	86.4
W2V-ALL+CHAR	**96.1**	94.0	95.0	89.3	82.2	85.5	**87.8**	86.1	**86.9**
W2V-ALL+ELMO	95.8	94.8	95.3	89.3	84.2	86.7	86.0	**87.5**	86.7

Table 2.5: Macro-averaged BILSTM-CRF results with alternative input representations.

three datasets (Contract Header, Lease Header). Adding POS tag and token shape embeddings (W2V-ALL) does not improve overall performance (see F1 scores). Adding character-level word embeddings (W2V-WORD+CHAR) also has no consistent or significant positive impact on F1. ELMO embeddings also do not lead to consistent noticeable improvements, possibly because the generic corpora that ELMO was trained on are very different than contracts. We suspect that in-domain knowledge is not important in Applicable Law, as entities (*governing law*, *jurisdiction*) are mostly locations (e.g., US states or districts and nationality adjectives) that are properly covered in generic corpora used to pre-train GLOVE and ELMO.

	Contract Header		Applicable Law		Lease Particulars	
	Full Vocab.	Entities	Full Vocab.	Entities	Full Vocab.	Entities
BERT	2.10	1.81	1.71	1.23	2.55	2.47
LEGAL-BERT	2.09	1.92	1.61	1.39	2.45	2.47

Table 2.6: *Word Fragmentation Ratio* (WFR), i.e., average ratio of sub-word units per word, for both BERT variants. We report WFRs (i) for the full vocabulary in each dataset and (ii) only for vocabulary tokens included in entities.

Why is BERT worse than BILSTM? In order to better understand the failure of BERT, which currently dominates NLP, in contract element extraction, we study three factors (see Table 2.7): (a) the importance of in-domain pre-training, where we compare LEGAL-BERT (a new BERT model pre-trained on legal corpora) with the original pre-trained BERT; (b) the impact of recurrency, comparing with a BILSTM-based model that operates on top of BERT (BERT-BILSTM-CRF) or on top of LEGAL-BERT (LEGAL-BERT-BILSTM-CRF) in two different settings (*fine-tuned* vs. *frozen*, i.e., without fine-tuning); and (c) the impact of using sub-word embeddings that lead to word fragmentation (Table 2.6), again comparing to BERT-BILSTM-CRF and LEGAL-BERT-BILSTM-CRF. Models relying on sub-word tokens need to correctly classify more tokens, contrary to models relying on words, which may lead to a performance drop, especially when sub-word tokenizers lead

	Contract Header			Applicable Law			Lease Particulars		
	P	R	F1	P	R	F1	P	R	F1
Fine-tuned Stand-alone Pre-trained transformers									
bert-crf	90.9	92.1	91.5	84.3	79.5	81.8	80.6	84.2	82.3
legal-bert-crf	93.6	93.5	93.5	81.6	78.8	80.1	81.6	83.2	82.4
bilstm-crf (Similar to Table 2.5)									
word2vec	95.7	95.1	95.4	89.0	**84.1**	86.5	**87.0**	86.2	86.6
bert (frozen)	95.1	92.4	93.7	90.5	83.9	**87.1**	84.4	85.5	84.9
legal-bert (frozen)	95.0	**95.4**	95.1	**90.6**	83.4	86.7	85.1	**88.2**	86.6
Finetuning End-to-End (bert-bilstm-crf)									
bert-bilstm-crf	94.0	94.1	94.0	88.1	82.7	85.2	83.8	86.4	85.0
legalbert-bilstm-crf	**96.3**	95.3	**95.8**	87.3	82.4	84.7	86.9	87.1	**86.9**

Table 2.7: Macro-averaged results for bert-based variants: *stand-alone* (upper section) and *combined* with bilstms (middle, lower). In the middle section, the bilstm is fed with frozen word2vec embeddings, or frozen embeddings produced by bert or legal-bert. In the lower section, bert and legal-bert are also fine-tuned during training.

to more word fragmentation. Table 2.7 reports results for the aforementioned models. Our observations are the following:

- Inspecting Table 2.7, we observe that legal-bert leads to better performance than bert in two out of three datasets, further highlighting the importance of in-domain knowledge in Contract Header and Lease Header, as we originally observed in Table 2.5. While bert is better in the Applicable Law subset, legal-bert seems to better cover companies and their roles, as suggested by the higher F1-score of legal-bert-crf comparing to bert-crf in the corresponding entity types, i.e., *party* (89.6 vs. 88.3), *landlord* (85.2 vs. 81.2) and *tenant* (90.5 vs. 86.1).[10] This observation is consistent across all methods presented in Table 2.7, especially when bilstms are included. Note that legal-bert uses a vocabulary that presumably better accommodates legal language and has been pre-trained on legal corpora, while bert uses a generic vocabulary and has been pre-trained in English Wikipedia and the Children Books Corpus (Hill et al., 2016).

- Inspecting the mid and lower sections of Table 2.7, we observe that the methods that combine bert or legal-bert with bilstm-crf are comparable to bilstm-crf relying on word2vec embeddings, especially when models are fine-tuned end-to-end (Table 2.7, lower section). These empirical results support two important

[10] *Per entity type* scores are not presented in Table 2.7 for brevity.

findings: (a) despite their ability to capture long-term dependencies, the lack of recurrency in TRANSFORMER-based methods leads to worse performance, as most of the entity types greatly depend on context and sequentiality, which can be better captured by BILSTMs; (b) BILSTM-CRF has comparable performance when operating on in-domain WORD2VEC or LEGAL-BERT embeddings; thus, there is no concrete evidence that the use of sub-words and the corresponding word fragmentation negatively affect performance. It seems that adding a BILSTM-CRF on top of BERT or LEGAL-BERT subword embeddings alleviates any potential negative impact caused by the word fragmentation.

An important take away is that in contract element extraction, the much simpler (at least in terms of number of trainable parameters) BILSTM-CRF with frozen in-domain WORD2VEC embeddings is very competitive to methods that employ BERT models, even when the latter are given in-domain pre-training (LEGAL-BERT) and combined with BILSTMs (LEGAL-BERT-BILSTM-CRF). Following the in depth quantitative analysis (evaluation), where we compare classification performance across all methods, it would be interesting to conduct a qualitative (error) analysis to identify the most prominent sources of errors. Although given the extend of this study, we leave this part for future work.

No.	Gold Class	Sentences/Clauses
1	Obligation	The Supplier ***is obliged to*** meet and comply with the Approved Requirements.
	None	Details shall be determined in the individual contracts.
2	Prohibition	***No*** Provider staff ***will*** provide services to any Customer Competitor.
	Obligation	Provider ***will*** take such measures to prevent these actions.
3	Prohibition	Provider ***is not entitled*** to suspend this Agreement prior to the lapse of the fifth year.
4	Oblig./Prohib. List Intro	The Supplier ***shall***:
	Obligation List Item	***(a)*** only process the Personal Data in accordance with Client's written instructions;
	Prohibition List Item	***(b) not*** transfer any Personal Data to any other third parties;
5	Oblig./Prohib. List Intro	The Receiving Party ***will***:
	Obligation List Item	***(i)*** keep the Confidential Information secret and confidential;
	Prohibition List Item	***(ii) not*** disclose the Confidential Information to any person other than in accordance with Clauses 13.3; and
	Prohibition List Item	***(iii) not*** use the Confidential Information other than for the purposes of this Agreement.
6	Oblig./Prohib. List Intro	A Party ***shall not*** directly solicit the employment of:
	Prohibition List Item	***(i)*** in the case of Client, Supplier's employees engaged in the provision of the Services,
	Prohibition List Item	***(ii)*** in the case of Supplier, Client's employees engaged.
	None	Nothing in this section will restrict either Party's right to recruit.

Table 2.8: Examples of sentences and clauses, with human annotations of classes. Terms that are highly indicative of the classes are shown in bold and underlined here.

2.5 Obligation Extraction

Obligation extraction is a kind of *deontic* sentence (or clause) classification. Obligations describe the terms of the contracts that have been agreed upon between the contracting parties. Different firms may use different or finer deontic classes (e.g., distinguishing between payment and delivery obligations), but obligations and prohibitions are the most common coarse deontic classes, thus we experiment with these two.

In Table 2.8, we present examples of obligations and prohibitions. In examples 1-3, we see common cases, where full sentences have been classified in different categories. Across examples, we see only a small fraction of the phrasing that is commonly used, which heavily relies on the use of modal verbs and negation. What is also common in the contractual language is the extensive use of clause lists describing alternative cases and exceptions upon the agreed terms (obligations). In examples 4-6, we present sentences that have been formatted as clause lists, including both obligations and prohibitions. In these cases, we have to consider the list structure to correctly categorize clauses. Based on this observation, we consider the following label set:

- **Obligation**: *Obligations* (Table 2.8, examples 1-2) are full sentences defining an obligatory term, i.e., the party must perform an action.
- **Prohibition**: *Prohibitions* (Table 2.8, examples 2-3) are full sentences defining a prohibited action, i.e., the party shall not perform an action.
- **Obligation/Prohibition Intro**: These are introductory clauses (Table 2.8, examples 4-6) for a group of obligations and/or prohibitions.
- **Obligation List Item**: These are list items (cases) (Table 2.8, examples 4-5) referring to obligatory terms.
- **Prohibition List Item**: These are list items (cases) (Table 2.8, examples 4-6) referring to prohibited terms.

Gold Class	Train	Val	Test	Class Percentage
None	15,401	3,905	4,141	51,2%
Obligation	11,005	2,860	970	32.8%
Prohibition	1,172	314	108	3.5%
Obligation List Intro	828	203	70	2.3%
Obligation List Item	2888	726	255	8,6%
Prohibition List Item	251	28	19	0.6%
Total	**31,545**	**8,036**	**5,563**	100%

Table 2.9: Sentences/clauses statistics on the Obligation Extraction dataset.

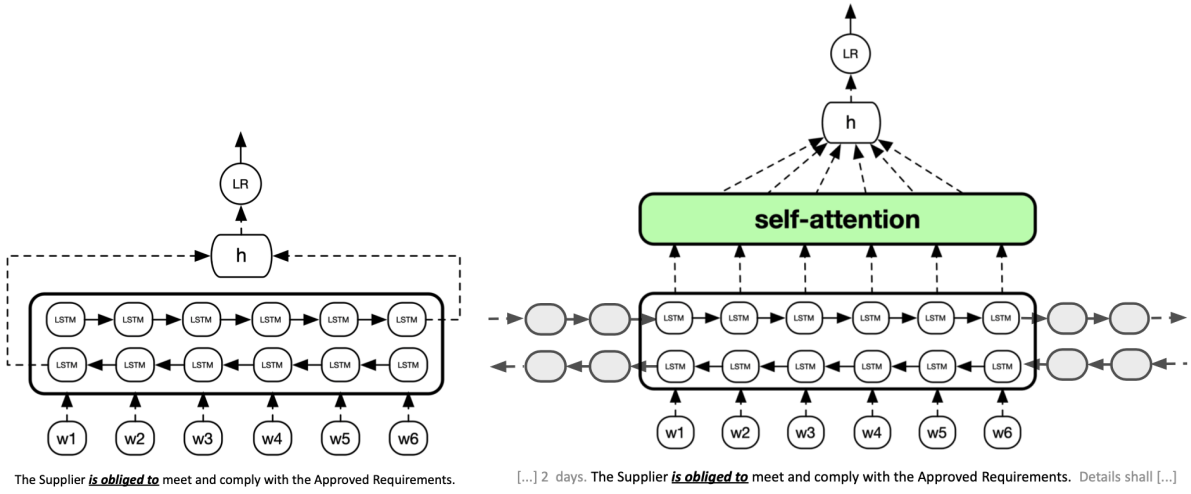
2.5.1 Dataset

We experimented with a dataset containing 9,400 sections from the main bodies (excluding introductions, covers, recitals) of 100 randomly selected English service agreements. The sections were preprocessed by a custom sentence splitter, which in clause lists (Examples 4–6 in Table 2.8) treats the introductory clause and each nested clause as separate sentences, since each nested clause may belong to a different class.[11] Table 2.9 shows the distribution of sentences in the six gold (correct) classes and the number of sentences in training, validation and test subsets. Each section was annotated by a single law student (5 students in total). All the annotations were checked and corrected by a single paralegal expert, who produces annotations of this kind on a daily basis, based on strict guidelines of the firm that provided the data. The dataset cannot be released publicly due to IPR.

[11] We use NLTK's splitter (`http://www.nltk.org/`), with additional post-processing to better handle legal text based on regular expressions. The splitter produced 31,545 training, 8,036 validation, and 5,563 test sentences/clauses (Table 2.9).

2.5.2 Methods

Across all RNN-based methods, we use in-domain pre-trained 200-dimensional word embeddings, 25-dimensional POS tag embeddings, and 25-dimensional token shape embeddings, as described in Section 2.4.4. Each token is represented by the concatenation of its word, POS, and token shape embeddings.



(a) (b)

Figure 2.5: (a) Vanilla BILSTM. (b) BILSTM with self-attention used on its own (BILSTM-ATT) or as the sentence encoder of the hierarchical BILSTM (H-BILSTM-ATT). In X-BILSTM-ATT, the BILSTM also considers the words of surrounding sentences.

BILSTMs: The first classifier we considered processes a single sentence (or clause) at a time. It feeds the concatenated word, POS, shape embeddings $(e_1, \ldots, e_n)$ of the tokens $w_1, w_2, \ldots, w_n$ of the sentence to a forward LSTM, and (in reverse order) to a backward LSTM, obtaining the forward and backward hidden states ($\overrightarrow{h}_1, \ldots, \overrightarrow{h}_n$ and $\overleftarrow{h}_1, \ldots \overleftarrow{h}_n$). The concatenation of the last states ($h = [\overrightarrow{h}_n; \overleftarrow{h}_1]$) is fed to a multinomial Logistic Regression (LR) layer, which produces a probability per class (Figure 2.5a).

BILSTM-ATT: When self-attention is added (Figure 2.5b), the sentence (or clause) is represented by the weighted sum (h) of the hidden states ($h_t = [\overrightarrow{h}_t; \overleftarrow{h}_t]$) of the BILSTM, where $a_1, \ldots, a_n \in \mathbb{R}$ are attention scores and v is a vector used to map h_t to a scalar.:

$$h = a_1 h_1 + \cdots + a_t h_t + \cdots + a_n h_n \tag{2.16}$$

$$a'_t = \tanh(v^T h_t + b) \tag{2.17}$$

$$a_t = \text{softmax}(a'_t; a'_1, \ldots, a'_n) \tag{2.18}$$

Again, h is then fed to a multinomial LR layer.

X-BILSTM-ATT: In an extension of BILSTM-ATT, called X-BILSTM-ATT, the BILSTM is fed with the token embeddings (e_t) not only of the sentence being classified, but also of the previous (and following) tokens (faded parts of Figure 2.5b), up to K previous (and K following) tokens. This might allow the BILSTM to 'remember' key parts of the surrounding sentences (e.g., a previous clause ending with 'shall not:') when producing the context-aware embeddings (states h_t) of the current sentence. The self-attention mechanism still considers the states (h_t) of the tokens of the current sentence only, and the sentence representation (h) is still computed as in Equation 2.16.

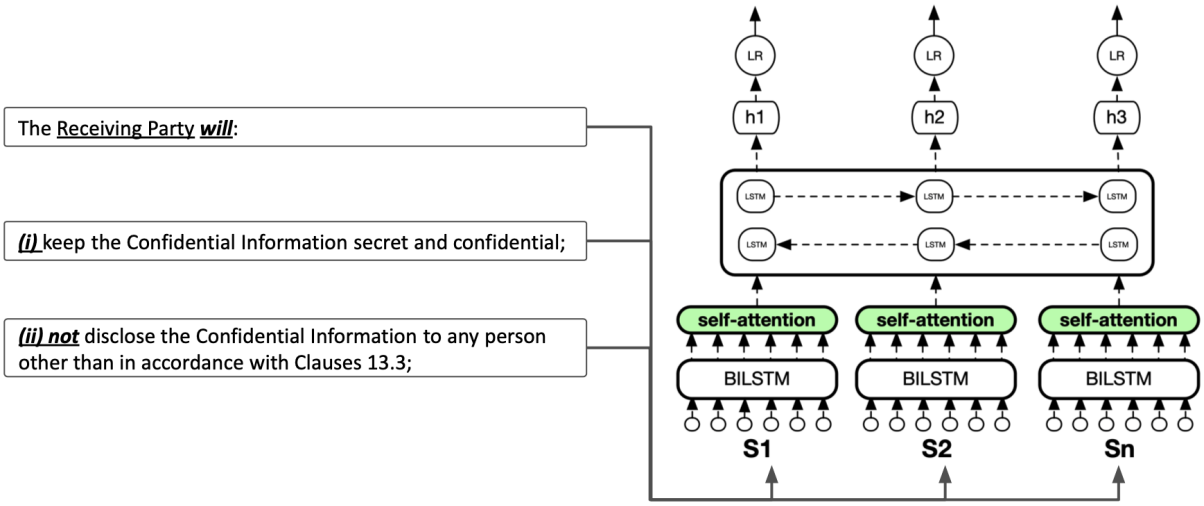


Figure 2.6: The hierarchical BILSTM (H-BILSTM-ATT).

H-BILSTM-ATT: The hierarchical BILSTM classifier, H-BILSTM-ATT, considers all the sentences (or clauses) of an entire section ($S_1, S_2, \ldots, S_n$). Each sentence (or clause) is first turned into a sentence embedding (h), as in BILSTM-ATT (Figure 2.5b). The sequence of sentence embeddings is then fed to a second BILSTM (Figure 2.6), whose hidden states ($h_t^{(2)} = [\overrightarrow{h}_t^{(2)}; \overleftarrow{h}_t^{(2)}]$) are treated as context-aware sentence embeddings. The latter are passed to a multinomial LR layer, producing a probability per class, for each sentence (or clause) of the section. We hypothesize that H-BILSTM-ATT would perform better because it considers an entire section at a time, and salient information about a sentence or clause (e.g., that the opening clause of a list contains a negation or modal) can be 'condensed' in its sentence embedding and interact with the sentence embeddings of distant sentences or clauses (e.g., a nested clause several clauses after the opening one) in the upper BILSTM (Figure 2.6).

2.5.3 Experimental SetUp

Memory constraints did not allow including more than $K = 150$ previous (and following) tokens in X-BILSTM-ATT, although in most cases the model considers all section tokens. All methods were implemented using KERAS (`https://keras.io/`).

Across all methods, hyper-parameters were tuned by grid-searching the following sets, and selecting the values with the best validation loss: LSTM hidden units {100, 200, 300}, batch size {8, 16, 32}, and drop-out rate {0.4, 0.5, 0.6}. We used categorical cross-entropy loss, Glorot initialization (Glorot and Bengio, 2010), Adam (Kingma and Ba, 2015), learning rate 0.001, and early stopping on the validation loss. We report precision, recall, and F1 score per class, as well as micro- and macro-averages.

	BILSTM			BILSTM-ATT			X-BILSTM-ATT			H-BILSTM-ATT		
Gold Class	P	R	F1	P	R	F1	P	R	F1	P	R	F1
None	0.95	0.91	0.93	0.97	0.90	0.93	0.96	0.90	0.93	**0.98**	**0.96**	**0.97**
Obligation	0.75	0.85	0.79	0.75	0.88	0.81	0.75	0.87	0.81	**0.87**	**0.92**	**0.90**
Prohibition	0.67	0.62	0.64	0.74	0.75	0.74	0.65	0.75	0.70	**0.84**	**0.83**	**0.84**
Obl. List Begin	0.70	0.86	0.77	0.71	0.85	0.77	0.72	0.75	0.74	**0.90**	**0.89**	**0.89**
Obl. List Item	0.53	0.66	0.59	0.48	0.70	0.57	0.49	0.78	0.60	**0.85**	**0.94**	**0.89**
Proh. List Item	0.59	0.35	0.43	0.61	0.55	0.59	**0.83**	0.50	0.62	0.80	**0.84**	**0.82**
Macro-average	0.70	0.70	0.70	0.73	0.78	0.74	0.73	0.76	0.73	**0.87**	**0.90**	**0.89**
Micro-average	0.90	0.88	0.88	0.90	0.88	0.89	0.90	0.88	0.89	**0.95**	**0.95**	**0.95**

Table 2.10: Precision, recall and F1 scores, with the best results in bold.

2.5.4 Experiments

Table 2.10 reports the performance across methods. The self-attention mechanism of BILSTM-ATT leads to clear overall improvements (0.74 vs. 0.70 in macro F1) comparing to the plain BILSTM, supporting the hypothesis that self-attention allows the classifier to focus on indicative tokens. Allowing the BILSTM to consider tokens of neighboring sentences (X-BILSTM-ATT) does not lead to any clear overall improvements. The hierarchical H-BILSTM-ATT clearly outperforms the other three methods (0.89, macro F1), supporting the hypothesis that considering entire sections and allowing the sentence embeddings to interact in the upper BILSTM (Figure 2.6) is beneficial. Notice that the three flat methods (BILSTM, BILSTM-ATT, X-BILSTM-ATT) obtain particularly lower F1 scores, compared to H-BILSTM-ATT, in the classes that correspond to nested clauses (obligation list item, prohibition list item). This is due to the fact that the flat methods have no (or only limited, in the case of X-BILSTM-ATT) view of the previous sentences, which often indicate if a nested clause is an obligation or prohibition (examples 4–6 in Table 2.8). Nonetheless, there are also improvements in the rest of the classes (Obligation, Prohibition).

2.6 Conclusions

In this chapter, we investigated two applications of information extraction to contracts. In contract element extraction, we found that BILSTM-based models lead to state-of-the-art results, even comparing to pre-trained TRANSFORMER-based models, i.e., BERT, that currently dominate the NLP literature. We linked this finding with the lack of inherent recurrency in the TRANSFORMERs architecture. Moreover, we observed that in-domain knowledge, as expressed in language and captured by in-domain WORD2VEC embeddings and LEGAL-BERT, leads to performance improvements in two out of three cases, we considered. In the second task, obligation extraction, we found that proper modeling of the text structure with hierarchical LSTMs leads to the best results with vast performance improvements in the list item categories, but also in the rest of the categories that correspond to full single sentences. This finding highlights the importance of context in the form of inter-sentence relations in this sentence classification task.

In future work, we would like to further investigate the impact of the lack of recurrency in TRANSFORMER-based models in contract element extraction with a qualitative analysis by identifying interesting examples to better understand this phenomenon. With respect to classification performance, an obvious direction would be to pre-train a BERT encoder, specialized solely on contractual text that could possibly benefit the overall performance. Similarly, it would be interesting to explore the use of TRANSFORMER-based models in the second task, obligation extraction, by employing a similar method to hierarchical BERT (HIER-BERT), discussed in Section 4.4.3 below.

Chapter 3

Large-Scale Multi-Label Classification for legal documents

3.1 Introduction

Large-scale Multi-label Text Classification (LMTC) is the task of assigning a subset of labels from a large predefined set (typically thousands) to a given document. LMTC has a wide range of applications in Natural Language Processing (NLP), such as building web directories (Partalas et al., 2015), labeling scientific publications with concepts from ontologies (Tsatsaronis et al., 2015), associating medical records with diagnostic and procedure labels (Mullenbach et al., 2018; Rios and Kavuluru, 2018), products with categories (Lewis et al., 2004), and legislation with relevant legal concepts (Mencia and Fürnkranzand, 2007). Apart from the large label space, LMTC datasets often have skewed label distributions (e.g., some labels have few or no training examples). Moreover, the label set and the hierarchies are periodically updated, thus requiring zero- and few-shot learning to cope with newly introduced labels. On top of the general LMTC challenges, the extensive size of legal documents (especially for modern pre-trained TRANSFORMER-based language models (Devlin et al., 2019; Liu et al., 2019b)) and the peculiarities of legal language can also be an additional challenge in legal applications.

In the legal domain, there is an overwhelming number of produced documents. From new legislation, regulations, and policies introduced on a daily basis to contracts shared and signed between companies and individuals and of course case judgments coming from various courts. Without technological assistance, there is a great challenge to successfully cope with the vast amount of information, leading to unexpected inconsistencies and penalties derived by legal (regulatory, contractual) noncompliance. Thus, employing supportive automated solutions to accelerate and improve document indexing, searching and retrieval are particularly crucial for legal professionals. Categorizing legal documents

considering refined categories, usually relying on legal taxonomies, is an important part of intelligent legal services. A typical case is the indexing of legal documents with EUROVOC concepts in systems of EU institutions, e.g., in web legislative databases, such as EUR-LEX (`https://eur-lex.europa.eu`), the official portal of EU, and CELLAR, which consists of a SPARQL endpoint and an HTTP RESTful API.[1] The automation of the aforementioned NLP applications, among others could minimize the cost of labeling data, while also benefit the user experience by improving the quality of labeling, i.e., reduce missing labels, while searching among thousands.

3.2 Related Work

Mencia and Fürnkranzand (2007) introduced a legal LMTC application and dataset obtained on the EUR-LEX database, which includes EU laws that have been tagged with EUROVOC concepts. They used multiple binary Perceptrons, one for each label, and multi-label pairwise Perceptrons on top of Bag-of-Words (BOW) representations. While the methods seem primal and inefficient by today's standards, the EUR-LEX dataset is widely adopted as a notable benchmark in LMTC literature.

Liu et al. (2017) proposed a CNN-based classifier similar to that of Kim (2014) for LMTC. They reported results on several benchmark datasets, most notably: RCV1 (Lewis et al., 2004), containing news articles; EUR-LEX (Mencia and Fürnkranzand, 2007), containing legal documents; AMAZON13K (McAuley and Leskovec, 2013), containing product descriptions; and Wiki-30K (Zubiaga, 2012), containing Wikipedia articles. Their proposed method outperformed both tree-based methods (e.g., FASTXml (Prabhu and Varma, 2014)) and embedding-based methods (e.g., SLEEC (Bhatia et al., 2015), FASTTEXT (Bojanowski et al., 2016)), relying on similarities given representations (e.g., centroids of word embeddings) in a latent space. You et al. (2018) used LSTMs with self-attention comparing also with tree-based methods and alternative deep learning methods, including vanilla LSTMs and CNNs. Their method outperformed the other approaches in three out of four datasets, demonstrating the effectiveness of attention-based RNNs.

Mullenbach et al. (2018) investigated the use of a label-wise attention mechanism in medical code prediction on the MIMIC-II and MIMIC-III datasets (Johnson et al., 2017). MIMIC-II and MIMIC-III contain over 20,000 and 47,000 documents tagged with approximately 9,000 and 5,000 ICD9 code descriptors, respectively. Their best method, Convolutional Attention for Multi-Label Classification (CAML), includes multiple attention heads, one for each one of the L labels. CAML outperformed logistic regression, vanilla

[1] `https://data.europa.eu/euodp/en/data/dataset/sparql-cellar-of-the-publications-office`

BIGRUs and CNNs. Another important fact is that CAML was found to have the best interpretability in comparison with the rest of the methods in human readers' evaluation.

Rios and Kavuluru (2018) discussed the challenge of few- and zero-shot learning on the MIMIC datasets. The same authors proposed a new method, named Zero-Shot Attentive CNN, called ZACNN here, which is similar to CAML, but also exploits the provided label descriptors. While ZACNN did not outperform CAML overall on MIMIC-II and MIMIC-III, it had exceptional results in few-shot and zero-shot learning, being able to identify labels with few or no instances at all in the training sets. In an extension of ZACNN, called ZAGCNN here, the authors also applied graph convolutions to hierarchical relations of the labels, which improved the performance on few- and zero-shot learning.

3.3 Contributions

- We publish a new dataset (EURLEX57K) that includes 57,000 EU laws tagged with 4,271 different EUROVOC concepts. A former version of the EUR-LEX dataset, released by Mencia and Fürnkranzand (2007), included 19,600 documents tagged with 3,993 different EUROVOC concepts. With almost one-third of the size of EURLEX57K, this dataset was one of the smallest among LMTC benchmarks.

- Current state-of-the-art LMTC models are based on Label-Wise Attention Networks (LWANs) (Mullenbach et al., 2018) relying on CNN encoders (CNN-LWAN). We study the effect of using RNN-based, PLT-based and TRANSFORMER-based methods. We show that RNN-based baselines (BIGRU-ATT, HAN) outperform CNN-based LWANs, and we propose a new BIGRU-based LWAN which is even better. Moreover, we show that newly introduced pre-trained TRANSFORMER-based methods (BERT, ROBERTA) further improve results, without relying on the label-wise attention mechanism, and we propose a new state-of-the-art method that combines BERT with LWAN achieving the best results overall.

- Few-shot and zero-shot learning are vastly understudied in LMTC. Following the work of Rios and Kavuluru (2018) for few and zero-shot learning on MIMIC-III in the biomedical domain, we investigate the use of structural information from the label hierarchy in LWAN. We propose new zero-shot capable LWAN-based models with improved performance in these settings.

- In a more general perspective, we validate our findings in other non-legal LMTC benchmark datasets (AMAZON13K, MIMIC-III). We have similar findings for AMAZON13K, while we interestingly show that TRANSFORMER-based methods underperform in MIMIC-III and study the possible reasons for this result.

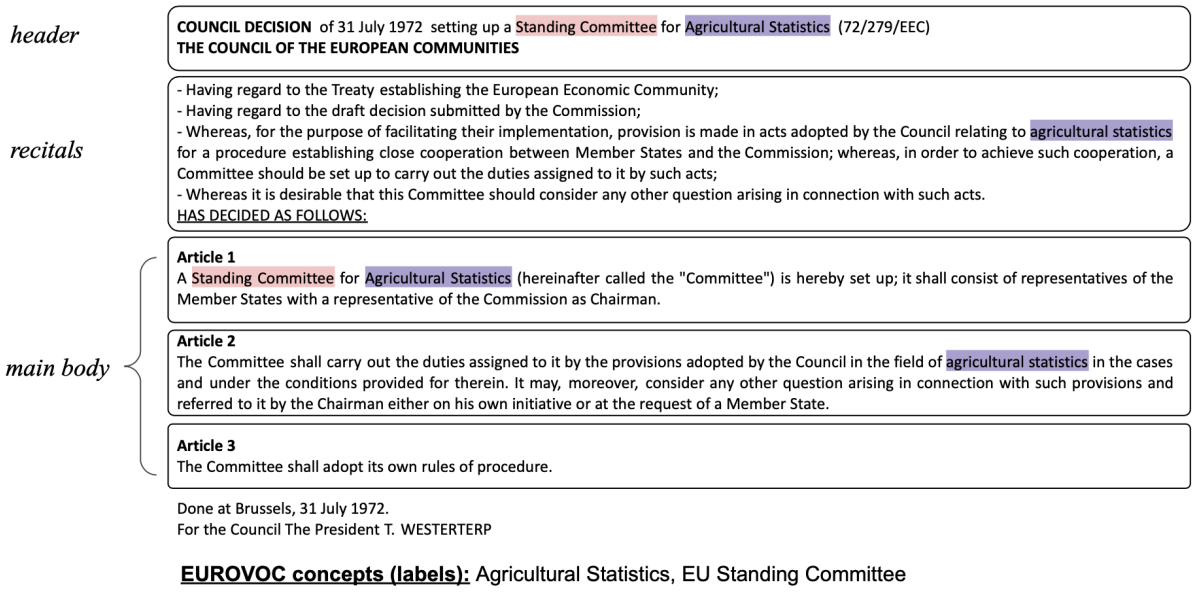


Figure 3.1: Council Decision 72/279/EEC. Rounded boxes indicate document zones (called 'sections' in this thesis). The assigned EUROVOC concepts are presented in the bottom. Spans with highly indicative words are highlighted for the purposes of the example; the highlighting is not included in the dataset.

3.4 Dataset

3.4.1 EUROVOC Thesaurus

EUROVOC is a multilingual thesaurus maintained by the Publications Office of the European Union (EU). It is used by the European Parliament, the national and regional parliaments in Europe, some national government departments, and other European organizations. The current version of EUROVOC contains more than 7,000 concepts referring to various activities of the EU and its Member States (e.g., economics, health-care, trade, etc.). It has also been used for indexing documents in systems of EU institutions, e.g., in web legislative databases, such as EUR-LEX and CELLAR. All EUROVOC concepts are represented as tuples called *descriptors*, each containing a unique numeric identifier and a (possibly) multi-word description of the concept, for example ⟨1309, import⟩, ⟨693, citrus fruit⟩, ⟨192, health control⟩, ⟨863, Spain⟩, ⟨2511, agri-monetary policy⟩.

3.4.2 The new Dataset: EURLEX57K

Our dataset (EURLEX57K) can be viewed as an improved version of the EUR-LEX dataset released by Mencia and Fürnkranzand (2007), which included 19,601 documents tagged with 3,993 different EUROVOC concepts. While EUR-LEX has been widely used in LMTC research, it is less than half the size of EURLEX57K and one of the smallest among

LMTC benchmarks.[2] Over the past years the EUR-LEX archive has been widely expanded. EURLEX57K is a more up to date dataset including 57,000 pieces of EU legislation from the EUR-LEX portal. All documents have been annotated by the Publications Office of EU with multiple concepts from the EUROVOC thesaurus. EURLEX57K is split in training (45,000 documents), development (6,000), and validation (6,000) subsets (Table 3.1).[3]

All documents are structured in four major zones (Figure 3.1): the *header* including the title and the name of the legal body that enforced the legal act; the *recitals* containing references in the legal background of the decision; the *main body*, which is usually organized in articles; and the *attachments* that usually include appendices and annexes. For simplicity, we will refer to each one of *header*, *recitals*, *attachments* and each of the *main body*'s articles as *sections*. We have pre-processed all documents in order to provide the aforementioned structure.

Subset	Documents (D)	Words/D	Labels/D
Train	45,000	729	5
Dev.	6,000	714	5
Test	6,000	725	5

Table 3.1: Statistics of EURLEX57K dataset.

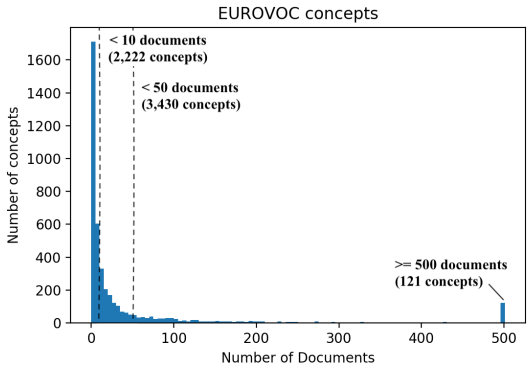


Table 3.2: Distribution of EUROVOC concepts across EURLEX57K documents.

While EUROVOC includes over 7,000 concepts (labels), only 4,271 (59.31%) of them are present in EURLEX57K. Another important fact is that most labels are under-represented; only 2,049 (47,97%) have been assigned to more than 10 documents. Such an aggressive Zipfian distribution (Figure 3.2) has also been noted in other domains, like medical examinations (Rios and Kavuluru, 2018), where LMTC has been applied to index documents with concepts from medical thesauri. The labels of EURLEX57K are divided into three categories: *frequent* labels (746), each occurring in more than 50 training documents (these labels also happen to occur in all three subsets, i.e., training, development, test); *few-shot* labels (3,362), each appearing in 1 to 50 training documents; and *zero-shot* labels (163), which appear in the development and/or test subset, but not in the training documents.

[2]The most notable benchmarks can be found at `http://manikvarma.org/downloads/XC/XMLRepository.html`.

[3]Our dataset is available at `http://nlp.cs.aueb.gr/software_and_datasets/EURLEX57K`, with permission of reuse under European Union Copyright, `https://eur-lex.europa.eu`, 1998–2020.

3.5 Methods

3.5.1 Rule-based and linear methods

For completeness, we experiment first with a naive rule-based baseline, dubbed Exact Match, which assigns only labels whose descriptors can be found verbatim in the document. A second baseline uses Logistic Regression with feature vectors containing TF-IDF scores of n-grams ($n = 1, 2, \ldots, 5$).

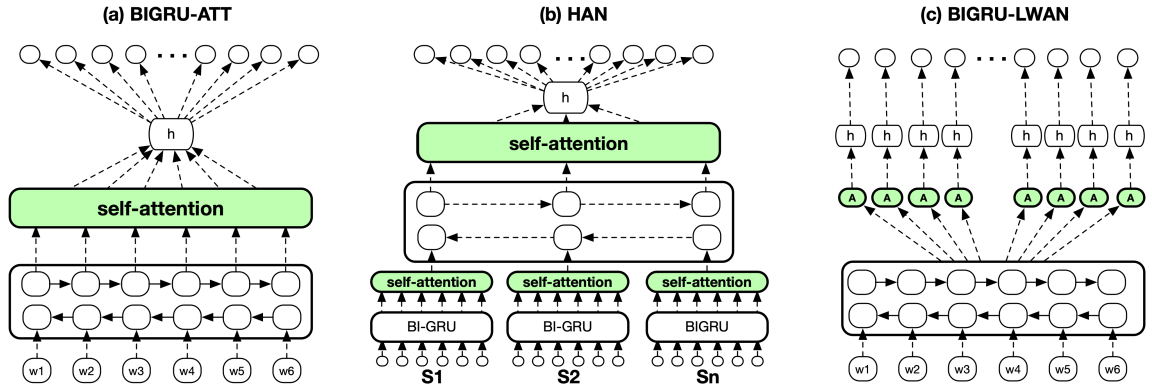


Figure 3.2: Flat neural methods: BIGRU-ATT, HAN, and BIGRU-LWAN.

3.5.2 Flat neural methods

We experiment with neural methods (Figure 3.2) consisting of: (i) a *token encoder* ($\mathcal{E}_w$), which makes token embeddings (w_t) context-aware (h_t); (ii) a *document encoder* ($\mathcal{E}_d$), which turns a document into a single embedding; (iii) an optional *label encoder* ($\mathcal{E}_l$), which turns each label into a label embedding; (iv) a *document decoder* ($\mathcal{D}_d$), which maps the document to label probabilities.

BIGRU-ATT: The first neural method is a BIGRU with self-attention (Xu et al., 2015). This method is very similar to the one presented in Figure 2.5a of the previous chapter, but uses GRUs instead of LSTMs. Each document is represented as the sequence of its word embeddings, which go through a stack of BIGRUs ($\mathcal{E}_w$). A document embedding (h) is computed as the sum of the resulting context-aware embeddings ($h = \sum_i a_i h_i$), weighted by the self-attention scores (a_i), and goes through a dense layer of $L = 4,271$ output units with sigmoids, producing L probabilities, one per label.

HAN: The Hierarchical Attention Network (Yang et al., 2016) is a strong baseline for text classification. We use a slightly modified version, where a BIGRU with self-attention reads the words of each section ($\mathcal{E}_w$), as in BIGRU-ATT but separately per section, producing section embeddings. A second-level BIGRU with self-attention ($\mathcal{E}_d$) reads the section embeddings, producing a single document embedding (h) that goes through a $\mathcal{D}_d$.

CNN-LWAN, BIGRU-LWAN: In the original Label-Wise Attention Network (LWAN), Mullenbach et al. (2018) used a CNN token encoder ($\mathcal{E}_w$), while a modified version that we developed, called BIGRU-LWAN, replaces the CNN encoder with a BIGRU. In both models, $\mathcal{E}_d$, contrary to BIGRU-ATT, uses one attention head per label to generate L document representations d_l (part (c) of Figure 3.2):

$$a_{lt} = \frac{\exp(h_t^\top u_l)}{\sum_{t'} \exp(h_{t'}^\top u_l)},\ d_l = \frac{1}{T}\sum_{t=1}^{T} a_{lt} h_t \tag{3.1}$$

T is the document length in tokens, h_t the context-aware representation of the t-th token, and u_l a trainable vector used to compute the attention scores of the l-th attention head; u_l can also be viewed as a label representation. Intuitively, each head focuses on possibly different tokens of the document to decide if the corresponding label should be assigned. In this model, $\mathcal{D}_d$ employs L linear layers with sigmoid activations, each operating on a different label-wise document representation d_l, to produce the probability of the corresponding label:

$$p_l = \text{sigmoid}(u_l^\top d_l + b_l) \tag{3.2}$$

Here, o_l denotes the probability of l_{th} label, given the label-wise document representation d_l, an equally sized trainable vector u_l, and bias term b_l.

3.5.3 Hierarchical PLT-based methods

In PLT-based methods (PLT stands for Probabilistic Label Trees), each label is represented as the average of the feature vectors of the training documents that are annotated with this label. The root of the PLT corresponds to the full label set. The label set is partitioned into k subsets using k-means clustering, and each subset is represented by a child node of the root in the PLT. The labels of each new node are then recursively partitioned into k subsets, which become children of that node in the PLT. If the label set of a node has fewer than m labels, the node becomes a leaf and the recursion terminates. During inference, the PLT is traversed top-down. At each non-leaf node, a multi-label classifier decides which children nodes (if any) should be visited by considering the feature vector of the document. When a leaf node is visited, the multi-label classifier of that node decides which labels of the node will be assigned to the document.

PARABEL, BONSAI: We experiment with PARABEL (Prabhu et al., 2018) and BONSAI (Khandagale et al., 2019), two state-of-the-art PLT-based methods. PARABEL employs binary PLTs ($k = 2$), while BONSAI uses non-binary PLTs ($k > 2$), which are shallower and wider. In both methods, a linear classifier is used at each node, and documents are represented by TF-IDF feature vectors.

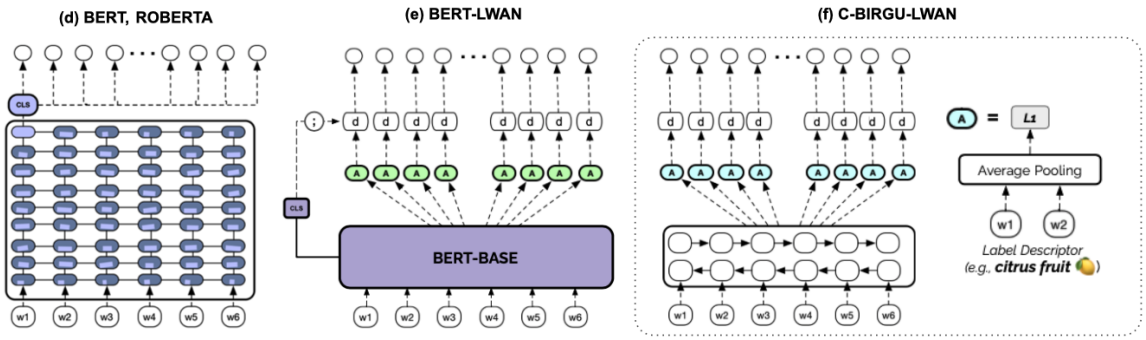


Figure 3.3: BERT, ROBERTA, BERT-LWAN and C-BIGRU-LWAN.

ATTENTION-XML: Recently, You et al. (2019) proposed a hybrid method that aims to leverage the advantages of both PLTs and neural models. Similarly to BONSAI, ATTENTION-XML uses non-binary trees. However, the classifier at each node of the PLT is now an LWAN with a BILSTM token encoder ($\mathcal{E}_w$), instead of a linear classifier operating on TF-IDF document representations.

3.5.4 Transfer learning based LMTC

BIGRU-LWAN-ELMO: In this model, we use ELMO (Peters et al., 2018) to obtain context-sensitive token embeddings, which we concatenate with the pre-trained word embeddings to obtain the initial token embeddings (w_t) of BIGRU-LWAN. Otherwise, the model is the same as BIGRU-LWAN.

BERT, ROBERTA: Following Devlin et al. (2019), we feed each document to BERT and obtain the top-level representation $h_{\texttt{[cls]}}$ of BERT's special `[cls]` token as the (single) document representation. $\mathcal{D}_d$ is now a linear layer with L outputs and sigmoid activations which operates directly on $h_{\texttt{[cls]}}$, producing a probability for each label (part (d) of Figure 3.3). The same arrangement applies to ROBERTA (Liu et al., 2019b).[4]

BERT-LWAN: Given the large size of the label set in LMTC datasets, we propose a combination of BERT and LWAN. Instead of using $h_{\texttt{[cls]}}$ as the document representation and pass it through a linear layer with L outputs (as with BERT and ROBERTA), we pass all the top-level output representations of BERT into a label-wise attention mechanism (part (e) of Figure 3.3). The final document representation (d'_l) for each label is the concatenation of $h_{\texttt{[cls]}}$ with the label-wise representation (d_l) for the corresponding label, as computed in BIGRU-LWAN ($d'_l = d_l + h_{\texttt{[cls]}}$). The entire model (BERT-LWAN) is jointly trained, also fine-tuning the underlying BERT encoder.

[4]Unlike BERT, ROBERTA uses dynamic masking, it eliminates the next sentence prediction pre-training task, and uses a larger vocabulary. Liu et al. (2019b) reported better results in NLP benchmarks using ROBERTA.

3.5.5 Zero-shot LMTC

C-BIGRU-LWAN is a zero-shot capable extension of BIGRU-LWAN. It was proposed by Rios and Kavuluru (2018), but with a CNN encoder; instead, we use a BIGRU. In this method, $\mathcal{E}_l$ creates u_l as the *centroid* of the token embeddings of the corresponding label descriptor (part (f) of Figure 3.3). The label representations u_l are then used by the attention heads.

$$v_t = \tanh(W h_t + b) \tag{3.3}$$

$$a_{lt} = \frac{\exp(v_t^\top u_l)}{\sum_{t'} \exp(v_{t'}^\top u_l)}, \quad d_l = \frac{1}{T} \sum_{t=1}^{T} a_{lt} h_t \tag{3.4}$$

Here h_t are the context-aware embeddings of $\mathcal{E}_w$, a_{lt} is the attention score of the l-th attention head for the t-th document token, viewed as v_t (Equation 3.3), and d_l is the label-wise document representation for the l-th label. $\mathcal{D}_d$ also relies on the label representations u_l to produce each label probability p_l.

$$p_l = \text{sigmoid}(u_l^\top d_l) \tag{3.5}$$

The centroid label representations u_l of both encountered (during training) and unseen (zero-shot) labels remain unchanged, because the token embeddings in the centroids are not updated. This keeps the representations of unseen labels close to those of similar labels encountered during training. In turn, this helps the attention mechanism (Equation 3.4) and the decoder (Equation 3.5) cope with unseen labels that have similar descriptors with encountered labels.

GC-BIGRU-LWAN: This model, originally proposed by Rios and Kavuluru (2018), applies graph convolutions (GCNs) to the label hierarchy.[5] The intuition is that the GCNs will help the representations of rare labels benefit from the (better) representations of more frequent labels that are nearby in the label hierarchy. $\mathcal{E}_l$ now creates graph-aware label representations u_l^3 from the corresponding label descriptors:

$$u_l^1 = f(W_s^1 u_l + \sum_{j \in N_{p,l}} \frac{W_p^1 u_j}{|N_{p,l}|} + \sum_{j \in N_{c,l}} \frac{W_c^1 u_j}{|N_{c,l}|} + b_l^1) \tag{3.6}$$

$$u_l^2 = f(W_s^2 u_l^1 + \sum_{j \in N_{p,l}} \frac{W_p^2 u_j^1}{|N_{p,l}|} + \sum_{j \in N_{c,l}} \frac{W_c^2 u_j^1}{|N_{c,l}|} + b_l^2) \tag{3.7}$$

$$u_l^3 = [u_l; u_l^2] \tag{3.8}$$

where u_l is again the centroid of the token embeddings of the descriptor of the l-th label; W_s^i, W_p^i, W_c^i are matrices for self, parent, and children nodes of each label; $N_{p,l}$, $N_{c,l}$ are the

[5] The original model uses a CNN token encoder ($\mathcal{E}_w$), whereas we use a BIGRU encoder, which is a better encoder.

sets of parents and children of the the l-th label; and f is the tanh activation. The label-wise document representations d_l are again produced by $\mathcal{E}_d$, as in C-BIGRU-LWAN (Eq. 3.3–3.4), but they go through an additional dense layer with tanh activation (Eq. 3.9). The resulting document representations $d_{l,o}$ and the graph-aware label representations u_l^3 are then used by $\mathcal{D}_d$ to produce a probability p_l for each label (Eq. 3.10).

$$d_{l,o} = \tanh(W_o d_l + b_o) \tag{3.9}$$

$$p_l = \text{sigmoid}\left((u_l^3)^\top d_{lo}\right) \tag{3.10}$$

DC-BIGRU-LWAN: The stack of GCN layers in GC-BIGRU-LWAN (Eq. 3.6–3.7) can be turned into a plain two-layer Multi-Layer Perceptron (MLP), unaware of the label hierarchy, by setting $N_{p,l} = N_{c,l} = \emptyset$. We call DC-BIGRU-LWAN the resulting (deeper than C-BIGRU-LWAN) variant of GC-BIGRU-LWAN. We use it as an ablation method to evaluate the impact of the GCN layers on performance.

DN-BIGRU-LWAN: As an alternative approach to exploiting the label hierarchy, we used a recent improvement of NODE2VEC (Grover and Leskovec, 2016) by Kotitsas et al. (2019) to obtain alternative hierarchy-aware label representations. NODE2VEC is similar to WORD2VEC (Mikolov et al., 2013b), but pre-trains node embeddings instead of word embeddings, replacing WORD2VEC's text windows by random walks on a graph (here the label hierarchy).[6] In a variant of DC-BIGRU-LWAN, dubbed DN-BIGRU-LWAN, we simply replace the initial centroid u_l label representations of DC-BIGRU-LWAN in Eq. 3.6 and 3.8 by the label representations g_l generated by the NODE2VEC extension.

DNC-BIGRU-LWAN: In another version of DC-BIGRU-LWAN, called DNC-BIGRU-LWAN, we replace the initial centroid u_l label representations of DC-BIGRU-LWAN by the concatenation $[u_l; g_l]$.

GNC-BIGRU-LWAN: Similarly, we expand GC-BIGRU-LWAN with the hierarchy-aware label representations of the NODE2VEC extension. Again, we replace the centroid u_l label representations of GC-BIGRU-LWAN in Eq. 3.6 and 3.8 by the label representations g_l of the NODE2VEC extension. The resulting model, GNC-BIGRU-LWAN, uses both NODE2VEC and the GCN layers to encode the label hierarchy, thus obtaining knowledge from the label hierarchy both in a self-supervised and a supervised fashion.

[6] The NODE2VEC extension we used also considers the textual descriptors of the nodes, using an RNN encoder operating on token embeddings.

3.6 Experimental set up

3.6.1 Evaluation measures:

Common LMTC evaluation measures are precision ($P@K$) and recall ($R@K$) at the top K predicted labels, averaged over test documents, micro-averaged F1 over all labels, and $nDCG@K$ (Manning et al., 2009). However, $P@K$ and $R@K$ unfairly penalize methods when the gold labels of a document are fewer or more than K, respectively. The macro-averaged versions of $R@K$ and $P@K$ are defined as follows:

$$R@K = \frac{1}{T}\sum_{t=1}^{T}\frac{S_t(K)}{R_t} \qquad (3.11) \qquad\qquad P@K = \frac{1}{T}\sum_{t=1}^{T}\frac{S_t(K)}{K} \qquad (3.12)$$

where T is the total number of test documents, K is the number of labels to be selected per document, $S_t(K)$ is the number of correct labels among those ranked as top K for the t-th document, and R_t is the number of gold labels for each document. Although these measures are widely used in LMTC, we question their appropriateness for the following reasons:

- $R@K$ leads to excessive penalization when documents have more than K gold labels. For example, evaluating at $K = 1$ for a single document with 5 gold labels returns $R@1 = 0.20$, if the system managed to return a correct label. The system is penalized, even though it was not allowed to return more than one label.

- $P@K$ does the same for documents with fewer than K gold labels. For example, evaluating at $K = 5$ for a single document with a single gold label returns $P@5 = 0.20$. The system is penalized, even though it did not miss any label.

- Both measures over- or under-estimate performance on documents whose number of gold labels largely diverges from K. This is illustrated in Figure 3.4 considering the slope of green ($R@K$) and red ($P@K$) lines.

Because of these drawbacks, both measures do not correctly single out the best methods. Similar concerns have led to the introduction of R-Precision (Equation 3.13) and $nDCG@K$ (Equation 3.15) in Information Retrieval (Manning et al., 2009), which we believe are also more appropriate for LMTC, leading to a more informative and fair evaluation. Note, however, that R-Precision requires the number of gold labels per document to be known beforehand, which is unrealistic in practical applications. Instead, the user will possibly define a hard threshold (K) considering how many labels can be manually

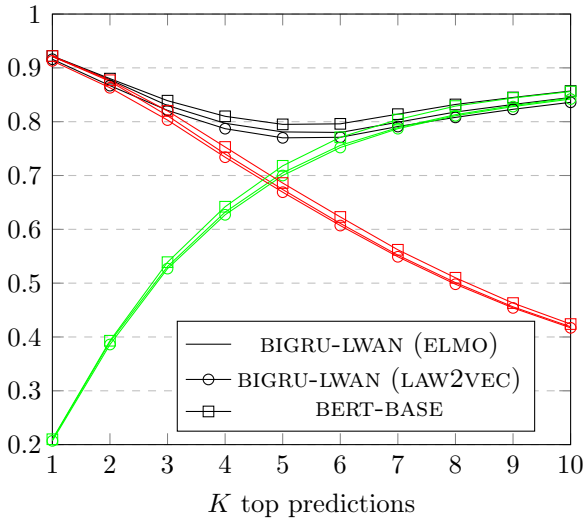


Figure 3.4: $R@K$ (green), $P@K$ (red), $RP@K$ (black) for $K \in [1, 10]$.

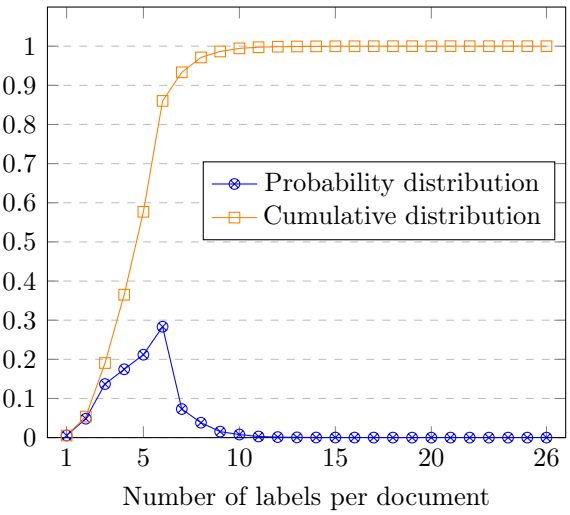


Figure 3.5: Distribution of number of labels per document in EURLEX57K.

reviewed. Therefore we propose using R-Precision@K ($RP@K$), where K is a parameter. This measure is the same as $P@K$ if there are at least K gold labels, otherwise K is reduced to the number of gold labels (Equation 3.14). The macro-averaged versions of the three measures are defined as follows:

$$RP = \frac{1}{T}\sum_{t=1}^{T}\frac{S_t(R_t)}{R_t} \qquad (3.13) \qquad RP@K = \frac{1}{T}\sum_{t=1}^{T}\frac{S_t(K)}{\min(K, R_t)} \qquad (3.14)$$

$$nDCG@K = \frac{1}{T}\sum_{t=1}^{T}\sum_{k=1}^{K}\frac{2^{S_t(k)}-1}{\log(1+k)} \qquad (3.15)$$

Again, T is the total number of test documents, K is the number of labels to be selected, $S_t(K)$ is the number of correct labels among those ranked as top K for the t-th document, and R_t is the number of gold labels for each document. Figure 3.4 shows $RP@K$ for the three best systems, macro-averaged over test documents. Unlike $P@K$, $RP@K$ does not decline sharply as K increases, because it replaces K by the number of gold labels when the latter is lower than K. For $K = 1$, $RP@K$ is equivalent to $P@K$, as confirmed by Figure 3.4. For large values of K that almost always exceed the number of gold labels, $RP@K$ asymptotically approaches $R@K$, as also confirmed by Figure 3.4. In our dataset, there are 5 labels per document on average, while the majority of the documents (57.7%) have at most 5 labels, hence $K = 5$ is reasonable. The detailed distributions can be seen in Figure 3.5. Evaluating at other values of K lead to similar conclusions.

3.6.2 Implementation details

Unless otherwise stated, we use binary cross-entropy loss, Adam (Kingma and Ba, 2015) with learning rate 0.001 and early stopping on the validation loss. For RNN-based and CNN-based methods, we select values in the following sets: number of BIGRU layers {1,2}, GRU hidden units {100, 200, 300}, batch size {8, 16}, drop-out rate {0.1, 0.2, 0.3, 0.4}. The aforementioned hyper-parameters are tuned using the HYPEROPT library selecting the values with the best loss on development data. [7] For Logistic Regression and all the BOW PLT-based (PARABEL, BONSAI), we apply stop-word filtering and use TF-IDF vectors for n-grams, where $n \in [1, 2, 3]$, grid-searching for the optimal vocabulary size of ngrams {200k, 300k, 400k}. For PLT-based methods, we use the code provided by their authors.[8]

3.7 Results

	μ_{words}	*RP*@5	*nDCG*@5
H	43	74.7	78.2
R	317	73.4	76.5
H+R	360	<u>76.5</u>	<u>79.6</u>
MB	187	64.3	67.4
Full	727	**76.6**	**79.7**

Table 3.3: BIGRU-LWAN with different document zones, as described in Section 3.4.2, on development data.

	RP@5	*nDCG*@5
GLOVE	77.1	80.1
LAW2VEC	77.5	80.4
GLOVE + ELMO	77.7	80.8
LAW2VEC + ELMO	**78.1**	**81.1**

Table 3.4: BIGRU-LWAN with GLOVE or LAW2VEC and their combination with ELMO embeddings on test data.

3.7.1 Overall predictive performance

Document length and in-domain language: In early work, we did an ablation study to measure the effect of two factors, document length, and in-domain language, that proved to be critical in our work on Information Extraction (Chapter 2). As we see in Table 3.1, the average document length in EURLEX57K is approximately up to 700 words, while there are documents up to 5,000 words. Table 3.3 compares the performance of BIGRU-LWAN on the development set for different combinations of document sections (zones) (Section 3.4.2): *header* (*H*), *recitals* (*R*), *main body* (*MB*), full text. Surprisingly *H+R* leads to almost the same results as full documents, indicating that *H+R*

[7] Our code is publicly available at `https://github.com/iliaschalkidis/lmtc-eurlex57k.git`.

[8] PARABEL: `http://manikvarma.org/code/Parabel/download.html`; BONSAI: `https://github.com/xmc-aalto/bonsai`; ATTENTION-XML: `http://github.com/yourh/AttentionXML`

provides most of the information needed to assign EUROVOC labels. Thus, we expect that TRANSFORMER-based models (BERT, ROBERTA) that are able to use up to 512 sub-word units, will not be negatively affected by using truncated versions of the documents based on their initial 512 sub-word units. These truncated versions will be on average larger than those considering *H+R* (approximately 360 words), and in many cases, they will be identical to the original texts, when the document length is less than 512 sub-word units.

In our work on Contract Element Extraction (Section 2.4), we observed that in-domain word embeddings play an important role as they capture in-domain (contractual) language. Table 3.4 shows that using WORD2VEC embeddings trained on legal texts (LAW2VEC) (Chalkidis and Kampas, 2019), including the EUR-LEX corpus, improves the performance of BIGRU-LWAN to a much smaller degree compared to GLOVE embeddings Pennington et al. (2014) and context-aware ELMO embeddings Peters et al. (2018), which both rely on pre-training on generic corpora. The empirical results highlight that the language used in EU legislation is not particularly different from the language used in the generic text (e.g., news articles, Wikipedia) captured in generic embeddings.

BIGRUs vs. CNNs: In Table 3.5, we observe that all BIGRU-based models, even BIGRU-ATT and HAN that are not specialized for LMTC tasks, outperform CNN-LWAN. This fact clearly indicates that BIGRUs is a better word encoder compared to CNNs. BIGRU-LWAN which combines BIGRUs with the label-wise attention mechanism has the best results among this group of methods.

PLTs vs. LWANs: Interestingly, the TF-IDF-based PARABEL and BONSAI outperform the best neural LWAN-based models, while being comparable to ATTENTION-XML, when all or frequent labels are considered (Table 3.5).

Effects of transfer learning: Adding context-aware ELMO embeddings to BIGRU-LWAN (BIGRU-LWAN-ELMO) improves the performance by a small margin, comparing to BIGRU-LWAN. Larger performance gains are obtained by fine-tuning BERT-BASE and ROBERTA-BASE. Our proposed new method (BERT-BASE-LWAN) that employs LWAN on top of BERT-BASE has the best results among all methods when all and frequent labels are considered. However, the results are almost comparable to BERT-BASE, indicating that the multi-head attention mechanism of BERT can effectively handle the large number of labels, similarly to LWANs and PLT-based methods, which have been specifically designed for LMTC. Considering the results for LEGAL-BERT, we observe that the effect of domain adaptation leads to minor improvements, while LEGAL-BERT-LWAN has similar results with BERT-BASE-LWAN. These findings are in line with the previous results considering generic (GLOVE) and in-domain (LAW2VEC) word embeddings.

	All Labels		Frequent ($n >= 50$)		Few ($n < 50$)	
	RP@5	*nDCG@5*	*RP@5*	*nDCG@5*	*RP@5*	*nDCG@5*
Naive baselines						
Exact Match	09.7	09.9	21.9	20.1	11.1	07.4
Logistic Regression	71.0	74.1	76.7	78.1	50.8	47.0
Flat neural methods						
bigru-att Xu et al. (2015)	75.8	78.9	79.9	81.3	63.1	58.0
han Yang et al. (2016)	74.6	77.8	78.9	80.5	59.7	54.4
cnn-lwan Mullenbach et al. (2018)	71.6	74.6	76.1	77.2	61.3	55.7
bigru-lwan (new)	77.1	80.1	81.0	82.4	65.6	61.7
bigru-lwan (law2vec) (new)	77.5	80.4	81.5	82.8	66.2	61.8
Hierarchical plt-based methods						
parabel (Prabhu et al., 2018)	78.1	80.6	82.4	83.3	59.9	57.3
bonsai (Khandagale et al., 2019)	79.3	81.8	83.4	84.3	65.0	61.6
attention-xml (You et al., 2019)	78.1	80.0	81.9	83.1	68.9	64.9
Transfer learning						
bigru-lwan-elmo (new)	78.1	81.1	82.1	83.5	66.8	61.9
bert-base (Devlin et al., 2019)	79.6	82.3	83.4	84.6	69.3	64.4
roberta-base (Liu et al., 2019b)	79.3	81.9	83.4	84.4	67.5	62.4
bert-base-lwan (new)	80.3	**82.9**	84.3	**85.4**	69.9	65.0
+ domain adaptation						
legal-bert (new)	80.0	82.6	84.0	85.0	68.9	64.6
legal-bert-lwan (new)	**80.4**	**82.9**	**84.6**	**85.4**	**70.0**	**65.1**

Table 3.5: Results (%) of experiments across base methods for all, frequent, and few label groups. All base methods are incapable of zero-shot learning. The best overall results are shown in bold. The best results in each zone are shown underlined.

	Few-shot ($n < 50$)		Zero-shot	
	RP@5	*nDCG@5*	*RP@5*	*nDCG@5*
bigru-lwan (Chalkidis et al., 2019b)	65.6	61.7	-	-
c-cnn-lwan (Rios and Kavuluru, 2018)	49.7	45.7	36.1	29.9
c-bigru-lwan (new)	55.7	51.0	46.1	33.5
dc-bigru-lwan (new)	64.3	62.1	46.2	41.5
dn-bigru-lwan (new)	56.9	52.2	34.3	23.8
dnc-bigru-lwan (new)	66.9	62.0	51.7	39.3
gc-cnn-lwan (Rios and Kavuluru, 2018)	52.3	48.4	37.1	29.6
gc-bigru-lwan (new)	66.2	61.8	**48.9**	**42.6**
gnc-bigru-lwan (new)	**67.7**	**62.6**	45.2	36.3

Table 3.6: Results (%) of experiments performed with zero-shot capable extensions of bigru-lwan. Best results shown in bold. Best results in each zone shown underlined. n is the number of training documents assigned with a label.

3.7.2 Few-shot and Zero-shot Learning

In Table 3.5, we intentionally omitted zero-shot labels, as the methods discussed so far, except gc-bigru-lwan, are incapable of zero-shot learning. In general, any model that

relies solely on trainable vectors to represent labels cannot cope with unseen labels, as it eventually learns to ignore unseen labels, i.e., it assigns them near-zero probabilities. In this section, we discuss the results of the zero-shot capable extensions of BIGRU-LWAN, that have been described in Section 3.5.5, presented in Table 3.6.

BIGRUs vs. CNNs: Similarly to the non zero-shot capable LWANs, we observe that BIGRUs is a better encoder compared to CNNs with a large impact in zero-shot capable models. Both BIGRU-LWAN and GC-BIGRU-LWAN have better performance compared to C-CNN-LWAN and GC-CNN-LWAN proposed by Rios and Kavuluru (2018). Thus we use BIGRUs in the rest of our proposed models (DC-BIGRU-LWAN, DN-BIGRU-LWAN, DNC-BIGRU-LWAN, GNC-BIGRU-LWAN).

C-BIGRU-LWAN vs. GC-BIGRU-LWAN: In line with the experiments of Rios and Kavuluru (2018), Table 3.6 shows that GC-BIGRU-LWAN (with GCNs) performs better than C-BIGRU-LWAN in zero-shot labels, while GC-BIGRU-LWAN is also comparable to BIGRU-LWAN in few-shot learning on EURLEX57K. The superior few-shot performance of BIGRU-LWAN highlights that given the proposed threshold $n = 50$, LWANs do not need zero-shot generalization and end-to-end fine-tuned models are capable of few-shot learning.

Are graph convolutions a key factor? In the work of Rios and Kavuluru (2018), it was unclear if the gains of GC-CNN-LWAN are due to the GCN encoder of the label hierarchy, or the increased depth of GC-CNN-LWAN compared to C-CNN-LWAN. Table 3.6 shows that DC-BIGRU-LWAN, which has the same depth as GC-BIGRU-LWAN, is competitive to GC-BIGRU-LWAN, indicating that the latter benefits mostly from its increased depth, and to a smaller extent from its awareness of the label hierarchy encoded by GCN layers (Equations 3.6-3.8). This motivated us to search for alternative ways to exploit the label hierarchy.

Exploiting label hierarchy with NODE2VEC: Table 3.6 shows that DN-BIGRU-LWAN, which replaces the centroids of token embeddings of the label descriptors of DC-BIGRU-LWAN with label embeddings produced by the NODE2VEC extension, is actually inferior to DC-BIGRU-LWAN. In turn, this suggests that although the NODE2VEC extension we employed aims to encode both topological information from the hierarchy and information from the label descriptors, the centroids of word embeddings still capture information from the label descriptors that the NODE2VEC extension misses. This also indicates that exploiting the information from the label descriptors is probably more important than the topological information of the label hierarchy for few and zero-shot learning generalization.

DNC-BIGRU-LWAN, which combines the centroids with the label embeddings of the NODE2VEC extension, is better than DC-BIGRU-LWAN in the few-shot group. This possibly indicates that leveraging topological information of the label hierarchy can benefit few-

shot labels that are represented in the training set, even to a small degree. This is not the case for zero-shot labels.

Combining the GCN encoder and the NODE2VEC extension (GNC-BIGRU-LWAN) leads to comparable performance with both DNC-BIGRU-LWAN and GC-BIGRU-LWAN, which encode the label hierarchy either with NODE2VEC or GCNs, in few-shot labels. Furthermore, GNC-BIGRU-LWAN has worse zero-shot performance compared to both GC-BIGRU-LWAN and DC-BIGRU-LWAN. This fact further highlights that zero generalization is heavily driven by encoding information from the label descriptors rather than topological information from the label hierarchy.

Overall, we conclude that the GCN label hierarchy encoder does not always improve LWANs in zero-shot settings, compared to equally deep LWANs, and that it may be preferable to use additional or no hierarchy-aware encodings for zero-shot learning.

	All Labels		Frequent		Few	
	RP@K	*nDCG@K*	*RP@K*	*nDCG@K*	*RP@K*	*nDCG@K*
MIMIC-III ($L_{AVG} = 15.45, K = 15$)						
BIGRU-LWAN	66.2	70.1	66.8	70.6	21.7	14.3
GC-BIGRU-LWAN (Rios and Kavuluru, 2018)	64.9	69.1	65.6	69.6	**35.9**	**21.1**
PARABEL (Prabhu et al., 2018)	58.7	63.3	59.3	63.7	9.6	6.0
BONSAI (Khandagale et al., 2019)	59.4	64.0	60.0	64.4	11.8	7.9
ATTENTION-XML (You et al., 2019)	**69.3**	**73.4**	**70.0**	**73.8**	26.9	19.5
BERT-BASE (Devlin et al., 2019)	52.7	58.1	53.2	58.4	18.2	10.0
BERT-BASE-LWAN (new)	50.1	55.2	50.6	55.5	15.3	9.1
AMAZON13K ($L_{AVG} = 5.04, K = 5$)						
BIGRU-LWAN	83.9	85.4	84.9	86.1	**80.0**	**73.6**
GC-BIGRU-LWAN (Rios and Kavuluru, 2018)	77.4	79.8	79.1	81.0	53.7	45.8
PARABEL (Prabhu et al., 2018)	85.1	86.7	86.3	87.4	76.8	71.9
BONSAI (Khandagale et al., 2019)	85.1	86.6	86.2	87.3	78.3	73.2
ATTENTION-XML (You et al., 2019)	84.9	86.7	86.0	87.4	76.0	69.7
BERT-BASE (Devlin et al., 2019)	86.8	88.5	88.5	89.6	70.3	62.2
BERT-BASE-LWAN (new)	**87.3**	**88.9**	**88.8**	**90.0**	77.2	68.9

Table 3.7: Results (%) of experiments across base methods for all, frequent, and few label groups. The best overall results are shown in bold. The best results in each zone are shown underlined. We show results for K close to the average number of labels L_{AVG}.

3.7.3 Results in other LMTC benchmark datasets

In parallel work, we also experimented with MIMIC-III (Johnson et al., 2017) and AMAZON13K (McAuley and Leskovec, 2013)) to get a broader view in LMTC and validate our findings in EURLEX57K. MIMIC-III contains approximately 52k English discharge summaries from US hospitals. Each summary has one or more codes (labels) from 8,771 leaves of the ICD-9 hierarchy, which has 8 levels (Figure 3.6). AMAZON13K contains approx.

1.5M English product descriptions from Amazon. Each product is represented by a title and a description, which are on average 250 words when concatenated. Products are classified into categories (labels) from a set of approx. 14k.

Overall performance: As we observe in Table 3.7, the overall results in AMAZON13K are similar to EURLEX57K, i.e., PLT-based methods are comparable to BIGRU-LWAN; TRANSFORMER-based methods further improve the results; and BERT-LWAN has the best results. This is not the case with MIMIC-III, where BIGRU-LWAN and ATTENTION-XML have far better results compared to TF-IDF-based PLT-based methods and TRANSFORMER-based methods. The poor performance of the two TF-IDF-based PLT-based methods on MIMIC-III seems to be due to the fact that their TF-IDF features ignore word order and are not contextualized, which is particularly important in this dataset. To confirm this, we repeated the experiments of BIGRU-LWAN on MIMIC-III after shuffling the words of the documents, and performance dropped by approx. 7.7% across all measures, matching the performance of PLT-based methods. By contrast, the drop was less significant in the other datasets (4.5% in EURLEX57K and 2.8% in AMAZON13K). The dominance of ATTENTION-XML in MIMIC-III further supports our intuition that word order is particularly important in this dataset, as the core difference of ATTENTION-XML with the rest of the PLT-based methods is the use of RNN-based classifiers that use word embeddings and are sensitive to word order, instead of linear classifiers with TF-IDF features, which do not capture word order. Meanwhile, in both EURLEX57K and AMAZON13K, the performance of ATTENTION-XML is competitive with both TF-IDF-based PLT-based methods and BIGRU-LWAN, suggesting that the bag-of-words assumption holds in these cases. Thus, we can fairly assume that word order and global context (long-term dependencies) do not play a drastic role when predicting labels (concepts) in these datasets.

Method		$\hat{T}$	$\hat{F}$	$nDCG@15$
ATTENTION-XML	(You et al., 2019)	full-text	-	**<u>73.4</u>**
BERT-BASE	(Devlin et al., 2019)	512	1.51	58.1
ROBERTA-BASE	(Liu et al., 2019b)	512	1.45	<u>58.9</u>
CLINICAL-BERT	(Alsentzer et al., 2019)	512	1.60	58.6
SCI-BERT	(Beltagy et al., 2019)	512	1.35	<u>60.5</u>
HIER-SCI-BERT	(new)	4096	1.35	<u>61.9</u>

Table 3.8: Performance of BERT and its variants compared to ATTENTION-XML on MIMIC-III. $\hat{T}$ is the maximum number of (possibly sub-word) tokens used per document. $\hat{F}$ is the fragmentation ratio, i.e., the number of tokens (BPEs or wordpieces) per word.

Poor performance of BERT on MIMIC-III: Quite surprisingly, all three BERT-based

models perform poorly on MIMIC-III (Table 3.8), so we examined two possible reasons. First, we hypothesized that this poor performance is due to the distinctive writing style and terminology of biomedical documents, which are not well represented in the generic corpora these models are pre-trained on. To check this hypothesis, we employed CLINICAL-BERT (Alsentzer et al., 2019), a version of BERT-BASE that has been further fine-tuned on biomedical documents, including discharge summaries. Table 3.8 shows that CLINICAL-BERT performs slightly better than BERT-BASE on the biomedical dataset, partly confirming our hypothesis. The improvement, however, is small and CLINICAL-BERT still performs worse than ROBERTA-BASE, which is pre-trained on larger generic corpora with a larger vocabulary. Examining the token vocabularies (Gage, 1994) of the BERT-based models reveals that biomedical terms are frequently over-fragmented; e.g., 'pneumonothorax' becomes ['p', '##ne', '##um', '##ono', '##th', '##orax'], and 'schizophreniform becomes ['s', '##chi', '##zo', '##ph', '##ren', '##iform']. This is also the case with CLINICAL-BERT, where the original vocabulary of BERT-BASE was retained. We suspect that such long sequences of meaningless sub-words are difficult to re-assemble into meaningful units, even when using deep pre-trained TRANSFORMER-based models. Thus we also report the performance of SCI-BERT (Beltagy et al., 2019), which was pre-trained from scratch (including building the vocabulary) on scientific articles, mostly from the biomedical domain. Indeed SCI-BERT performs better, but still much worse than ATTENTION-XML.

A second possible reason for the poor performance of BERT-based models on MIMIC-III is that they can process texts only up to 512 tokens long, truncating longer documents. This is not a problem in EURLEX57K, because the first 512 tokens contain enough information to classify EURLEX57K documents (Section 3.7.1). It is also not a problem in AMAZON13K, where texts are short (250 words on average). In MIMIC-III, however, the average document length is approx. 1.6k words, and documents are severely truncated. In BPEs, the average document length is approx. 2.1k, as many biomedical terms are over-fragmented, thus on average only the 1/4 of a document's text actually fits in BERT-based models. To check the effect of text truncation, we employed a hierarchical version of SCI-BERT, dubbed HIER-SCI-BERT, similar to H-BILSTM-ATT used in Obligation Extraction (Section 2.5.2). This model encodes consecutive segments of text (each up to 512 tokens) using a shared SCI-BERT encoder, then applies max-pooling over the segment encodings to produce a final document representation. HIER-SCI-BERT outperforms SCI-BERT, confirming that truncation is an important issue, but it still performs worse than ATTENTION-XML.

In another interesting direction, Kassner and Schütze (2020) recently showed that BERT struggles with both negation and misprimes, both important on event-based pre-

diction, where BERT falls short. By manually inspecting some of the discharge summaries, we observe that the hospitalization events are documented in a chronological order and a patient's medical conditions may change quickly over time during hospitalization. For example, in a single case, we see that a drug addict was admitted to the hospital for hepatitis with zero heart issues and rapidly, in a few days, ended up with a deadly heart failure. Thus, different parts of the text can potentially lead to different conclusions regarding the correct labels (topics) of the text. Also, the correct labels are mostly described periphrastically, in long quite complicated sentences. Hence MIMIC-III is potentially much more context-sensitive than both EURLEX57K and AMAZON13K, thus TRANSFORMER-based methods may fail for similar reasons, as those highlighted in Section 2.4.

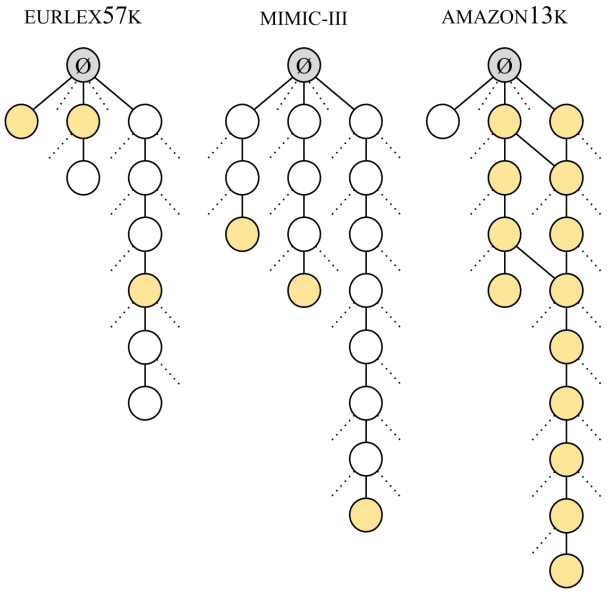


Figure 3.6: Examples from LMTC label hierarchies. ∅ is the root label. Yellow nodes denote gold label assignments. In EURLEX57K, documents have been tagged with both leaves and inner nodes. In MIMIC-III, only leaf nodes can be used. In AMAZON13K, documents are tagged with leaf nodes, but it is assumed that all the parent nodes are also assigned.

Few-shot and zero-shot learning: In MIMIC-III and AMAZON13K label hierarchies (Figure 3.6) are used with different labeling guidelines (e.g., they may require documents to be tagged only with leaf nodes, or they may allow both leaf and other nodes to be used). The latter affects graph-aware annotation proximity, i.e., the proximity of the gold labels in the label hierarchy, thus it also affects the impact of graph-aware zero-shot capable methods. When gold label assignments are dense, neighboring labels co-occur more frequently, thus models can leverage topological information and learn how to better cope with neighboring labels, which is what both GCNs and NODE2VEC do. The denser

	MIMIC-III		AMAZON13K	
	FEW ($n < 5$)	ZERO	FEW ($n < 100$)	ZERO
BIGRU-LWAN (Chalkidis et al., 2019b)	21.7	-	80.0	-
C-CNN-LWAN (Rios and Kavuluru, 2018)	21.2	37.3	8.6	19.5
C-BIGRU-LWAN (new)	26.9	52.6	13.8	29.9
DC-BIGRU-LWAN (new)	33.6	**63.9**	47.0	57.1
DN-BIGRU-LWAN (new)	19.5	43.9	27.1	36.9
DNC-BIGRU-LWAN (new)	**41.3**	59.4	50.2	59.6
GC-CNN-LWAN (Rios and Kavuluru, 2018)	23.7	38.2	41.3	45.6
GC-BIGRU-LWAN (new)	35.9	56.6	53.7	56.1
GNC-BIGRU-LWAN (new)	31.6	57.5	**53.8**	**63.4**

Table 3.9: Results (%) of experiments performed with zero-shot capable extensions of BIGRU-LWAN. Best results in each zone shown underlined. n is the number of training documents assigned with a label.

the gold label assignments, the more we can rely on more distant neighbors, and the better it becomes to include graph embedding methods that conflate larger neighborhoods, like NODE2VEC (included in GNC-BIGRU-LWAN) on AMAZON13K, when predicting unseen labels. When label assignments are sparse, as in MIMIC-III, where only non-neighboring leaf labels are assigned in the same document, leveraging the topological information (e.g., knowing that a rare label shares an ancestor with a frequent one) is not always helpful, which is why encoding the label hierarchy shows no advantage in zero-shot learning in MIMIC-III; however, it can still be useful when we have at least a few training instances, as the few-shot results of MIMIC-III indicate.

3.7.4 Attention Heat-Maps as Explanation

Attention mechanisms do not only lead to performance improvements in text classification tasks but might also provide useful evidence for the predictions (i.e., assisting in human decision-making). In Figures 3.7 and 3.8, we demonstrate such indicative results by visualizing the attention heat-maps of BIGRU-ATT and BIGRU-LWAN for EUR-LEX documents. Recall that BIGRU-LWAN uses a separate attention head per label. This allows producing multi-color heat-maps (a different color per label) separately indicating which words the system attends most when predicting each label. By contrast, BIGRU-ATT uses a single attention head and, thus, the resulting heat-maps include only one color.

We observe in practice that the multi-color heat-maps of BIGRU-LWAN are more intuitive (discrete highlighting per label) and sharp, i.e., use less, more specific words, compared to those of BIGRU-ATT. In other words, BIGRU-ATT distributes attention scores more equally across many contextualized BIGRU representations associated with input tokens, while some of those seem irrelevant or trivial. On the contrary, LWAN has more

picky attention heads, as those only target a single label. Based on these observations, we hypothesize that label-wise attention mechanisms do not suffer from information leak across tokens to the same degree as BIGRU-ATT, because the multiple attention heads of LWAN can rely on more sparse label-specific information (tokens) across the text. In general, legal topic classification can benefit from these supportive explanations, for example, consider a human actor validating (approving) the predicted labels. Unfortunately, we do not have the resources to annotate the indicative words that support gold label assignments in order to perform a formal human evaluation.

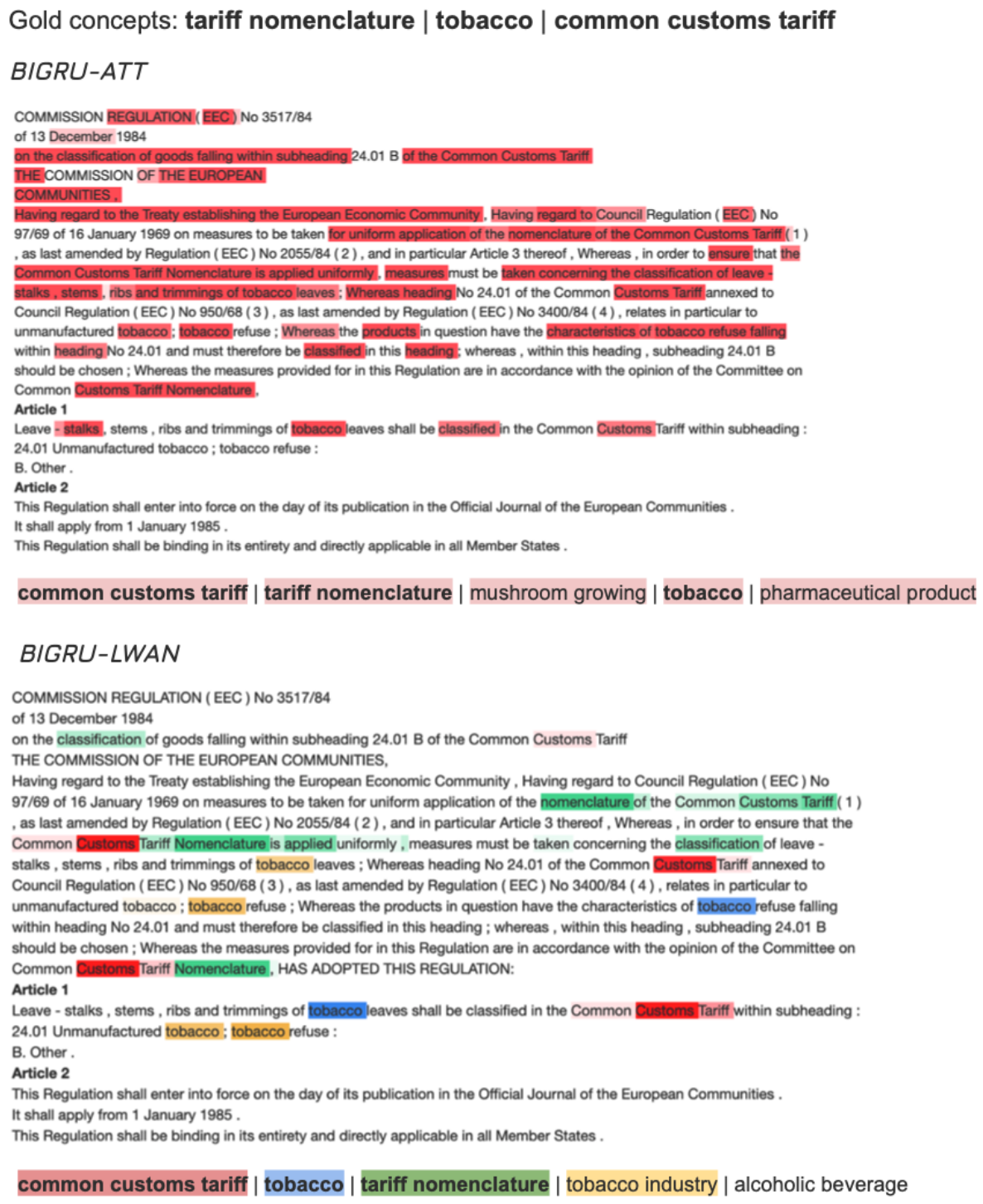


Figure 3.7: Attention heat-maps for BIGRU-ATT and BIGRU-LWAN on COMMISSION REGULATION (EEC) No 3517/84. Gold labels (concepts) are shown at the top, while the top 5 predicted labels are shown at the bottom. Correct predictions are shown in bold.

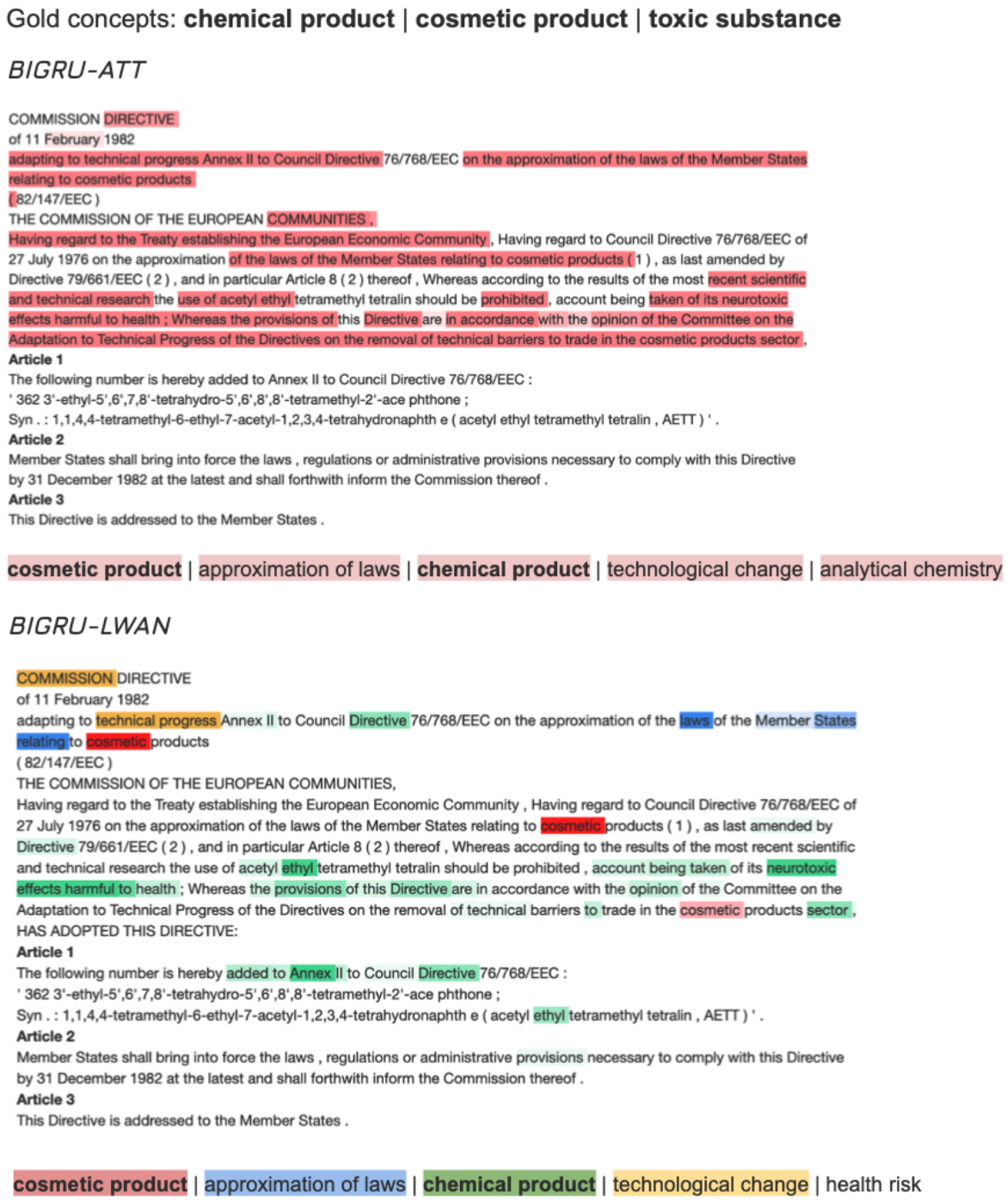


Figure 3.8: Attention heat-maps for BIGRU-ATT and BIGRU-LWAN on COMMISSION DIRECTIVE (EEC) No 82/147. Gold labels (concepts) are shown at the top, while the top 5 predicted labels are shown at the bottom. Correct predictions are shown in bold.

3.8 Conclusions

We presented an extensive study of LMTC considering the task of tagging EU legislation with EUROVOC concepts, to answer three understudied questions on (1) the competitiveness of PLT-based methods against neural models, (2) the use of the label hierarchy, and (3) the benefits from transfer learning. A condensed summary of our findings is that: (1) TF-IDF PLT-based methods are definitely worth considering, but are not always competitive, while ATTENTION-XML, a neural PLT-based method that captures word order, is robust across datasets; (2) transfer learning (via pre-trained models) leads to state-of-the-art results, especially combined with the label-wise attention mechanism. Considering datasets from other domains (MIMIC-III, AMAZON13K), we find that no single method is best across all domains and label groups (all, few, zero) as the language, the size of documents, and the label assignment strongly vary with direct implications in the performance of each method.

In future work, we would like to further exploit information from the label hierarchy in TRANSFORMER-based methods. In follow up work (Manginas et al., 2020), we found that mapping label hierarchy levels across BERT layers further improves the classification performance in EURLEX57K. In another direction, we would like to further investigate few and zero-shot learning in LMTC, especially in BERT models that are currently unable to cope with zero-shot labels. BERT uses sub-word units, instead of full tokens, which are contextualized and probably richer than typical WORD2VEC word embeddings; thus, they could benefit zero-shot capable LWANs. In a completely different set-up, one could follow the recent work of Wenpeng Yin and Roth (2019) that models zero-shot classification as a natural language inference task, considering a document-label pair at the time.

Chapter 4

Legal Judgment Prediction and Explainability

4.1 Introduction

Legal judgment prediction is the task, where given a text describing the facts of a legal case, the goal is to predict the court's outcome (Aletras et al., 2016; Şulea et al., 2017; Luo et al., 2017; Zhong et al., 2018; Hu et al., 2018). Such models may assist legal practitioners and citizens, while reducing legal costs and improving access to justice (Lawlor, 1963; Katz, 2012; Stevenson and Wagoner, 2015). Given the gravity that legal outcomes have for individuals, *explainability*, i.e., justification for a model's decision, is essential to increase the trust of both legal professionals and laypersons on system decisions and promote the use of supportive tools (Barfield, 2020). Justifying decisions is crucial for the legal domain but is also applicable in other domains (e.g., financial, biomedical), where justifications of automated decisions are essential components of both legitimacy and acceptability. Considering the above, in the legal domain we can classify the most prominent use cases into two categories:

AI-assisted judicial decision making: Lawyers and judges can use predictive models to estimate the likelihood of winning a case and come to more consistent and informed judgments, respectively (Katz, 2012).[1] In this direction, NLP models can provide valuable insights by predicting the outcome of the case considering historical data, i.e., previous cases and decisions, but also provide evidence (justification) for their predictions in the form of identifying key parts of the case facts or relevant precedent cases (case law) with respect to the facts and the legal questions.

[1]"Why a computer could help you get a fair trial", Naughton, 2017, Guardian, `https://www.theguardian.com/technology/commentisfree/2017/aug/13/why-a-computer-could-help-you-get-a-fair-trial`

Fairness and Ethics research: Human rights organizations and legal scholars can employ NLP models to scrutinize the fairness of judicial decisions unveiling if they correlate with biases (Doshi-Velez and Kim, 2017; Binns et al., 2018). NLP models can support legal scholars and researchers to faster analyze case law or even expose bias to specific types of information, i.e., a model relies, to a smaller or larger degree, on demographics (e.g., racial, gender or other characteristics) to predict the outcome, etc.

4.2 Related Work

Legal judgment prediction: Initial work on legal judgment prediction in English used linear models with features based on bags of words and topics, applying them to European Court of Human Rights (ECtHR) cases (Aletras et al., 2016; Medvedeva et al., 2018).

In all previous work, legal judgment prediction is tackled in an over-simplified experimental setup where only textual information from the cases themselves is considered, ignoring many other important factors that judges consider, more importantly, general legal argument and past case law. Also, Aletras et al. (2016) and Medvedeva et al. (2018) treat ECtHR judgment prediction as a single binary classification task per case (any article violation or not) and train linear classifiers (SVMs with TF-IDF features). In fact, the ECtHR actually considers and rules on the violation of individual articles of the European Convention of Human Rights (ECHR). In reality, the Court considers only alleged violations of particular articles, i.e., violations argued by applicants. Establishing for each case which articles are allegedly violated can be also an important preliminary task to identify an applicant's claims. Thus, in the later part of our study, we take a step back and focus on the preliminary task. While the newly introduced task could assist in legal judgment prediction, we use it as a testbed for generating paragraph-level rationales in multi-label text classification tasks.

More sophisticated neural models have been considered only in Chinese. Luo et al. (2017) use HANs (Yang et al., 2016) to encode the facts of a case and a subset of relevant law articles to predict criminal charges that have been manually annotated. In their experiments, the importance of few-shot learning (Section 3.5.5) is not taken into account since the criminal charges that appear fewer than 80 times are filtered out. However, in reality, a court is able to judge even under rare conditions. Hu et al. (2018) focused on few-shot charges prediction using a multi-task learning scenario, predicting in parallel a set of annotated discriminative attributes, which are tailored and dependent on the court, as an auxiliary task. Zhong et al. (2018) decompose the problem of charge prediction into different subtasks that are tailored to the Chinese criminal courts using multi-task learning. Similarly to the literature on legal judgment prediction for ECtHR cases, the

aforementioned approaches ignore the crucial aspect of justifying the models' predictions.

Explainability: Model *interpretability* (or *explainability*) is an emerging field of research in NLP (Lei et al., 2016; DeYoung et al., 2020; Jacovi and Goldberg, 2020). From a *model-centric* point of view, the main focus is to demystify a model's inner workings, for example targeting self-attention mechanisms (Jain and Wallace, 2019; Serrano and Smith, 2019; Wiegreffe and Pinter, 2019), and more recently Transformer-based language models (Clark et al., 2019; Kovaleva et al., 2019; Lin et al., 2019; Rogers et al., 2020).

From a *user-centric* point of view, the main focus is to study a model's reasoning in a post-hoc manner or build models that learn to provide proper justification for their decisions, similar to those of humans. Contrary to earlier work that required supervision in the form of human-annotated rationales (Zaidan et al., 2007; Zhang et al., 2016), Lei et al. (2016) introduced an *unsupervised* methodology to extract rationales (that supported aspect-based sentiment analysis predictions), i.e., gold rationale annotations were used only for evaluation. Furthermore, models were designed to produce rationales *by construction*, contrary to work studying saliency maps (generated by a model without explainability constraints) using gradients or perturbations at inference time (Ribeiro et al., 2016; Alvarez-Melis and Jaakkola, 2017; Murdoch et al., 2018). Lei et al. (2016) aimed to produce short coherent rationales that could replace the original full texts, maintaining the model's predictive performance. The rationales were extracted by generating binary masks indicating which words should be selected. To this end, Lei et al. (2016) introduced two additional loss regularizers, which penalize long rationales and sparse masks (that would select non-consecutive words).

Yu et al. (2019) proposed another constraint to ensure that the rationales would contain all the relevant information. They formulated this constraint through a *minimax* game, where two players, one using the predicted binary mask and another using the complement of this mask, aim to correctly classify the text. If the first player fails to outperform the second, the model is penalized.

Chang et al. (2019) use a Generative Adversarial Network (GAN) (Goodfellow et al., 2014), where a generator producing factual rationales competes with a generator producing counterfactual rationales to trick a discriminator. The GAN was not designed to perform classification. Given a text and a label it produces a rationale supporting the label. Hence, it can justify the label prediction of a classifier in a *post-hoc* manner.

Jain et al. (2020) decoupled the model's predictor from the rationale extractor to produce 'faithful' explanations, ensuring that the predictor considers only the rationales and not other parts of the text. Faithfulness refers to how accurately an explanation reflects the true reasoning of a model (Lipton, 2018; Jacovi and Goldberg, 2020).

All the aforementioned work conceives rationales as selections of words, targeting bi-

nary classification tasks even when this is inappropriate. For instance, DeYoung et al. (2020) and Jain et al. (2020) over-simplified the task of the multi-passage reading comprehension (MultiRC) dataset (Khashabi et al., 2018) turning it into a binary classification task with word-level rationales, while sentence-level rationales seem more suitable. By contrast, we focus on paragraph-level rationale extraction in a realistic multi-label setting of alleged violation prediction.

Responsible AI: Our work complies with the ECtHR data policy.[2] By no means, we aim to build a 'robot' lawyer or judge and we acknowledge the plausible harmful impacts (Angwin et al., 2016; Dressel and Farid, 2018) of irresponsible deployment. Instead, we support fair and explainable AI-assisted judicial decision making and empirical legal studies considering critical research on responsible AI (Irani et al., 2021) that aim to provide explainable, and fair supportive systems.

4.3 Contributions

- We release two new datasets including 11,000 ECtHR cases. For reference, the dataset published by Aletras et al. (2016) includes 600 cases, while a more recent dataset published by Medvedeva et al. (2018) includes approximately 2,500 cases. The first dataset supports multi-label judgment prediction, not considered in previous work. We also introduce the task of allegation prediction, a supportive task for judgment prediction, where models have to identify alleged violations that can be raised by plaintiffs, in our case in ECHR cases. For this reason, we publish another second dataset of equal size, which also includes annotations with allegation prediction rationales from both the court judges and lawyers.

- We propose the use of the Hierarchical Attention Network (Yang et al., 2016) to capture the document structure of ECHR cases. Further on, we proposed a hierarchical variation of BERT (HIER-BERT). While both methods have been recently used by several researchers in various classification tasks, we were among the first who proposed and used hierarchical neural networks.

- We study model bias by investigating how sensitive our models are to demographic information (e.g., ethnicity, locations, etc.) appearing in the facts of a case.

- We study explainability in the legal domain, introducing a new legal task accompanied by paragraph-level rationales. While we adopt and compare various rationale

[2] https://www.echr.coe.int/Pages/home.aspx?p=privacy

constraints, in the form of regularizers, from the literature in the new paragraph-level setting, we also propose new rationale constraints that produce state-of-the-art results by improving rationale quality, while also improving faithfulness.

4.4 Legal Judgment Prediction

4.4.1 ECtHR Dataset

ECtHR hears allegations that a state has breached human rights provisions of the European Convention of Human Rights (ECHR).[3] Our dataset contains approximately 11.5k cases from ECHR's public database.[4] For each case, the dataset provides a list of *facts* extracted using regular expressions from the case description, as in Aletras et al. (2016).[5] Each case is also related to *articles* of the Convention that were violated (if any). An *importance score* is also assigned by ECHR signifying the importance of this case with respect to case law (see Section 4.4.2). The dataset is split into training, development, and test sets (Table 4.1). The training and development sets contain cases from 1959 through 2013, and the test set from 2014 through 2018.

Subset	Cases (C)	Avg. words/C	Avg. facts/C	Avg. violated articles/C
Train	7,100	2,421	43	0.71
Dev.	1,380	1,931	30	0.96
Test	2,998	2,588	45	0.71

Table 4.1: Statistics of the ECtHR dataset. The size of the label set (ECHR articles) per case (C) is $L = 66$.

4.4.2 Legal Prediction Tasks

We consider three tasks in the legal judgment prediction part of this chapter:

Binary Violation Given the facts of a case, we aim to classify it as positive if *any* human rights article or protocol has been violated and negative otherwise.

Multi-label Violation Similarly, the second task is to predict which specific human rights articles have been violated (if any). The total number of articles and protocols of

[3] An up-to-date copy of the European Convention of Human Rights is available at `https://www.echr.coe.int/Documents/Convention_ENG.pdf`.

[4] See `https://hudoc.echr.coe.int`.

[5] Using regular expressions to segment legal text from ECHR is usually trivial, as the text has a specific structure. An example can be found at `http://hudoc.echr.coe.int/eng?i=001-193071`.

the European Convention of Human Rights are currently 66. For that purpose, we define a multi-label classification task where no labels are assigned when there is no violation.

Case Importance We also predict the importance of a case on a scale from 1 (key case) to 4 (unimportant) in a regression task. These scores, provided by the ECHR, denote a case's contribution to the development of case-law allowing legal practitioners to identify pivotal cases. Overall in the dataset, the scores are: 1 (1096 documents), 2 (904), 3 (2,982) and 4 (6,496), indicating that approx. 10% are landmark cases, while the vast majority (83%) are considered more or less unimportant for further review.

4.4.3 Methods

BIGRU-ATT: The first model is a BIGRU with self-attention (Xu et al., 2015) where the facts of a case are concatenated into a word sequence. This model is identical to the one used in Chapter 3. Words are mapped to embeddings and passed through a stack of BIGRUs. A single case embedding (h) is computed as the sum of the resulting context-aware embeddings ($\sum_i a_i h_i$) weighted by self-attention scores (a_i). The case embedding (h) is passed to the output layer using a sigmoid for binary violation, softmax for multi-label violation, or no activation for case importance regression.

HAN: The Hierarchical Attention Network (Yang et al., 2016) is a strong baseline for text classification, already presented in Chapter 3 for legal topic classification. First, a BIGRU with self-attention reads the words of each fact, as in BIGRU-ATT, producing fact embeddings. A second-level BIGRU with self-attention reads the fact embeddings, producing a case embedding that goes through a similar output layer as in BIGRU-ATT.

BERT: For each task, a task-specific layer has to be added on top of BERT (Devlin et al., 2019) and trained jointly by fine-tuning on task-specific data, as we already presented in Chapters 2 and 3. We add a linear layer on top of BERT, with a sigmoid, softmax, or no activation, for binary violation, multi-label violation, and case importance, respectively. BERT can process texts up to 512 wordpieces, whereas our case descriptions are up to 2.6k words, thus we truncate them to BERT's maximum length, which affects its performance. This also highlights an important limitation of BERT in processing long documents, a common characteristic in legal text processing, as previously mentioned in Chapter 3.

HIER-BERT: To surpass BERT's maximum length limitation, we also propose a hierarchical version of BERT (HIER-BERT). Firstly BERT-BASE reads the words of each fact,

producing fact embeddings. Then a self-attention mechanism reads fact embeddings, producing a single case embedding that goes through a similar output layer as in HAN.

4.4.4 Experiments

Experimental Setup

Hyper-parameters: We use pre-trained GLOVE (Pennington et al., 2014) embeddings ($d = 200$) for all experiments. Hyper-parameters are tuned by random sampling 50 combinations and selecting the values with the best development loss in each task in the following sets: GRU hidden units {200, 300, 400}, number of stacked BIGRU layers {1, 2}, batch size {8, 12, 16}, and dropout rate {0.1, 0.2, 0.3, 0.4}. Given the best hyper-parameters, we perform five runs for each model reporting mean scores and standard deviations on the test set. We use categorical cross-entropy loss for the classification tasks and mean absolute error for the regression task, Glorot initialization (Glorot and Bengio, 2010), Adam (Kingma and Ba, 2015) with default learning rate 0.001, and early stopping on the development loss. For BERT-based methods, we set the dropout rate to 0.1 and grid-search for learning rate {2e-5, 3e-5, 4e-5, 5e-5}, as suggested by Devlin et al. (2019). In the case of HIER-BERT, we were able to use up to 25 paragraphs of 128 words.

Baselines: A majority-class (MAJORITY) classifier is used in binary violation, where all cases are auto-classified as positives (violation, the majority class in the training set); for case importance, we assign to all cases the most frequent score (4) across cases. A second baseline (COIN-TOSS) randomly predicts violation or not in the binary violation task. We also compare our methods against a linear SVM with bag-of-words features (most frequent [1, 5]-grams across all training cases weighted by TF-IDF), dubbed BOW-SVM, similar to Aletras et al. (2016) and Medvedeva et al. (2018) for the binary task, multiple one-vs-rest classifiers for the multi-label task, and Support Vector Regression (BOW-SVR) for the case importance prediction.[6]

Binary Violation Results: Table 4.2 (upper part) shows the results for binary violation. We evaluate models using macro-averaged precision (P), recall (P), F1. The weak baselines (MAJORITY, COIN-TOSS) are widely outperformed by the rest of the methods. BIGRU-ATT outperforms in F1 (79.5 vs. 71.8) the previous best performing method, BOW-SVM, in English judicial prediction (Aletras et al., 2016). This is aligned with results in Chinese (Luo et al., 2017; Zhong et al., 2018; Hu et al., 2018). HAN slightly improves

[6]We tune the hyper-parameters of BOW-SVM/SVR and select kernel (RBF, linear) with a grid search on the development set.

	P	R	F1
MAJORITY	32.9 ± 0.0	50.0 ± 0.0	39.7 ± 0.0
COIN-TOSS	50.4 ± 0.7	50.5 ± 0.8	49.1 ± 0.7
Non-Anonymized			
BOW-SVM	71.5 ± 0.0	72.0 ± 0.0	71.8 ± 0.0
BIGRU-ATT	87.1 ± 1.0	77.2 ± 3.4	79.5 ± 2.7
HAN	88.2 ± 0.4	78.0 ± 0.2	80.5 ± 0.2
BERT	24.0 ± 0.2	50.0 ± 0.0	17.0 ± 0.5
HIER-BERT	**90.4** ± 0.3	**79.3** ± 0.9	**82.0** ± 0.9
Anonymized			
BOW-SVM	71.6 ± 0.0	70.5 ± 0.0	70.9 ± 0.0
BIGRU-ATT	**87.0** ± 1.0	76.6 ± 1.9	78.9 ± 1.9
HAN	85.2 ± 4.9	**78.3** ± 2.0	**80.2** ± 2.7
BERT	17.0 ± 3.0	50.0 ± 0.0	25.4 ± 0.4
HIER-BERT	85.2 ± 0.3	78.1 ± 1.3	80.1 ± 1.1

Table 4.2: Macro precision (P), recall (R), F1 for the **binary violation** prediction task (± std. dev.).

over BIGRU-ATT (80.5 vs. 79.5), while being more robust across runs (0.2% vs. 2.7% std. dev.). BERT's poor performance is due to the truncation of case descriptions, as indicated by the fact that HIER-BERT, which uses the full text of the case, leads to the best results. We omit BERT from the following tables, since it performs poorly.

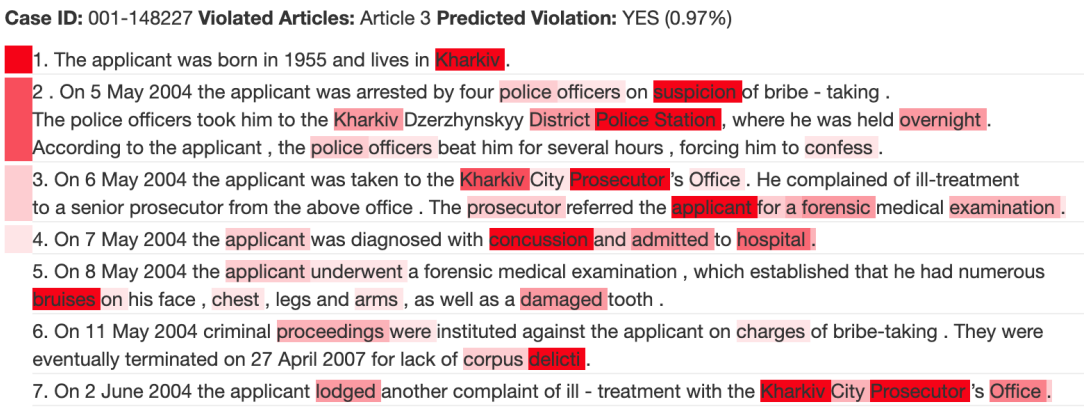


Figure 4.1: Attention over words (colored words) and facts (vertical heat bars) as produced by HAN.

Figure 4.1 shows the attention scores over words and facts of HAN for a case that ECHR found to violate Article 3, which prohibits torture and 'inhuman or degrading treatment or punishment'. Although fact-level attention wrongly assigns high attention to the first fact, which seems irrelevant, it then successfully focuses on facts 2–4, which report that police officers beat the applicant for several hours, that the applicant complained, was referred for forensic examination, diagnosed with concussion, etc. Word attention also successfully focuses on words like 'concussion', 'bruises', 'damaged', but it also highlights

OVERALL (all labels)			
	P	R	F1
BOW-SVM	56.3 ± 0.0	45.5 ± 0.0	50.4 ± 0.0
BIGRU-ATT	62.6 ± 1.2	50.9 ± 1.5	56.2 ± 1.3
HAN	65.0 ± 0.4	**55.5** ± 0.7	59.9 ± 0.5
HIER-BERT	**65.9** ± 1.4	55.1 ± 3.2	**60.0** ± 1.3
FREQUENT (≥50)			
BOW-SVM	56.3 ± 0.0	45.6 ± 0.0	50.4 ± 0.0
BIGRU-ATT	62.7 ± 1.2	52.2 ± 1.6	57.0 ± 1.4
HAN	65.1 ± 0.3	**57.0** ± 0.8	**60.8** ± 1.3
HIER-BERT	**66.0** ± 1.4	56.5 ± 3.3	**60.8** ± 1.3
FEW ([1,50))			
BOW-SVM	-	-	-
BIGRU-ATT	36.3 ± 13.8	03.2 ± 23.1	05.6 ± 03.8
HAN	30.2 ± 35.1	01.6 ± 01.2	02.8 ± 01.9
HIER-BERT	**43.6** ± 14.5	**05.0** ± 02.8	**08.9** ± 04.9

Table 4.3: Micro precision, recall, F1 in **multi-label violation** for all, frequent, and few training instances.

entities like 'Kharkiv', its 'District Police Station', and 'City Prosecutor's office', which may be indications of bias.

Models Biases: We next investigate how sensitive our models are to demographic information appearing in the facts of a case. Our assumption is that an unbiased model should not rely on information about nationality, gender, age, etc. To test the sensitivity of our models to such information, we train and evaluate them in an anonymized version of the dataset. The data is anonymized by using SPACY's (`https://spacy.io`) Named Entity Recognizer, replacing all recognized entities with type tags (e.g., 'Kharkiv' → LOCATION). While neural methods seem to exploit named entities among other information, as in Figure 4.1, the results in Table 4.2 indicate that performance is comparable even when this information is masked, with the exception of HIER-BERT that has quite worse results (2%) compared to using non-anonymized data, suggesting a mild model bias. We speculate that HIER-BERT is more prone to over-fitting compared to the other neural methods that rely on frozen GLOVE embeddings because the embeddings of BERT's sub-word embeddings are trainable and thus can freely adjust to the vocabulary of the training documents including demographic information. The results of BERT are improved on anonymized data because the anonymization causes longer text to fit in.

	MAE	SPEARMAN'S ρ
MAJORITY	**.369** ± .000	*N/A**
BOW-SVR	.585 ± .000	.370 ± .000
BIGRU-ATT	.539 ± .073	.459 ± .034
HAN	.524 ± .049	.437 ± .018
HIER-BERT	.437 ± .018	**.527** ± .024

Table 4.4: Mean Absolute Error and Spearman's ρ for **case importance**. Importance ranges from 1 (most important) to 4 (least). * Not Applicable.

Multi-label Violation Results:

Table 4.3 reports micro-averaged precision (P), recall (R), and F1 results for all methods in multi-label violation prediction. The results are also grouped by label frequency for all (OVERALL), FREQUENT, and FEW labels (articles), counting frequencies on the training subset. We observe that predicting specific articles that have been violated is a much more difficult task than predicting if *any* article has been violated in a binary setup (cf. Table 4.2). Overall, HIER-BERT outperforms BIGRU-ATT, while being comparable with HAN (60.0 vs. 59.9 micro-F1). All models under-perform in labels with FEW training examples, demonstrating the difficulty of few-shot learning in ECHR legal judgment prediction. The main reason is that labels in the FEW group, 11 in total, are extremely rare and have been assigned in 1.25% of the documents across all datasets, while the most frequent 4 labels overall (Articles 3, 5, 6 and 13) have been assigned in approx. 42% of the documents.

Case Importance Results: Table 4.4 shows the mean absolute error (MAE) obtained when predicting case importance. Surprisingly, MAJORITY outperforms the rest of the methods. As already noted in Section 4.4.2, the distribution of importance scores is highly skewed in favour of the majority class, thus MAJORITY can correctly predict the score in most cases cases, which leads to a lower mean absolute error (MAE), comparing to the other methods. BOW-SVR performs worse than BIGRU-ATT, while HAN is slightly better compared to the BIGRU-ATT. HIER-BERT further improves the results, outperforming HAN and the rest of the methods.

While MAJORITY has the lowest mean absolute error, it cannot distinguish important from unimportant cases, thus it is practically useless. To evaluate the methods on that matter, we measure the correlation between the gold scores and each method's predictions with SPEARMAN'S ρ. HIER-BERT has the best ρ (.527), indicating a moderate positive correlation (> 0.5), which is not the case for the rest of the methods. The overall results indicate that a case's importance cannot be predicted solely by the case facts and possibly

also relies on background knowledge (e.g., judges' experience, court's history, rarity of article's violation).

Discussion: We can only speculate that HAN's fact embeddings distill importance-related features from each fact, allowing its second-level GRU to operate on a sequence of fact embeddings that are being exploited by the fact-level attention mechanism and provide a more concise view of the entire case. The same applies to HIER-BERT, which relies on BERT's fact embeddings and the same fact-level attention mechanism. By contrast, BIGRU-ATT operates on a single long sequence of concatenated facts, making it more difficult for its BIGRU to combine information from multiple, especially distant, facts. This may explain the good performance of HAN and HIER-BERT across all tasks.

	BINARY			MULTI-LABEL		
	P	R	F1	P	R	F1
HIER-BERT	**90.4** ± 0.3	79.3 ± 0.9	82.0 ± 0.9	**65.9** ± 1.4	55.1 ± 3.3	60.0 ± 1.3
HIER-LEGAL-BERT	85.7 ± 0.7	**80.8** ± 0.5	**82.4** ± 0.4	65.4 ± 1.2	**59.3** ± 2.5	**62.2** ± 1.5

Table 4.5: Results for HIER-LEGAL-BERT in **binary** and **multi-label** violation.

Domain adaptation Table 4.5 shows the results of both HIER-BERT and HIER-LEGAL-BERT in binary and multi-label violation prediction. We observe small differences in performance on the binary classification task (0.4% improvement). On the contrary, we observe a more substantial improvement in the more difficult multi-label task (1.4%) indicating that the LEGAL-BERT model benefits from in-domain knowledge.

4.5 Allegation Prediction and Rationale Extraction

4.5.1 Explaining model decisions

While legal judgment prediction has been studied in the past for cases ruled by the European Court of Human Rights (Aletras et al., 2016; Medvedeva et al., 2018), including our work (see Section 4.4), and for Chinese criminal court cases (Luo et al., 2017; Hu et al., 2018; Zhong et al., 2018), there is no precedent of work investigating the justification of the models' decisions. Reviewing attention heat-maps in our initial experiments on legal judgment prediction (Section 4.4 and Figure 4.1), we observed several limitations. First of all, providing word-level saliency maps is not particularly helpful from a practical perspective for legal practitioners given the complexity of the task and document length; thus extracting explanation on paragraph level seems a better alternative. Further on, based on the current paragraph-level saliency maps we observe that the reasoning of the

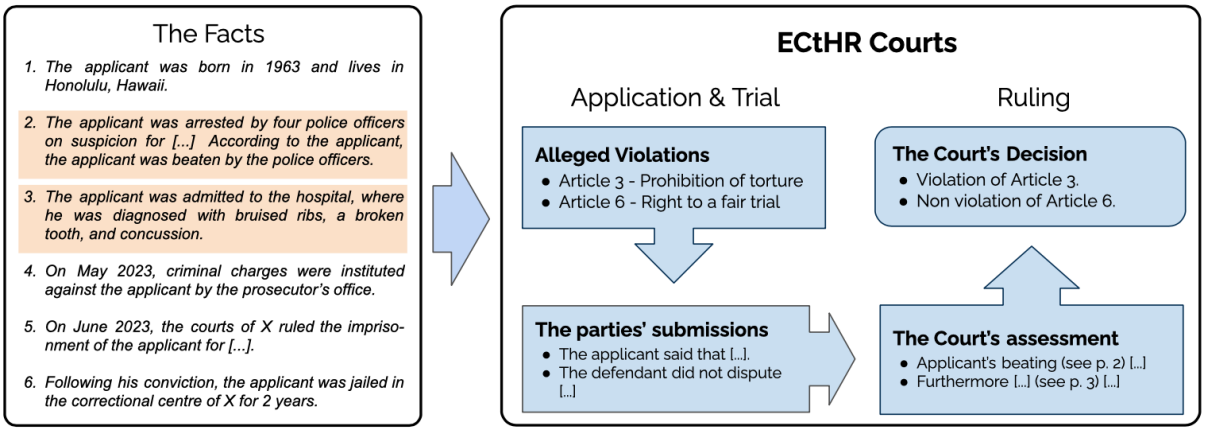


Figure 4.2: A depiction of the ECtHR process: With respect to the facts, the applicant(s) (plaintiff(s)) request a hearing from ECtHR considering specific accusations (alleged violations of ECHR articles) against the defendant state(s). The Court (panel of judges) access the facts and the rest of the parties' submission, and rules its decision for the violation or not of the allegedly violated ECHR articles. Prominent facts (rationales) are highlighted.

current models is far from human reasoning, i.e., high attention in trivial paragraphs or paragraphs including only demographic information. The use of saliency maps as an explanation has been criticized in recent literature (Wiegreffe and Pinter, 2019), raising important questions on the faithfulness (Jacovi and Goldberg, 2020) of the produced explanations, while recent studies (Lei et al., 2016; Yu et al., 2019; Jain et al., 2020) focus on methods that aim to extract faithful explanations by-design.

Given the importance of the legal domain, similarly to other domains (e.g., financial, biomedical), explainability is a key feature that may potentially improve the trustworthiness of systems. Thus, we investigate the explainability of the decisions of state-of-the-art models that aim to extract rationales by-design. To formally evaluate the rationale quality of these models, we compare the rationale paragraphs they select to those of legal professionals, both litigants and lawyers, in the much simpler task of alleged violation prediction that can be resolved given only the text of the facts of each case, as explained by legal experts with experience in ECtHR literature. By contrast, legal judgment prediction vastly relies on case law and subjective interpretation of facts.

The new classification task (*alleged violation prediction*) and its justification (rationale extraction) can be used as a part of a broader system that aims to assist the judicial process (Figure 4.2). The most prominent use-cases, as highlighted by the legal experts, are the following: (a) support legal scholars/researchers to faster analyze case law (Katz, 2012); (b) AI-assisted judicial decision making, i.e., more informed judicial decision making (Zhong et al., 2020; Naughton, 2017); and (c) support applicants to prepare their case by identifying alleged violations that fit their case (facts).

	Cases (C)	Sparsity	#Allegations
Train	9K	24%	1.8
Dev,	1K	30%	1.7
Test	1K	31%	1.7

Table 4.6: Statistics of the new ECtHR dataset. 'Sparisty' is the average percentage of paragraphs included in the silver rationales. '#Allegations' is the average number of allegedly violated articles.

4.5.2 The new augmented ECtHR Dataset

The court (ECtHR) hears allegations regarding breaches in human rights provisions of the European Convention of Human Rights (ECHR) by European states (Figure 4.2).[7] The court rules on a subset of all ECHR articles, which are predefined (alleged) by the applicants (*plaintiffs*). Our dataset comprises 11k ECtHR cases and can be viewed as an enriched version of the ECtHR dataset presented in the previous section, which did not provide ground truth for alleged article violations (articles discussed) and rationales (Table 4.6). The new dataset includes the following:

Facts: Each judgment includes a list of paragraphs that represent the facts of the case, i.e., that describe the main events that are relevant to the case, in numbered paragraphs. We hereafter call these paragraphs *facts* for simplicity. Note that the facts are presented in chronological order. Not all facts have the same impact or hold crucial information with respect to alleged article violations and the court's assessment; i.e., facts may refer to information that is trivial or otherwise irrelevant to the legally crucial allegations against *defendant* states.

Allegedly violated articles: Judges rule on specific accusations (allegations) made by the applicants (Harris, 2018). In ECtHR cases, the judges discuss and rule on the violation, or not, of specific articles of the Convention. The articles to be discussed (and ruled on) are put forward (as alleged article violations) by the applicants and are included in the dataset as ground truth. We identify 40 violable articles in total.[8] In our experiments, however, the models are not aware of the allegations. They predict the Convention articles that will be discussed (the allegations) based on the case's facts, and they also produce rationales for their predictions. Models of this kind could be used by potential applicants to help them formulate future allegations (articles they could claim to have been violated) based on case facts, but here we mainly use the task as a test-bed for rationale extraction.

[7] The Convention is available at `https://www.echr.coe.int/Documents/Convention_ENG.pdf`.

[8] The rest of the articles are procedural, i.e., the number of judges, criteria for office, election of judges, etc.

Violated articles: The court decides which allegedly violated articles have indeed been violated. These decisions are also included in our dataset and could be used for legal judgment prediction experiments. However, they are not used in the experiments of this section, as we did in the previous one.

Silver allegation rationales: Each decision of the ECtHR includes references to facts of the case (e.g., "*See paragraphs 2 and 4.*") and case law (e.g., "*See Draci vs. Russia (2010).*"). We identified references to each case's facts and retrieved the corresponding paragraphs using regular expressions. These are included in the dataset as silver allegation rationales, on the grounds that the judges refer to these paragraphs when ruling on the allegations.

Gold allegation rationales: A legal expert with experience in ECHR cases annotated a subset of 50 test cases to identify the relevant facts (paragraphs) of the case that support the allegations (alleged article violations). In other words, each identified fact justifies (hints) one or more alleged violations.

Task definition: In this work, we investigate *alleged violation prediction*, a multi-label text classification task where, given the facts of a ECtHR case, a model predicts which articles, among 40 ECHR articles, were alleged by the applicant(s). In addition, a model needs to identify which facts are most prominent to assist the classification task.

4.5.3 Methods

We first describe a baseline model that we use as our starting point. It adopts the framework proposed by Lei et al. (2016), which generates rationales by construction in two steps: first a *rationale extraction* sub-network produces a binary mask indicating the most important words of the text; and subsequently a *prediction* sub-network classifies a hard-masked version of the text. We then discuss additional constraints that have been proposed to improve word-level rationales, which can be added to the baseline as regularizers. We argue that some of them are not appropriate for paragraph-level rationales and multi-label classification tasks. We also consider variants of previous constraints and introduce a new one.

Baseline Model

Our baseline is a hierarchical variation of BERT (Devlin et al., 2019), dubbed HIERBERT-HA, which is similar to HIER-BERT described in Section 4.4.3, although it uses hard attention instead of (soft) self-attention. Each case (document) D is viewed as a list of facts (paragraphs) $D = [P_1, \ldots, P_N]$. Each paragraph is a list of tokens $P_i = [w_1, \ldots, w_{L_i}]$. We first pass each paragraph independently through a shared BERT encoder (Figure 4.3)

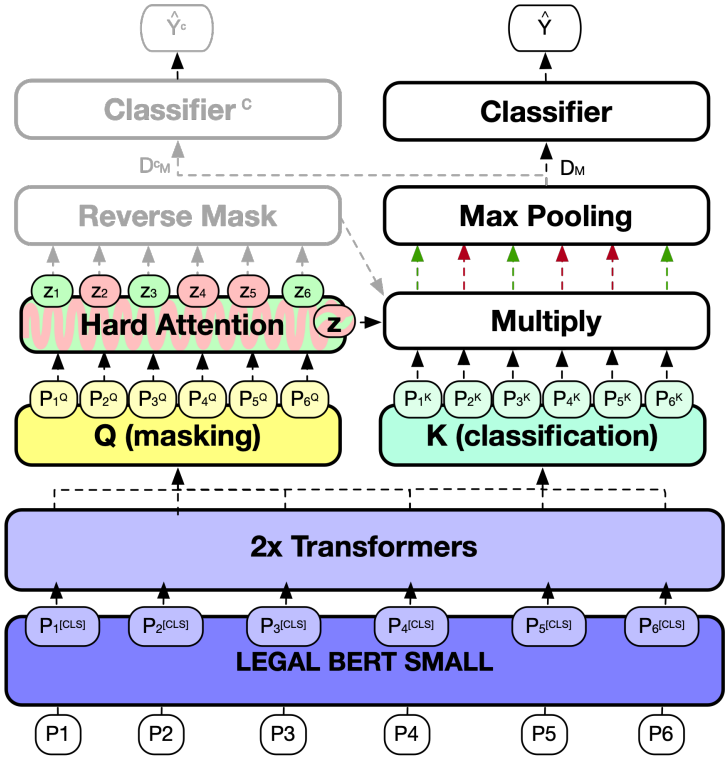


Figure 4.3: Illustration of HIERBERT-HA. The shaded parts operate only when when L_g (comprehensiveness loss) or L_r (sigularity loss) are used.

to extract context-unaware paragraph representations $P_i^{\texttt{[cls]}}$, using the `[cls]` embedding of BERT. Then, a shallow encoder with two Transformer layers (Vaswani et al., 2017) produces contextualized paragraph embeddings, which are in turn projected to two separate spaces by two different fully-connected layers, K and Q, with SELU activations (Klambauer et al., 2017). K produces the paragraph encoding P_i^K, to be used for classification; and Q produces the paragraph encoding P_i^Q, to be used for rationale extraction. The rationale extraction sub-network passes each P_i^Q encoding independently through a fully-connected layer with a sigmoid activation to produce soft attention scores $a_i \in [0, 1]$. The attention scores are then binarized using a 0.5 threshold, leading to hard attention scores z_i ($z_i = 1$ iff $a_i > 0.5$). The hard-masked document representation D_M is obtained by hard-masking paragraphs and max-pooling:

$$D_M = \text{maxpool}\big([z_1 \cdot P_1^K, \ldots, z_N \cdot P_N^K]\big)$$

D_M is then fed to a dense layer with sigmoid activations, which produces a probability estimate per label, $\widehat{Y} = [\hat{y}_1, \ldots, \hat{y}_{|A|}]$, in our case per article of the Convention, where $|A|$ is the size of the label set. For comparison, we also experiment with a model that masks no facts, dubbed HIERBERT-ALL.

The thresholding that produces the hard (binary) masks z_i is not differentiable. To address this problem, Lei et al. (2016) used reinforcement learning (Williams, 1992),

while Bastings et al. (2019) proposed a differentiable mechanism relying on the re-parameterization trick (Louizos and Welling, 2017). We follow a simpler trick, originally proposed by Chang et al. (2019), where during back-propagation the thresholding is detached from the computation graph, allowing the gradients to bypass the thresholding and reach directly the soft attentions a_i.

Rationale Constraints as Regularizers

Sparsity: Modifying the word-level sparsity constraint of Lei et al. (2016) for our paragraph-level rationales, we hypothesize that good rationales include a small number of facts (paragraphs) that sufficiently justify the allegations; the other facts are trivial or secondary. For instance, an introductory fact like "*The applicant was born in 1984 and lives in Switzerland.*" does not support any allegation, while a fact like "*The applicant contended that he had been beaten by police officers immediately after his arrest and later during police questioning.*" suggests a violation of Article 3 "Prohibition of Torture". Hence, we use a *sparsity* loss to control the number of selected facts:

$$L_s = \left| \frac{1}{N} \sum_{i=1}^{N} z_i - T \right| \tag{4.1}$$

where T is a predefined threshold expressing the percentage (e.g., 20%) of selected facts per case. For example, we can approximate T considering silver rationales (Table 4.6).

Continuity: In their work on word-level rationales, Lei et al. (2016) also required the selected words to be *contiguous*, to obtain more coherent rationales. In other words, the transitions between selected ($z_i = 1$) and not selected ($z_i = 0$) words in the hard mask should be minimized. This is achieved by adding the following *continuity* loss:

$$L_c = \frac{1}{N-1} \sum_{i=2}^{N} |z_i - z_{i-1}| \tag{4.2}$$

In paragraph-level rationale extraction, where entire paragraphs are masked, the continuity loss forces the model to select contiguous paragraphs. In ECtHR cases, however, the facts are self-contained and internally coherent paragraphs (or single sentences). The facts are usually ordered chronologically, but, apart from temporal order, there are usually no other clear inter-paragraph discourse relations. Hence, we hypothesize that the *continuity* loss is inappropriate in our case. Nonetheless, we empirically investigate its effect.

Comprehensiveness: We also adapt the *comprehensiveness* loss of Yu et al. (2019), which was introduced to force the hard mask $Z = [z_1, \ldots, z_N]$ to (ideally) keep *all* the words (in our case, paragraphs about facts) of the document D that support the correct

decision Y. In our task, $Y = [y_1, \ldots, y_{|A|}]$ is a binary vector indicating the Convention articles the court discussed (gold allegations) in the case of D. Intuitively, the complement Z^c of Z, i.e., the hard mask that selects the words (in our case, facts) that Z does not select, should not select sufficient information to predict Y. Given D, let D_M, D_M^c be the representations of D obtained with Z, Z^c, respectively; let $\widehat{Y}, \widehat{Y}^c$ be the corresponding probability estimates; let L_p, L_p^c be the classification loss, typically total binary cross-entropy, measuring how far $\widehat{Y}, \widehat{Y}^c$ are from Y. In its original form, the comprehensiveness loss requires L_p^c to exceed L_p by a margin h.

$$L_g = \max(L_p - L_p^c + h, 0) \tag{4.3}$$

While this formulation may be adequate in binary classification tasks, in multi-label classification it is very hard to pre-select a reasonable margin, given that cross-entropy is unbounded, that the distribution of true labels (articles discussed) is highly skewed, and that some labels are easier to predict than others. To make the selection of h more intuitive, we propose a reformulation of L_g that operates on class probabilities rather than classification losses. The Equation 4.3 becomes:

$$L_g = \frac{1}{|A|}\sum_{i=1}^{|A|} y_i(\hat{y}_i^c - \hat{y}_i + h) + (1 - y_i)(\hat{y}_i - \hat{y}_i^c + h) \tag{4.4}$$

The margin h is now easier to grasp and tune. It encourages the same gap between the probabilities predicted with Z and Z^c across all labels (articles).

We also experiment with a third variant of comprehensiveness, which does not compare the probabilities we obtain with Z and Z^c, comparing instead the two latent document representations:

$$L_g = |\cos(D_M, D_M^c)| \tag{4.5}$$

where cos denotes cosine similarity. This variant forces D_M and D_M^c to be as dissimilar as possible, without requiring a preset margin.

Singularity: A limitation of the comprehensiveness loss (any variant) is that it only requires the mask Z to be better than its complement Z^c. This does not guarantee that Z is better than *every* other mask. Consider a case where the gold rationale identifies three articles and Z selects only two of them. The model may produce better predictions with Z than with Z^c, and D_M may be very different than D_M^c in Equation 4.5, but Z is still not the best mask. To address this limitation, we introduce the *singularity* loss L_r, which requires Z to be better than a mask Z^r, randomly generated per training instance and epoch, that selects as many facts as the sparsity threshold T allows:

$$L_r = \gamma \cdot L_g(Z, Z^r) \tag{4.6}$$
$$\gamma = 1 - \cos(Z^r, Z)$$

Here $L_g(Z, Z^r)$ is any variant of L_g, but now using Z^r instead of Z^c; and γ regulates the effect of $L_g(Z, Z^r)$ by considering the cosine distance between Z^r and Z. The more Z and Z^r overlap, the less we care if Z performs better than Z^r.

The total loss of our model is computed as follows. Again L_p is the classification loss; L_p^c, L_p^r are the classification losses when using Z^c, Z^r, respectively; and all λs are tunable hyper-parameters.

$$\begin{aligned} L = L_p + \lambda_s \cdot L_s + \lambda_c \cdot L_c \\ + \lambda_g \left(L_g + L_p^c\right) + \lambda_r \left(L_r + L_p^r\right) \end{aligned} \tag{4.7}$$

We include L_p^c in Equation 4.7, because otherwise the network would have no incentive to make D_M^c and $\widehat{Y}^c$ competitive in prediction; and similarly for L_p^r.

Rationales supervision: For completeness we also experimented with a variant that utilizes silver rationales for noisy rationale supervision (Zaidan et al., 2007). In this case the total loss becomes:

$$L = L_p + \lambda_{ns} \cdot \text{mae}(Z, Z^s) \tag{4.8}$$

where mae is the mean absolute error between the predicted mask, Z, and the silver mask, Z_s, and λ_{ns} weighs the effect of mae in the total loss.

4.5.4 Experiments

Experimental Setup

For all methods, we conducted grid-search to tune the hyper-parameters λ_*. We used the Adam optimizer (Kingma and Ba, 2015) across all experiments with a fixed learning rate of 2e-5. In preliminary experiments, we tuned the baseline model on development data as a stand-alone classifier and found that the optimal learning rate was 2e-5, searching in the set {2e-5, 3e-5, 4e-5, 5e-5}. The optimal drop-out rate was 0. All methods rely on LEGAL-BERT-SMALL, a variant of BERT (Devlin et al., 2019), with 6 layers, 512 hidden units and 8 attention heads, trained on legal corpora. Based on this model, we were able to use up to 50 paragraphs of 256 words each in a single 32GB GPU.

Preliminary experiments In preliminary experiments, we found that the proposed model (41.9M parameters) (Table 4.7, third row), relying on a shared paragraph encoder (HIER-BERT) to produce both context-aware representations and rationales, has comparable classification performance and better rationale quality compared to: (i) a model with two independent paragraph encoders (82.8M parameters) (Table 4.7, first row), similar to the one used in the literature (Lei et al., 2016; Yu et al., 2019; Jain et al., 2020); (ii) a

model that omits the Q and K projection layers (41.4M parameters) (Table 4.7, second row). Recall that Lei et al. (2016), Yu et al. (2019), and Jain et al. (2020) extract rationales at the word-level, and their encoders, either BILSTMs (Lei et al., 2016; Yu et al., 2019) or BERT (Jain et al., 2020), operate on that level of granularity.

Method	Training parameters	classification micro-F1	Silver rationales mRP	Silver rationales F1
2x HIER-BERT	82.8M	73.4 ± 0.6	35.1 ± 7.9	29.3 ± 8.7
1x HIER-BERT excl. (Q,K)	41.4M	73.5 ± 0.7	29.2 ± 7.9	26.4 ± 7.9
1x HIER-BERT + (Q,K)	41.9M	73.2 ± 0.5	35.9 ± 4.7	39.0 ± 4.9

Table 4.7: Classification performance and rationale quality on development data.

Evaluation Measures

We evaluate: (a) classification performance, (b) *faithfulness* (Section 4.2), and (c) *rationale quality*, while respecting a given sparsity threshold (T).

Classification performance: Given the label skewness, we evaluate classification performance using *micro-F1*, i.e., for each Convention article, we compute its F1, and micro-average over articles.

Faithfulness: Recall that *faithfulness* refers to how accurately an explanation reflects the true reasoning of a model. To measure faithfulness, we report *sufficiency* and *comprehensiveness* (DeYoung et al., 2020). Sufficiency measures the difference between the predicted probabilities for the gold (positive) labels when the model is fed with the whole text ($\widehat{Y_+}^f$) and when the model is fed only with the predicted rationales ($\widehat{Y_+}$). Comprehensiveness (not to be confused with the homonymous loss of Equations 4.3–4.5) measures the difference between the predicted probabilities for the gold (positive) labels obtained when the model is fed with the full text ($\widehat{Y_+}^f$) and when it is fed with the complement of the predicted rationales ($\widehat{Y_+}^c$). We also compare classification performance (again using *micro-F1*) in both cases, i.e., when considering *masked inputs* (using Z) and *complementary inputs* (using Z^c).

Rationale quality: Faithful explanations (of system reasoning) are not always appropriate for users (Jacovi and Goldberg, 2020), thus we also evaluate rationale quality from a user perspective. The latter can be performed in two ways. *Objective* evaluation compares predicted rationales with gold annotations, typically via *Recall, Precision, F1* (comparing system-selected to human-selected facts in our case). In *subjective* evaluation, human annotators review the extracted rationales. We opt for an objective evaluation, mainly due to lack of resources. As rationale sparsity (number of selected paragraphs) differs

across methods, which affects *Recall, Precision, F1*, we evaluate rationale quality with *mean R-Precision* (mRP) (Manning et al., 2009). That is, for each case, the model ranks the paragraphs it selects by decreasing confidence, and we compute Precision@k, where k is the number of paragraphs in the gold rationale; we then average over test cases. For completeness, we also report F1 (comparing predicted and gold rationale paragraphs), although it is less fair, because of the different sparsity of different methods, as noted.

ECHR article	Training instances	Classification F1
2 - Right to life	623	78.3 ± 2.3
3 - Prohibition of torture	1740	85.9 ± 0.9
5 - Right to liberty and security	1623	81.1 ± 1.5
6 - Right to a fair trial	5437	80.1 ± 1.0
8 - Right to respect for private and family life	1056	72.5 ± 1.8
10 - Freedom of expression	441	77.4 ± 1.6
11 - Freedom of assembly and association	162	72.1 ± 3.3
13 - Right to an effective remedy	1665	29.2 ± 3.3
14 - Prohibition of discrimination	444	44.8 ± 7.4
34 - Individual applications	547	10.0 ± 5.0
46 - Binding force and execution of judgments	187	2.6 ± 3.2
P1-1 - Protection of property	1558	77.9 ± 1.3
Rest of the articles	< 100	< 50.0
Overall performance (micro-F1)		72.7 ± 1.2

Table 4.8: Classification performance of HIERBERT-ALL (no mask) across ECHR articles on development data; with respect to the number of training cases (instances), where the corresponding article has been discussed.

Initial Classification Performance

Table 4.8 reports the classification performance of HIERBERT-ALL (no masking, no rationales), across ECHR articles. We observe that F1 is 75% or greater for most of the articles with 1,000 or more training instances. The scores are higher for articles 2, 3, 5, and 6, because (according to the legal expert who provided the gold allegation rationales), (i) there is a sufficient number of cases regarding these articles, and (ii) the interpretation and application of these articles is more fact-dependent than those of other articles, such as articles 10 or 11 (Harris, 2018). On the other hand, although there is a fair amount of train occurrences for articles 13, 34, and 46, these articles are triggered in a variety of ways, many of which turn on legal procedural technicalities.

Tuning the λ Hyper-parameters

Instead of tuning simultaneously all the λ_* hyper-parameters of Equation 4.7, we adopt a greedy, but more intuitive strategy: we tune one λ at a time, fix its value, and proceed to

the next. We begin by tuning λ_s, aiming to achieve a desirable level of sparsity without harming classification performance. We set the sparsity threshold of L_s to $T = 0.3$ (select approx. 30% of the facts), which is the average sparsity of the silver allegation rationales. We found $\lambda_s = 0.1$ achieves the best overall results (classification F1 and sparsity) on development data, thus we use this value for the rest of the experiments. To check our hypothesis that continuity (L_c) is inappropriate in our task, we tried different values of λ_c on development data, confirming that the best overall results are obtained for $\lambda_c = 0$. Thus we omit L_c in the rest of the experiments.

L_g variant	classification micro-F1	sparsity (aim for 30%)	rationale quality F1	rationale quality mRP
Equation 4.3	73.0 ± 0.5	31.4 ± 1.9	35.4 ± 5.8	38.4 ± 5.9
Equation 4.4	73.1 ± 0.7	31.9 ± 1.4	30.3 ± 3.0	32.6 ± 2.6
Equation 4.5	72.8 ± 0.8	31.8 ± 1.3	38.3 ± 2.3	41.2 ± 2.1

Table 4.9: Development results for variants of L_g (*comprehensiveness*) and varying λ_g values (omitted).

Comprehensiveness/Singularity Variants

Next, we tuned and compared the variants of the comprehensiveness loss L_g (Table 4.9). Targeting the label probabilities (Equation 4.4) instead of the losses (Equation 4.3) leads to lower rationale quality. Targeting the document representations (Equation 4.5) has the best rationale quality results, retaining the original classification performance (micro-F1) of Table 4.8. Hence, we keep the L_g variant of Equation 4.5 in the remaining experiments of this section, with the corresponding λ_g value (1e-3).

L_r variant	classification micro-F1	sparsity (aim for 30%)	rationale quality F1	rationale quality mRP
Equations 4.3, 4.6	73.4 ± 0.8	32.8 ± 2.8	36.9 ± 3.6	39.0 ± 3.9
Equations 4.4, 4.6	72.5 ± 0.7	32.0 ± 1.0	39.7 ± 3.1	42.6 ± 3.8
Equations 4.5, 4.6	72.8 ± 0.3	31.5 ± 0.9	33.0 ± 2.7	35.5 ± 2.6

Table 4.10: Development results for variants of L_c (*singularity*) and varying λ_r values.

Concerning the singularity loss L_r (Table 4.10), targeting the label probabilities (Equations 4.4, 4.6) provides the best rationale quality, comparing to all the methods considered. Interestingly Equation 4.5, which performed best in L_g (Table 4.9), does not perform well in L_r, which uses L_g (Equation 4.6). We suspect that in L_r, where we use a random mask Z^r that may overlap with Z, requiring the two document representations D_M, D_M^r to be dissimilar (when using Equation 4.5, 4.6) may be a harsh regularizer with negative effects.

Method	sparsity (aim: 30%)	classification micro-F1	Complementary Input micro-F1	Complementary Input Comp. ↑	Masked Input micro-F1	Masked Input Suff. ↓
RANDOM CLASSIFIER	-	30.8 ± 0.3	-			
HIERBERT-ALL (no masking)	-	**73.4 ± 1.2**	-			
HIERBERT-HA + L_s (Eq. 4.1) Lei et al. (2016)	31.7 ± 1.1	73.1 ± 0.6	58.8 ± 1.5	0.181	69.5 ± 2.4	0.063
HIERBERT-HA + L_s + L_g (Eq. 4.3) Yu et al. (2019)	31.4 ± 1.9	72.8 ± 0.6	59.0 ± 1.5	0.171	68.1 ± 4.4	0.069
HIERBERT-HA + L_s + L_g (Eq. 4.5) (ours)	31.4 ± 1.3	72.6 ± 1.5	59.6 ± 2.7	0.156	69.8 ± 0.8	0.043
HIERBERT-HA + L_s + L_r (Eq. 4.4, 4.6) (ours)	31.5 ± 0.8	72.8 ± 0.5	**55.9 ± 2.8**	**0.204**	**70.5 ± 0.8**	**0.040**
HIERBERT-HA + rationale supervision (Eq. 4.8)	33.1 ± 6.0	73.1 ± 0.5	56.7 ± 6.6	0.191	69.2 ± 1.1	0.053

Table 4.11: *Classification performance* (classification micro-F1) and *faithfulness* results on test data. Faithfulness is measured by considering *Sufficiency* (Suff.) and *Comprehensiveness* (Comp.), i.e., how close the label probabilities of the model are when using the rationales (masked input) or the complements of the rationales (complementary input), respectively, as opposed to using the full input. Lower Suff. and higher Comp. scores are better. We also report micro-F1 for the masked and complementary input; higher and lower F1, respectively, are better.

Task Performance and Faithfulness

Table 4.11 presents results on test data. The models that use the hard attention mechanism and are regularized to extract rationales under certain constraints (HIERBERT-HA + L_*) have comparable classification performance to HIERBERT-ALL. Furthermore, although paragraph embeddings are contextualized and probably have some information leak for all methods, our proposed extensions in rationale constraints better approximate faithfulness, while also respecting sparsity. Our proposed extensions lead to low sufficiency (lower is better, ↓), i.e., there is only a slight deterioration in label probabilities when we use the predicted rationale instead of the whole input. They also lead to high comprehensiveness (higher is better, ↑); we see a 20% deterioration in label probabilities when using the complement of the rationale instead of the whole input. Interestingly, our variant with the singularity loss (Equation 4.4, 4.6) is more faithful than the model that uses supervision on silver rationales (Equation 4.8).

Rationale Quality

We now consider rationale quality, focusing on HIERBERT-HA variants without rationale supervision. Similarly to our findings on development data (Tables 4.9, 4.10), we observe (Table 4.12) that using (a) our version of *comprehensiveness* loss (Equation 4.5) or (b) our *singularity* loss (Equations 4.4, 4.6) achieves better results compared to former methods, and (b) has the best results. The *singularity* loss is better in both settings (silver or gold test rationales), even compared to a model that uses supervision on silver rationales. The random masking of the singularity loss, which guides the model to learn to extract masks

Method	Silver rationales (31%)		Gold rationales (36%)	
	mRP	F1	mRP	F1
RANDOM RATIONALE	30.2 ± 1.1	27.8 ± 1.1	35.1 ± 1.7	30.2 ± 2.2
HIERBERT-HA + L_s (Equation 4.1)	43.1 ± 6.5	37.3 ± 5.4	51.9 ± 5.7	45.7 ± 5.4
HIERBERT-HA + L_s + L_g (Equation 4.3)	41.0 ± 5.1	37.5 ± 6.7	48.9 ± 6.5	44.5 ± 6.8
HIERBERT-HA + L_s + L_g (Equation 4.5)	43.0 ± 1.5	38.5 ± 1.9	50.9 ± 3.2	45.8 ± 3.3
HIERBERT-HA + L_s + L_r (Equations 4.4, 4.6)	45.1 ± 2.1	40.9 ± 2.5	53.6 ± 2.3	48.3 ± 1.2
HIERBERT-HA + supervision (Equation 4.8)	43.1 ± 5.0	39.1 ± 7.1	51.4 ± 6.7	46.8 ± 0.5

Table 4.12: *Rationale quality* results on the 50 test cases that have both silver and gold allegation rationales. Avg. silver/gold rationale sparsity (%) in brackets.

that perform better than *any* other mask, proved to be particularly beneficial in rationale quality. Similar observations are derived given the results on the full test set considering silver rationales. In general, however, we observe that the rationales extracted by all models are far from human rationals, as indicated by the poor results (mRP, F1) on both silver and gold rationales. Hence, there is ample scope for further research.

Qualitative Analysis

Quality of silver rationales: Comparing silver rationales with gold ones, annotated by the legal expert, we find that silver rationales are not complete, i.e., they are usually fewer than the gold ones. They also include additional facts that have not been annotated by the expert. According the expert, these facts do not support allegations, but are included for technical reasons (e.g., *"The national court did not accept the applicant's allegations."*). Nonetheless, ranking methods by their rationale quality measured on silver rationales produces the same ranking as when measuring on gold rationales in the common subset of cases (Table 4.12). Hence, it may be possible to use silver rationales, which are available for the full dataset, to rank systems participating in ECtHR rationale generation challenges.

Model bias: Low mRP with respect to gold rationales means that the models rely partially on non causal reasoning, i.e., they select secondary facts that do not justify allegations according to the legal expert. In other words, the models are sensitive to specific language, e.g., they misuse (are easily fooled by) references to health issues and medical examinations as support for Article 3 alleged violations, or references to appeals in higher courts as support for Article 5, even when there is no concrete evidence. Manually inspecting the predicted rationales, we did not identify bias on demographics. Although such spurious features may be buried in the contextualized paragraph encodings ($P_i^{\texttt{[cls]}}$). In general, *de-biasing* models could benefit rationale extraction and we aim to investigate this direction in future work (Huang et al., 2020).

Plausibility: *Plausibility* refers to how convincing the interpretation is to humans (Jacovi and Goldberg, 2020). While the legal expert annotated all relevant facts with respect to allegations, according to his manual review, allegations can also be justified by subselections (parts) of rationales. Thus, although a method may fail to extract all the available rationales, the provided (incomplete) set of rationales may still be a convincing explanation. To properly estimate plausibility across methods, one has to perform a subjective human evaluation which we did not conduct due to lack of resources.

Manual review of extracted rationales vs. gold rationales: In this section (Figures 4.4-4.8), we present examples of ECtHR cases that have been highlighted (green background color) with gold rationales and marked (green dot on the left) with the ones extracted by our best model.

001-177696 - CASE OF KNEŽEVIĆ v. CROATIA

Alleged Violations: 5 - Right to liberty and security Predicted Alleged Violations: 5 - Right to liberty and security

Model	Facts
	5. The applicant was born in 1961 and lives in Split.
	6. On 19 May 2011 the applicant and several other individuals (see, for further information, Šoš v. Croatia, no. 26211/13, § 17, 1 December 2015) were arrested on suspicion of drug trafficking and detained under Article 123 § 1(2), (3) and (4) of the Code of Criminal Procedure (risk of collusion, risk of reoffending, and seriousness of charges).
	7. During the investigation, an investigating judge of the Split County Court (Županijski sud u Splitu) several times extended the pre-trial detention in respect of the applicant and the other suspects under Article 123 § 1(2), (3) and (4) of the Code of Criminal Procedure (risk of collusion, risk of reoffending, and seriousness of charges). The reasoning of the relevant decisions is outlined in the case of Šoš (cited above, §§ 20 and 23).
●	8. On 18 August 2011 the investigating judge extended the pre-trial detention in respect of the applicant and the other suspects under Article 123 § 1 (3) and (4) of the Code of Criminal Procedure (risk of reoffending and seriousness of charges). He found that all the relevant witnesses had been questioned and that there was no further possibility of remanding the suspects in detention on the grounds of the risk of collusion. As to the other grounds relied upon for the pre-trial detention, the investigating judge reiterated his previous findings.
	9. The investigating judge relied on the same reasons extending the pre-trial detention in respect of the applicant and the other suspects in the further course of the investigation. The reasoning of the relevant decisions is outlined in the Šoš case (cited above, §§ 28, 31, 36 and 41).
	10. On 16 May 2012 the applicant and nine other individuals were indicted on charges of drug trafficking in the Split County Court.
●	11. Following the submission of the indictment, on 18 May 2012 a three-judge panel of the Split County Court extended the pre-trial detention in respect of the applicant and the other accused relying on Article 123 § 1 (3) and (4) of the Code of Criminal Procedure (risk of reoffending and seriousness of charges). His pre-trial detention was extended several times on the same grounds. The reasoning of the relevant decisions is outlined in the Šoš case (cited above, §§ 44, 47 and 52).
●	12. On 20 February 2013 the Supreme Court (Vrhovni sud Republike Hrvatske), acting as a court of appeal, found that the applicant's detention should be extended only under Article 123 § 1 (3) of the Code of Criminal Procedure (risk of reoffending). It explained that the 2013 amendments to the Criminal Code provided that the offence at issue was punishable by a prison sentence of between three and fifteen years and no longer by long-term imprisonment. It was therefore not possible to remand the applicant on the grounds of the seriousness of the charges since the possibility of imposing a sentence of long-term imprisonment was one of the conditions for extending pre-trial detention under Article 123 § 1 (4) of the Code of Criminal Procedure (seriousness of charges).
	13. On 20 April 2013 a three-judge panel of the Split County Court extended the pre-trial detention in respect of the applicant and the other accused under Article 123 § 1 (3) of the Code of Criminal Procedure (risk of reoffending), without changing its previous reasoning.
●	14. On 17 May 2013 a three-judge panel of the Split County Court extended the maximum two-year statutory time-limit for the applicant's pre-trial detention under Article 133 § 1 (4) of the Code of Criminal Procedure for a further six months (until 19 November 2013) relying on section 35(2) of the Office for the Suppression of Corruption and Organised Crime Act (hereinafter "the OSCOCA").
	15. The applicant appealed to the Supreme Court arguing that section 35(2) of the OSCOCA was inapplicable to his case since he was not detained during the investigation.
●	16. On 7 June 2013 the Supreme Court dismissed the applicant's appeal on the grounds that the said provision of the OSCOCA made a mistaken reference to Article 130 § 2 of the Code of Criminal Procedure. It also considered that the cited provision was incomprehensible since, if understood as provided in that Act, it merely repeated paragraph 1 of section 35 of the OSCOCA, which would be obsolete. Instead it should be interpreted in line with the previous abrogated version of the OSCOCA, which in its section 28(3) had provided for a possibility of extension of the overall maximum period of detention for a further six months, which was in the applicant's case until 19 November 2013.
	17. On 18 June 2013 the applicant lodged a constitutional complaint with the Constitutional Court (Ustavni sud Republike Hrvatske) reiterating his previous arguments.
	18. On 11 July 2013 the Constitutional Court dismissed the applicant's constitutional complaint as unfounded, endorsing the reasoning of the Supreme Court.
	19. The applicant's pre-trial detention was extended, under Article 123 § 1 (3) of the Code of Criminal Procedure (risk of reoffending), until the maximum period expired on 19 November 2013, when he was released.

Figure 4.4: Automatically extracted (dots) and gold (highlighted text) allegation prediction rationales in ECtHR case no. 001-177696.

In Figure 4.4 The model extracted most of the relevant facts indicating a possible violation of Article 5. Note that 67% (10 of 15) the facts were considered relevant by the legal expert. Our model has a disadvantage in this case because, being trained to operate at a predefined sparsity level (30%), it extracted only 5 of the 15 facts (33%).

001-178361 - CASE OF K.I. v. RUSSIA

Alleged Violations: 3 - Prohibition of torture, 5 - Right to liberty and security Predicted Alleged Violations: 3 - Prohibition of torture

Model	Facts
	7. The applicant was born in 1980. He arrived in Russia in 2003. He travelled to Tajikistan on a number of occasions to visit his parents for short periods of time.
	8. On 3 May 2011 the applicant was charged in absentia in Tajikistan with participating in an extremist religious movement, the Islamic Movement of Uzbekistan, and an international search and arrest warrant was issued in his name. On 6 May 2011 the Tajik authorities ordered his pre-trial detention.
	9. On 3 November 2013 the applicant was arrested in Moscow and detained. On 4 November 2013 the Meshchanskiy District Court of Moscow ("the District Court") ordered his detention pending extradition.
	10. On 4 December 2013 the Tajik prosecution authorities requested the applicant's extradition on the basis of the above charges. The request included assurances regarding his proper treatment, which were formulated in standard terms.
	11. On 12 December 2013 the District Court extended the applicant's detention until 3 May 2014.
	12. An appeal by the applicant of 16 December 2013 was dismissed by the Moscow City Court ("the City Court") on 3 February 2014.
	13. On 29 April 2014 the District Court again extended the applicant's detention until 3 August 2014.
	14. An appeal by the applicant of 5 May 2014 was dismissed by the City Court on 23 July 2014.
	15. On 9 October 2014 the applicant's extradition was refused by the Deputy Prosecutor General of the Russian Federation, owing to the absence of culpable actions under Russian criminal law.
	16. On 13 October 2014 the applicant was released from detention.
	17. On 13 October 2014, immediately after his release, the applicant was rearrested for violating migration regulations.
●	18. On 14 October 2014 the District Court found the applicant guilty of violating migration regulations, fined him and ordered his administrative removal. Allegations by the applicant regarding a real risk of ill-treatment were dismissed, and he was detained pending expulsion. The District Court assessing the risks stated that "[t]he claims of the representative ... are of a speculative nature and not confirmed by the case materials"
●	19. The above judgment was upheld on appeal by the City Court on 24 October 2014. Claims by the applicant under Article 3 of the Convention were dismissed with reference to the District Court's assessment of the case, which took into consideration "...the nature of the administrative offence, the character of the accused [who was criminally convicted in Russia]... the length of his stay in Russia and other circumstances of the case".
	20. According to the latest submissions of his representative in 2015, the applicant was still in detention.
	21. On 18 December 2013 the applicant lodged a request for refugee status, referring to persecution in Tajikistan and a real risk of ill-treatment.
●	22. On 15 September 2014 his request was refused by a final administrative decision of the migration authorities. The applicant challenged that decision in the courts, referring, inter alia, to the risk of ill-treatment.
	23. On 12 November 2015 his appeals were dismissed by a final decision of the City Court.

Figure 4.5: Automatically extracted (dots) and gold (highlighted text) allegation prediction rationales in ECtHR case no. 001-178361.

In Figure 4.5, paragraphs 9, 11, 13 and 20 clearly indicate possible violation of the right to liberty (Article 5), as they refer to continuous extension of applicant detention but our model was unable to extract them, thus it is unable to predict this allegation. The model targeted only paragraphs that indicate ill-treatment, which is connected to plausible violation of Article 3 (Prohibition of Torture).

In Figure 4.6, paragraphs 16 and 19 clearly indicate that the applicant's health (life) was at risk and authorities did not pay attention, although these paragraphs were not selected by the model. Instead, paragraph 10 states that the applicant initially informed the authorities for his medical history and they provided medication. This is an indication for model sensitivity in language describing health issues in general (tuberculosis) and not specific well-defined accusations.

In Figure 4.7, a causal inference would connect paragraph 8 (initial trial) with paragraphs 20-22 (next trials) to infer mistrial because there was no verdict after a reasonable period of time. Instead, the model seems to be sensitive to references for the implication of higher courts as justification of mistrial (paragraphs 10, 13, 18, and 21). This suggests that the model probably follows poor (greedy) reasoning.

001-178748 - CASE OF KAIMOVA AND OTHERS v. RUSSIA

Alleged Violations: 2 - Right to life Predicted Alleged Violations: 2 - Right to life

Model	Facts
	6. Ms Damani Kaimova, Ms Maryam Moldyyevna Kaimova, and Ms Zarina Tamiyevna Maskhurova were born on 16 February 1953, 13 January 2005, and 18 September 1981, respectively. They live in the Chechen Republic. The first applicant is the mother, the second applicant is a daughter, and the third applicant is the widow of the late Mr Kaimov.
	7. On 23 September 2006 Mr Kaimov was arrested for being a member of an illegal military organisation in the Chechen Republic. He remained in detention throughout the investigation and trial. On 1 November 2006 the Achkhoiy-Martan District Court of the Chechen Republic found him guilty of charges related to the military organisation and illegal acquisition of weapons. He was sentenced to two and a half years' imprisonment.
	8. In the meantime, he was charged with attempted murder of law-enforcement officials, with making a homemade explosive, and other offences. He was convicted as charged by the Supreme Court of the Chechen Republic on 28 October 2008 and sentenced to six and a half years' imprisonment.
	9. Prior to his detention Mr Kaimov had been diagnosed with tuberculosis for which he had been receiving outpatient treatment in a local hospital.
●	10. On admission to a remand prison Mr Kaimov informed the custodial authorities of his history of tuberculosis. A chest X-ray in January 2007 examination revealed the signs of that disease. A standard treatment with first-line medication was prescribed.
	11. In early 2009 Mr Kaimov was sent to serve his sentence to the Republic of Tatarstan. In March 2009 he was admitted to prison medical institution no. 1 in Nizhnekamsk, where his tuberculosis was cured as confirmed by a medical board on 7 June 2009.
	12. On 2 October 2009 Mr Kaimov was discharged from the prison medical institution to remand prison no. IZ-16/2 in Kazan. Shortly thereafter his health worsened.
	15. On 14 April 2010, in response to Mr Kaimov's "negligent attitude towards his treatment", a doctor talked to him about the importance of taking his drugs regularly. On 22 April and 1 May 2010 the doctor had repeated talks with him on the issue.
	16. In late May 2010, the first applicant visited her son. Mr Kaimov was in a poor health. He claimed that no treatment had been given to him and that "the medical staff [had] paid absolutely no attention to his condition".
	17. On 22 April, 31 May and 8 June 2010 the doctor responsible for Mr Kaimov's treatment again noted in the medical file that the patient was not taking his drugs as prescribed and insisted that he should follow medical instructions properly. The medical records were not signed by Mr Kaimov.
	18. By mid-June 2010 Mr Kaimov's condition became serious. He was no longer able to leave his bed.
	19. On 24 June 2010 an inmate of the remand prison allegedly informed the first applicant that her son's condition had become very serious and that no medical care was being given to him.
●	20. Four days later Mr K., a lawyer working with the Russian Justice Initiative, interviewed Mr Kaimov in the prison hospital. He said that he had not received the medicines, as the prison hospital did not have them. The prison hospital's management refused to accept parcels with drugs for detainees.
	21. Mr Kaimov died of heart failure caused by tuberculosis on 1 July 2010. The first applicant did not allow an autopsy to take place.
●	22. According to the Government, the investigating authorities carried out a criminal inquiry into the circumstances of Mr Kaimov's death, which ended with a decision of 21 July 2010 not to open a criminal case.
●	23. On 22 November 2010 Mr K. asked the head of the Investigative Committee of the Republic of Tatarstan to investigate the circumstances of Mr Kaimov's death. He pointed out that the detainee had complained of the lack of treatment in detention. A copy of the interview record of 28 June 2010 was attached to the request.
	24. The investigating authorities interviewed Mr K., who confirmed his statements, and Ms I., the head of the tuberculosis unit responsible for Mr Kaimov's treatment in 2009 and 2010. The doctor stated that the patient had received tuberculosis treatment until late May 2010, when he had refused to take any drugs.
	25. On 6 December 2010, citing statements by Ms I., the investigating authorities concluded that there had been no appearance of negligence on the part of the medical authorities. They decided not to open a criminal case.

Figure 4.6: Automatically extracted (dots) and gold (highlighted text) allegation prediction rationales in ECtHR case no. 001-178748.

001-180500 - CASE OF BRAJOVIĆ AND OTHERS v. MONTENEGRO

Alleged Violations: 6 - Right to a fair trial Predicted Alleged Violations: 6 - Right to a fair trial

Model	Facts
	5. The applicants were born in 1931, 1972, 1948, 1965, 1970, and 1964 respectively, and live in Golubovci.
	6. The facts of the case, as submitted by the parties, may be summarised as follows.
	7. The applicants intervened, as injured party, in criminal proceedings against X, in the course of which they sought 2.705,70 euros (EUR) as compensation for legal costs.
	8. On 14 October 2008 the High Court (Viši sud) in Podgorica found X guilty and, inter alia, ordered him to pay the applicants 505.70 euros (EUR) for the costs of legal representation, without specifying what exactly was covered by this amount.
	9. On an unspecified date X and the High State Prosecutor appealed.
●	10. On 30 March 2009 the applicants appealed in respect of costs and expenses. On 6 May 2009 the High Court transmitted the applicants' appeal to the Court of Appeal (Apelacioni sud) in Podgorica.
	11. On 22 September 2009 the Court of Appeal ruled on the appeals lodged by the High State Prosecutor and X. The applicants learned of this judgment on 27 May 2010 when checking the case-file at the High Court. It was served on them on 3 October 2013.
	12. On 28 May 2010 the applicants complained to the President of the Supreme Court that the Court of Appeal had failed to rule on their appeal.
●	13. On 7 June 2010 the President of the Supreme Court notified them that she had been informed by the High Court President that the case file had been "at the Court of Appeal in order for it to rule on [their] appeal in respect of costs of criminal proceedings given that it had not been ruled upon by [its] judgment of 22 September 2009".
	14. On 24 October 2011 the applicants requested the President of the High Court to transmit the case file to the Court of Appeal given that they had learnt that the file had been archived in the High Court, contrary to what that court had said to the President of the Supreme Court.
	15. On 11 January 2012 the applicants again complained to the President of the Supreme Court.
	16. It would appear that the Court of Appeal has not ruled on the applicants' appeal.
	17. On 14 March 2011, in the absence of any ruling by the Court of Appeal, the applicants filed a compensation claim against the State.
●	18. On 17 June 2011 the Court of First Instance (Osnovni sud) in Podgorica rejected the claim (odbacuje se) finding that the High Court had awarded them the costs, which judgment had become final in the meantime, and that the issue was thus res iudicata.
	19. On 7 July 2011 the High Court upheld this judgment.
●	20. On 12 July 2012 the Constitutional Court (Ustavni sud) dismissed the applicants' constitutional appeal, considering that there was no violation of Article 6 as res iudicata was indeed a procedural obstacle which prevented further proceedings. It further held that the applicants' dissatisfaction with the costs awarded in the criminal proceedings did not mean that they could claim them by a regular civil claim (putem redovne građanske tužbe). In any event, the civil proceedings could not serve to correct the final decisions issued in criminal proceedings.
●	21. On 8 December 2011 the High Court issued a decision ordering its finance department (računovodstvo) to pay the applicants' representative the amount awarded by the High Court on 14 October 2008. This decision became final on 5 January 2012, given that no appeal was lodged against it.
●	22. On 10 January 2017 the High Court informed the Agent's office that the Court of Appeal had not ruled on the applicants' appeal in respect of costs of criminal proceedings, but that the High Court, after its judgment of 14 October 2008 had become final, had issued a decision on 8 December 2011 ordering that the applicants' representative be paid the sum awarded thereby.

Figure 4.7: Automatically extracted (dots) and gold (highlighted text) allegation prediction rationales in ECtHR case no. 001-180500.

001-181279 - CASE OF RAJAK v. MONTENEGRO

Alleged Violations: 6 - Right to a fair trial Predicted Alleged Violations: 6 - Right to a fair trial, P1-1 - Protection of property

Model	Facts
	4. The applicant was born in 1961 and lives in Bijela, Montenegro.
●	5. On 1 March 2012 the Herceg Novi First Instance Court rendered a judgment in favour of the applicant and ordered the applicant's employer "Vektra Boka" AD Herceg Novi (hereinafter "the debtor") to carry out a re-allocation of plots for the construction of apartments. This judgment became final on 21 December 2012.
●	6. On 15 January 2013 the applicant requested enforcement of the above judgment and the Herceg Novi First Instance Court issued an enforcement order on 31 January 2013.
	7. On 12 June 2015 the Commercial Court opened insolvency proceedings in respect of the debtor.
	8. On 28 January 2016 the Herceg Novi First Instance Court transferred the case to the Commercial Court for further action.
●	9. On 22 March 2016 the Commercial Court suspended (obustavio) the enforcement due to the opening of the insolvency proceedings, which decision became final on 11 May 2016.
	10. The judgement in question remains unenforced to the present day.
●	11. On 8 February 2013 the applicant instituted administrative proceedings seeking, on the basis of the above judgment, the removal of competing titles from the Land Register.
	12. On 29 July 2015 the Real Estate Directorate terminated (prekinuo) the administrative proceedings because the Commercial Court had commenced insolvency proceeding in respect of the debtor.
	13. On 7 September 2015 the applicant submitted an objection against the above decision. This objection was rejected as being out of time by the Real Estate Directorate on 5 October 2015.
	14. The administrative proceedings are still pending.
●	15. On an unspecified day in 2003, the applicant instituted separate civil proceedings against the debtor, as his former employer, seeking reinstatement and damages. Following three remittals, on 3 March 2014 the Herceg Novi First Instance Court rendered a judgment in the applicant's favour.
	16. On 22 September 2015 the High Court upheld this judgment on the merits, but quashed it as regards the costs.
	17. On 31 October 2016 the Herceg Novi First Instance Court transferred the case to the Commercial Court for further action due to the commencement of the insolvency proceedings in respect of the debtor.
	18. On 22 February 2017 the Commercial Court ruled partly in favour of the applicant regarding the costs.
	19. The parties did not inform the Court about when the Commercial Court's decision became final and was served on the applicant.

Figure 4.8: Automatically extracted (dots) and gold (highlighted text) allegation prediction rationales in ECtHR case no. 001-181279.

In Figure 4.8, similarly to the case presented in Figure 4.7, the main argument, in this case, is mistrial because there was no verdict after a reasonable period of time (paragraphs 5 and 18-19). The model selected paragraph 11, which does not indicate possible violations. Given the model's prediction for allegations with respect to Article 1 of the 1st Protocol on the protection of property, we believe that this paragraph was conceived as justification on that matter.

4.6 Conclusions

In this Chapter, we discussed the challenging application of legal judgment prediction. Contrary to our findings in LMTC with EURLEX57K dataset (Section 3.6), in ECtHR cases, we have to consider the full text of the documents (court cases). As court cases are extremely long and modern pre-trained transformer-based are limited to processing up to 512 tokens at most, we proposed a new state-of-art method HIER-BERT to tackle this issue, while effectively modeling the natural document structure, i.e., list of paragraphs. For the first time, we considered three downstream tasks (binary and multi-label classification, case importance prediction), highlighting the escalated difficulties in the latter two cases that have not been considered in the literature. Further on, we studied the extent of model bias in demographic information, where both quantitative (classification performance drop) and qualitative (attention to demographics) evidence highlighted that model bias exists, but only slightly affects the predictions of the models on average. Considering the limitations of studying soft attention scores as a means for explainability, we followed recent work in rationale extraction and introduced a new task, paragraph-level rationale extraction, where a model has to extract the most relevant paragraphs of a court case with respect to the alleged violations in this case. For this purpose, we rely on a similar model to HIER-BERT, using hard attention masking with additional regularizers (HIERBERT-HA). We study several alternatives following the literature (Lei et al., 2016; Yu et al., 2019), and propose a new regularizer, which leads to state-of-the-art performance in both faithfulness and rationale quality measures proposed by DeYoung et al. (2020).

In the future, we would like to consider a pipe-lined framework, where alleged violation prediction could be a first module, providing primary information for the more difficult task of judgment prediction in the realistic and challenging multi-label setup. In the same direction, additional modules (systems) trained to predict the relevant case law to a given case, could also provide valuable information for the aforementioned task, which is also part of the court reasoning considering the facts, the law and the case law on par. In the case of explainability, which is a core feature to provide formal explanations of the overall system's decisions, more structured explanations closer to human reasoning shall be constructed. In this direction, modeling the inter-paragraph relations is a crucial characteristic to provide factual patterns, i.e., paragraph X1 in light of paragraph X2 and X3 leads to conclusion Y, instead of factual masks that are currently extracted by the methods described in Section 4.5.3. Last but not least, a more extensive in-depth analysis on model bias identification and model de-biasing would be an important step towards improving fairness, which may also lead to improvements in faithfulness and rationale quality, as the model will then be unable to rely in this kind of spurious information.

Chapter 5

Legal Document to Document Information Retrieval

5.1 Introduction

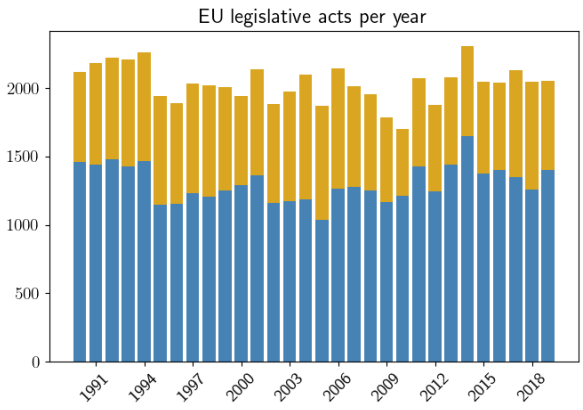


Figure 5.1: Number of legislative acts issued by the EU per year. The gold color of the bars indicates how many of the published acts are amendments to older ones

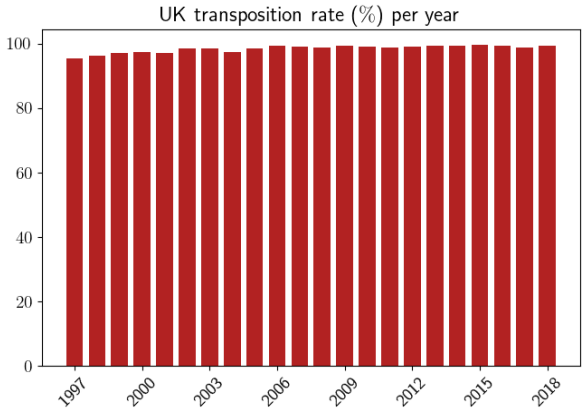


Figure 5.2: The percentage of EU directives transposed by UK legislation per year. Over 98% of the published EU directives have been transposed.

Major scandals in corporate history, from Enron to Tyco International, Olympus, and Tesco,[1] have led to the emergence of stricter regulatory mandates and highlighted the need for *regulatory compliance* where organizations need to ensure that they comply with relevant regulations (Lin, 2016). However, keeping track of the constantly changing legislation (Figure 5.1) is hard, thus organizations are increasingly adopting Regulatory Technology (RegTech) to facilitate the process.

[1] `www.theguardian.com/business/2015/jul/21/the-worlds-biggest-accounting-scandals-toshiba-enron-olympus`

Dataset	Domain	$\tilde{q}$	$\tilde{d}$
IR datasets in the literature			
TREC ROBUST (Voorhees, 2005)	News	3 / 14	254
BIOASQ (Tsatsaronis et al., 2015)	Biomedical	9	197
IR datasets with verbose queries			
GOV2 (Clarke et al., 2004)	Web	11 / 57	682
WT10G (Chiang et al., 2005)	Web	11 / 35	457
Regulatory Compliance datasets			
EU2UK (ours)	Law	2,642	1,849
UK2EU (ours)	Law	1,849	2,642

Table 5.1: Statistics for query and document length (counted in words) for IR datasets used in literature.

Typically, a compliance regimen includes three distinct but related types of measures, *corrective*, *detective*, and *preventive* Sadiq and Governatori (2015). Corrective measures are usually undertaken when new regulations are introduced to update existing policies. Detective measures, ensure "after-the-fact" compliance, i.e., following a procedure, a manual or automated check is carried out, to ensure that every step of the procedure complied with the corresponding regulations. Finally, preventive measures ensure compliance "by design", i.e., during the creation of new policies. All types of measures include an underlying information retrieval (IR) task, where relevant regulations need to be retrieved given a policy or vice versa. We identify two use cases:

1. *Given a new regulation retrieve all the policies of the organization affected by this law.* The organization can then apply corrective measures to ensure compliance for these policies.
2. *Given a policy retrieve all relevant laws the control should comply with.* This is useful for ensuring compliance after a procedure has been carried out (detective measures) or when creating new policies (preventive measures).

Regulatory information retrieval (REG-IR), similarly to other applications of *document-to-document* (DOC2DOC) IR, for example, case law rerieval, is much more challenging than traditional IR where the query typically contains a few informative words and the documents are relatively short (Table 5.1), often documents abstracts only. In DOC2DOC IR the query is a long document (e.g., a regulation) containing thousands of words, most of which are uninformative. Consequently, matching the query with other long documents where the informative words are also sparse, becomes extremely difficult.

Although legislation is publicly available, organizations' policies are private and very

hard to obtain. Fortunately, the European Union (EU) has a legislation scheme analogous to regulatory compliance for organizations. According to the Treaty on the Functioning of the European Union (TFEU),[2] all published EU *directives* must take effect at the national level. Thus, all EU member states must adopt a law to transpose a newly issued directive within the period set by the directive (typically 2 years). Notably, the United Kingdom (UK) having a high compliance level with the EU (Figure 5.2),[3] is a good test-bed for REG-IR. Thus we compile and release two datasets for REG-IR, EU2UK and UK2EU, containing EU directives and UK regulations, which can serve both as queries and documents under the ground truth assumption that a UK law is relevant to the EU directives it transposes and vice versa.[4]

5.2 Related Work

IR in the legal domain is widely connected with the Competition on Legal Information Extraction/Entailment (COLIEE). From 2015 to 2017 (Kim et al., 2015a, 2016a; Kano et al., 2017) the task was to retrieve Japanese Civil Code articles given a question, while in COLIEE 2018 and 2019 (Kano et al., 2018; Rabelo et al., 2019) the task was to retrieve supporting cases given a short description of an unseen case. However, the texts of these competitions are short compared to our datasets. Also, most submitted systems do not consider recent advances in IR, i.e, neural ranking models (Guo et al., 2016; Hui et al., 2017; McDonald et al., 2018; MacAvaney et al., 2019) which have recently managed to improve rankings of conventional IR, or end-to-end neural models which have recently been proposed (Fan et al., 2018; Khattab and Zaharia, 2020).

In the first phase of COLIEE 2015, the proposed systems needed to extract a subset of Japanese Civil Code articles relevant to a given question. The proposed system of Kim et al. (2015b) won this competition by retrieving the most similar documents (articles) to the question according to TF-IDF scores and then re-ranking those documents using the ranking SVMs model. Three types of features were used for this method: binary representations of lemmas and dependency pairs, TF-IDF scores, and a Linear Discriminant Analysis (LDA) (Gado et al., 2016) score. Kim et al. (2016b) also won the first phase of COLIEE 2016 proposing an ensemble to compute similarity using a Least Squares Regression method (LSR) and LDA based on a variety of features, including lexical similarity, syntactic similarity, and semantic similarity.

In phase one of COLIEE 2017, the methodology that Morimoto et al. (2017), the

[2]Articles 291 (1) and 288 paragraph 3.

[3]Data for Figures 5.1 and 5.2 obtained from `ec.europa.eu/internal_market/scoreboard/performance_by_governance_tool/eu_pilot`.

[4]The work of this chapter was carried out before the UK left the EU.

winners of this competition, followed to identify the similarity of a query to a civil law article, comprised the extraction of the requirement (condition), the effect (conclusion) in law articles and the examined query Q. The authors extracted the legal requirement and effect parts from queries and articles using rule-based (pattern-matching) methods. The distance between a query Q and an article T was calculated as the sum of the distances of the requirement parts of Q and T and the distance between the effect parts of Q and T. The articles to be retrieved should not exceed a pre-defined distance threshold. Word Mover's Distance (WMD) (Kusner et al., 2015) was used for all distances.

In COLIEE 2018, Vu Tran and Nguyen (2018) proposed the best system for the case law retrieval task, meaning extracting supporting cases given a new case. They explored benefits from analyzing the summaries of legal documents and logical structures. They extended the summary of both the query and the candidates to include more attributes from factual paragraphs (Chapter 4). They obtained document and query embeddings from the corresponding summaries computing word embedding centroids. This information is used to estimate the score of each document given the summaries and factual paragraphs of the document and the query.

5.3 Contributions

- We introduce regulatory information retrieval (REG-IR), an application of document-to-document IR, which belongs in a new family of IR tasks, where both queries and documents are long, typically containing thousands of words.

- We compile and release the two first publicly available datasets, EU2UK and UK2EU, suitable for REG-IR and DOC2DOC IR in general.

- We show that fine-tuning BERT on an in-domain classification task produces the best document representations with respect to IR and improves pre-fetching results compared to various methods including the traditional BM_{25} and generic BERT models (BERT-BASE of Devlin et al. (2019), S-BERT of Reimers and Gurevych (2019)).

5.4 Datasets

5.4.1 Data sources

EU/UK Legislation: We have downloaded approximately 56K pieces of EU legislation (approx. 3.9K directives), from the EUR-LEX portal (`eur-lex.europa.eu`). EU laws are 2,642 words long on average and are structured in three major parts: the *title* (Table 5.3,

Dataset	Documents in pool	Train		Development		Test	
		Queries	Avg. relevant	Queries	Avg. relevant	Queries	Avg. relevant
EU2UK	52,515	1,400	1.79	300	2.09	300	1.74
UK2EU	3,930	1,500	1.90	300	1.46	300	1.29

Table 5.2: Detailed statistics for EU2UK and UK2EU. Both datasets have a relatively small number of relevant documents per query. EU2UK has a much larger document pool, which which may impose extra difficulties in the retrieval.

query), the *recitals* consisting of references in the legal background of the act, and the *main body*. We have also downloaded approx. 52K UK laws, publicly available from the official UK legislation portal.[5] UK laws are 1,849 words long on average and contain the *title* (Table 5.3, document title) and the *main body*.

Transpositions: We have retrieved all transposition relations (approx. 3.7K) between EU directives and UK laws from the CELLAR database (Chapter 3). CELLAR only provides the mapping between the CELLAR ids of EU directives and the title of each UK law. Therefore we aligned the CELLAR ids with the official UK ids based on the law title. One or more UK laws may transpose one or more EU directives.

5.4.2 Datasets compilation

Let $\mathcal{E}$, $\mathcal{U}$ be the sets of EU directives and UK laws, respectively. We define REG-IR as the task where the query q is a document, e.g, an EU directive, and the objective is to retrieve a set of relevant documents, $\mathcal{R}_q$, from the pool of all available documents, e.g., all UK laws. We create two datasets:

EU2UK: $q \in \mathcal{E}$, $\mathcal{R}_q = \{r_i : r_i \in \mathcal{U}, r_i \xrightarrow{\text{transposes}} q\}$.

UK2EU: $q \in \mathcal{U}$, $\mathcal{R}_q = \{r_i : r_i \in \mathcal{E}, q \xrightarrow{\text{transposes}} r_i\}$.

Table 5.2 shows the statistics for the two datasets, which are split in three parts, *train*, *development*, and *test*, retaining a chronological order for the queries (i.e., all development queries were published later than all training queries). EU2UK has a much larger pool of available documents than UK2EU (52.5K vs. 3.9K), which may impose an extra difficulty during retrieval. More importantly, the average number of relevant documents per query is small (at most 2) for both datasets, as our ground truth assumption is strict, i.e., relevant documents are those linked to the query with a transposition relation. In addition, EU legislation is frequently amended (Figure 5.1) which also increases the difficulty in the retrieval task. Let $d_1 \in \mathcal{E}$ be a directive transposed by $u_1 \in \mathcal{U}$ and $d_2 \in \mathcal{E}$ be

[5] `legislation.gov.uk`

Query: DIRECTIVE 2006/66/EC OF THE EUROPEAN PARLIAMENT AND OF THE COUNCIL of 6 September 2006 on batteries and accumulators and waste batteries and accumulators and repealing Directive 91/157/EEC

BM_{25} rank	Relevant	Document title
1	No	The Batteries and Accumulators (Placing on the Market) (Amendment) Regulations 2012
2	No	The Batteries and Accumulators (Containing Dangerous Substances) (Amendment) Regulations 2000
3	No	The Batteries and Accumulators (Placing on the Market) (Amendment) Regulations 2015
4	No	The Batteries and Accumulators (Containing Dangerous Substances) Regulations 1994
5	No	The Waste Batteries and Accumulators (Amendment) Regulations 2015
6	Yes	The Waste Batteries and Accumulators Regulations 2009
12	Yes	The Batteries and Accumulators (Placing on the Market) Regulations 2008

Table 5.3: Example from the EU2UK dataset where the retrieved UK laws are ranked by BM_{25}. The top-5 documents seem similar to the query but are not relevant (not linked by transposition). The documents ranked 1st, 3rd, and 5th are amendments of the relevant documents, i.e., they are amendments of the UK laws that actually transpose (are relevant to) the query.

a directive amending d_1. The UK must adopt a law, u_2, to transpose d_2. Both d_2 and u_2 cover similar concepts to those of d_1 (d_2 is an amendment and u_2 must comply with d_2), but, strictly speaking u_2 is relevant only to d_2. In the rest of this chapter, as we already defined above, we consider two documents $d \in \mathcal{E}$, $u \in \mathcal{U}$ to be relevant if and only if u transposes d. Table 5.3 shows an example from EU2UK, where the top-5 documents seem very similar to the query but are not considered relevant, in the sense the term has in this chapter. Note that the documents ranked 1st, 3rd and 5th, are amendments of the relevant documents.

5.5 Methods

Since REG-IR is a new task, our starting point is the two-step pipeline approach followed by most modern neural information retrieval systems (Guo et al., 2016; Hui et al., 2017; McDonald et al., 2018). First, a conventional IR system (*pre-fetcher*) retrieves the k most prominent documents. Then a neural model attempts to rank relevant documents higher than irrelevant ones. While this configuration is widely adopted in literature, the re-ranking step could be omitted, provided an effective pre-fetching mechanism is available, i.e., the pre-fetcher would then act as an end-to-end IR system. In Sections 5.5.1-5.5.2, we describe the examined methods for both pre-fetching and re-ranking steps.

5.5.1 Document pre-fetching

Okapi BM$_{25}$ (Robertson et al., 1995) is a bag-of-words scoring function estimating the relevance of a document d to a query q, based on the query terms appearing in d, regardless their proximity within d:

$$\sum_{i=1}^{n} \text{idf}(q_i) \cdot \frac{\text{tf}(q_i, d) \cdot (k_1 + 1)}{\text{tf}(q_i, d) + k_1 \cdot \left(1 - b + b \cdot \frac{L}{\bar{L}}\right)} \tag{5.1}$$

where q_i is the i-th query term, with $\text{idf}(q_i)$ being the inverse document frequency and $\text{tf}(q_i, d)$ the term frequency. L is the length of d in words, $\bar{L}$ is the average length of the documents in the collection, k_1 is a parameter that favors high tf scores and b is a parameter penalizing long documents.[6]

W2V-CENT: Following Brokos et al. (2016), we represent query/document terms with pre-trained embeddings. For each query/document we calculate the tf-idf weighted centroid of its embeddings:

$$\text{cent}(t) = \frac{\sum_{i=1}^{l} \boldsymbol{x}_i \cdot \text{tf}(x_i, t) \cdot \text{idf}(x_i)}{\sum_{i=1}^{l} \text{tf}(x_i, t) \cdot \text{idf}(x_i)} \tag{5.2}$$

where t is a text (query or document) and x_i is the i-th text term with embedding $\boldsymbol{x}_i$. The documents are ranked, with respect to the query, by a k nearest neighbours (kNN) algorithm with cosine distance:

$$\cos_{\text{d}}(q, d) = 1 - \frac{\text{cent}(q) \cdot \text{cent}(d)}{\|\text{cent}(q)\| \cdot \|\text{cent}(d)\|} \tag{5.3}$$

BERT (Devlin et al., 2019) similarly to W2V-CENT, relies on pre-trained token representations, which now are extracted from BERT, thus being context-aware.[7] A text can be represented by its `[cls]` token embedding or by the centroid of its token embeddings. In the latter case, the embeddings can be extracted from any of the 12 layers of BERT.[8] Note that the texts in our datasets do not entirely fit in BERT. We thus split them into c chunks (1 to 3 per text) and pass each chunk through BERT to obtain a list of token embeddings per layer (i.e, the concatenation of c token embeddings lists) or c `[cls]` tokens. The final representation is either the centroid of the token embeddings or the centroid of the `[cls]` tokens.

SENTENCE-BERT (**S-BERT**) (Reimers and Gurevych, 2019) is a BERT model fine-tuned for Natural Language Inference (NLI) in STS-B dataset (Cer et al., 2017). According

[6] We use *elastic*, a widely used IR engine with the BM$_{25}$ scoring function. See `www.elastic.co/`.

[7] All models with a BERT encoder use the -BASE version, i.e., 12 layers, 768 hidden units, 12 attention heads.

[8] BERT is not fine-tuned during this process.

to the authors, training S-BERT results in better representations than BERT for tasks involving text comparison, like IR. We use the same setting as in BERT.

EU-BERT: We will also use a BERT model further pre-trained on EU legislation, dubbed here EU-BERT. The goal is to estimate the impact of in-domain language, similar to our work in the rest of the legal NLP tasks; comparing its performance with the two previous generic models (BERT, S-BERT). We use the same setting as in BERT.

CONCEPT-BERT (C-BERT): EU laws are also annotated with EUROVOC concepts (Chapter 3) covering the main subjects of the laws (e.g., trade, energy, etc.). Our intuition is that a UK law transposing an EU directive will most probably cover the same core subjects, as the EU directive. Thus we expect that a BERT model, fine-tuned to predict EUROVOC concepts, will learn rich representations describing these concepts which may be useful for pre-fetching. We fine-tune BERT similarly to our work in Chapter 3. We use all EU laws excluding EU directives and use the resulting model to extract representations for documents and queries similarly to the previous BERT-based methods.

ENSEMBLE: This method is simply a linear combination of our best two pre-fetchers, C-BERT and BM_{25}:

$$\text{ENS}(q, d) = \alpha \cdot \text{CB}(q, d) + (1 - \alpha) \cdot \text{BM}_{25}(q, d) \tag{5.4}$$

where CB is the score of C-BERT, α is an importance weight, which is tuned on development data. The scores of the pre-fetchers are normalized in $[0, 1]$ using min-max normalization.

5.5.2 Document re-ranking

Modern neural re-rankers operate on pairs of the form (q, d) to produce a relevance score, $\text{rel}(q, d)$, for a document d with respect to a query q. Note, however that the main objective is to rank relevant documents higher than irrelevant. Thus, during training a max marginal loss is usually calculated as:

$$\mathcal{L} = \max(0, 1 - \text{rel}(q, d^+) + \text{rel}(q, d^-)) \tag{5.5}$$

where d^+ is a relevant document and d^- is an irrelevant document. The loss of Equation 5.5 requires the relevance score between the query and a relevant document to be higher than the score between the query and an irrelevant document by a margin of 1. We have experimented with several neural re-ranking methods, each having a function that produces a relevance score s_r for each of the top-k documents returned by the best pre-fetcher. The final relevance score of a document is calculated as: $\text{rel}(q, d) = w_r \cdot s_r + w_p \cdot s_p$,

where s_p is the normalized score of the pre-fetcher and w_s, w_p are learned during training. Given the concerns on the strictness of the ground truth assumption raised in Section 5.4.2, we hypothesize that re-rankers will eventually over-utilize the pre-fetcher score, s_p, when calculating document relevance, $\text{rel}(q, d)$. As shown in Table 5.3, in many cases both relevant and irrelevant documents may have high similarity with the query. This in turn may confuse and therefore degenerate the re-ranker's term matching mechanism, i.e., MLPs or CNNs over term similarity matrices.

DRMM (Guo et al., 2016) uses pre-trained word embeddings to represent query and document terms. A histogram (with each bar showing the frequency of each cosine similarity bucket) capture the cosine similarities of a query term, q_i, with all the terms of a particular document. Then an MLP is fed with the histogram of each query term q_i to produce a document-aware score for each q_i, which is weighted by a gating mechanism assessing the importance of q_i based on its embedding. The sum of the weighted weighted document-aware q_i scores of all the query terms is the relevance score of the document. A caveat of DRMM is that it completely ignores the context of the terms which could be of particular importance in our datasets where texts are long.

PACRR (Hui et al., 2017) represents query and document terms with pre-trained embeddings and calculates a matrix S containing the cosine similarities of all query-document term pairs. A row-wise k-max pooling operation on S keeps the highest similarities per query term (matrix S_k). Then, wide convolutions of different kernel (filter) sizes ($n \times n$) with multiple filters per size are applied on S. Each filter of size $n \times n$ attempts to capture n-gram similarities between queries and documents. A max-pooling operation keeps the strongest signals across filters and a row-wise k-max pooling keeps the strongest signals per query n-gram, resulting in the matrix $S_{n,k}$. Subsequently, a row-wise concatenation of S_k with all $S_{n,k}$ matrices (for different values of n) is performed and a column containing the softmax-normalized idf scores of the query terms is concatenated to the resulting matrix (S_{sim}). In effect, each row of the matrix contains different n-gram based similarity views of the corresponding query term, q_i, along with an idf-based importance score. The relevance score is produced as the last hidden state of an LSTM with one hidden unit, which consumes the rows of S_{sim}. PACRR tries to take into account the context of the query and document terms using n-grams but this context-sensitivity is weak and we do not expect much benefits in our datasets that contain long texts.

BERT-based re-rankers: Recent work tries to exploit BERT to improve re-ranking. Following MacAvaney et al. (2019), we use DRMM and PACRR on top of contextualized BERT embeddings derived from BERT. Based on the results of Figure 5.5, discussed below, we use C-BERT as the most promising BERT model. We call these two models C-BERT-

DRMM and C-BERT-PACRR. We also experiment with two settings depending on whether C-BERT weights are updated (*tuned*) or not (*frozen*) during training.

5.6 Experimental setup

5.6.1 Pre-processing - document denoising

One of the major challenges in DOC2DOC IR, as opposed to traditional IR, is the length of the queries and the documents, which may induce noise (many uninformative words) during retrieval. Thus we applied several filters (stop-word, punctuation, and digits elimination) on both queries and documents and reduce their length by approx. 55% (778 words for UK laws and 1,222 words for EU directives on average). Further on, we filtered both queries and documents by eliminating words with idf score less than the average idf score of the stop-words. Our intuition is that words (e.g., 'regulation', 'EU', 'law', etc.) with such a small idf score are uninformative. Still, the texts are much longer (387 words for UK laws and 631 words for EU directives on average) than the queries used in traditional IR (Table 5.1). As an alternative to drastically decrease the query size, we experimented with using only the title of a legislative act as a query but the results were worse, i.e., approx. 5-20% lower R@100 on average across datasets, indicating that the full-text is more informative, although the information is sparse.

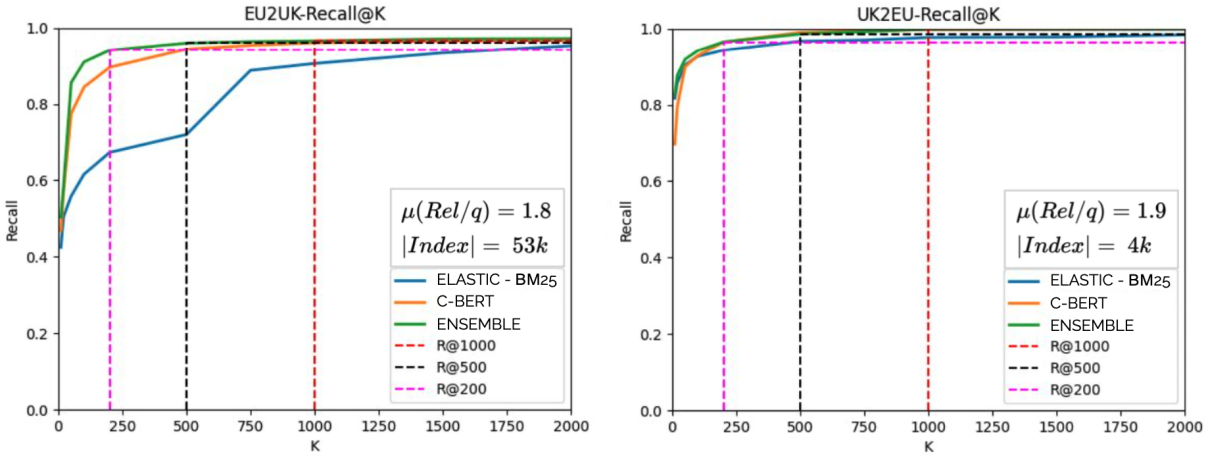


Figure 5.3: Recall@k, where $k \in [0, 2000]$, across the three best pre-fetchers (i.e., BM_{25}, C-BERT and ENSEMBLE) on the development dataset.

5.6.2 Evaluation measures

Pre-fetching aims to bring all the relevant documents in the top-k. Thus we report R@k with k = 100. We observe that after k = 100 the best pre-fetchers have no significant gains

in performance in development data, thus we select k = 100, as a reasonable threshold. For re-ranking we report R@20, nDCG@20 and R-Precision (RP) following the literature Manning et al. (2009).[9] We report the average and standard deviation across three runs on the test set, using the best set of hyper-parameters on development data for neural re-rankers.

5.6.3 In-domain pre-trained word embeddings

As several methods rely on word embeddings, we trained a new WORD2VEC model (Mikolov et al., 2013a) in both corpora (EU and UK legislation) to better accommodate legal language. Preliminary experiments with the W2V-CENT pre-fetcher on validation data showed that domain-specific embeddings perform better than generic 200-dimensional GLOVE embeddings (Pennington et al., 2014) (EU2UK: 66.5 vs. 59.3 R@100 and UK2EU: 72.6 vs. 69.8 R@100).

5.6.4 Pre-trained BERT models

All BERT (pre-fetching) encoders and BERT-based re-rankers use the -BASE version, i.e., 12 layers, 768 hidden units and 12 attention heads, similar to the one of Devlin et al. (2019). All BERT variants (BERT, S-BERT, LEGAL-BERT) are publicly available from Hugging Face (`http://huggingface.co/models`).

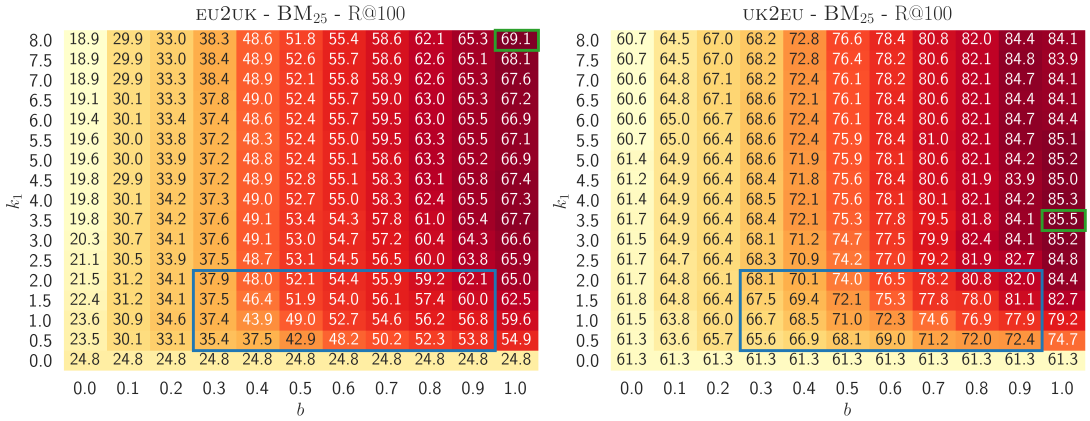


Figure 5.4: Heatmaps showing R@100 for different values of k_1 and b on EU2UK (left) and UK2EU (right). The selected optimal values (green boxes) are outside the proposed ranges in the literature (blue boxes).

[9]The measures have been already defined in Chapter 3

5.6.5 Tuning BM_{25}: The case of DOC2DOC IR

The effectiveness of BM_{25} is highly dependant on properly selecting the values of k_1 and b (Equation 5.1). In traditional (ad-hoc) IR, k_1 is typically evaluated in the range$[0, 3]$ (usually $k_1 \in [0.5, 2.0]$); b needs to be in $[0, 1]$ (usually $b \in [0.3, 0.9]$) (Taylor et al., 2006; Trotman et al., 2014; Lipani et al., 2015). As a general rule of thumb, BM_{25} with k_1=1.2 and b=0.75 seems to give good results in most cases (Trotman et al., 2014). We observe that in the case of DOC2DOC IR, where the queries are much longer, the optimal values are outside the proposed ranges (Figure 5.4). In both datasets the optimal values for k_1 and b are relatively high, favoring terms with high tf, while penalizing long documents. In effect BM_{25} uses k_1 and b as a denoising regularizer to over-utilize highly frequent query terms normalized by document length.

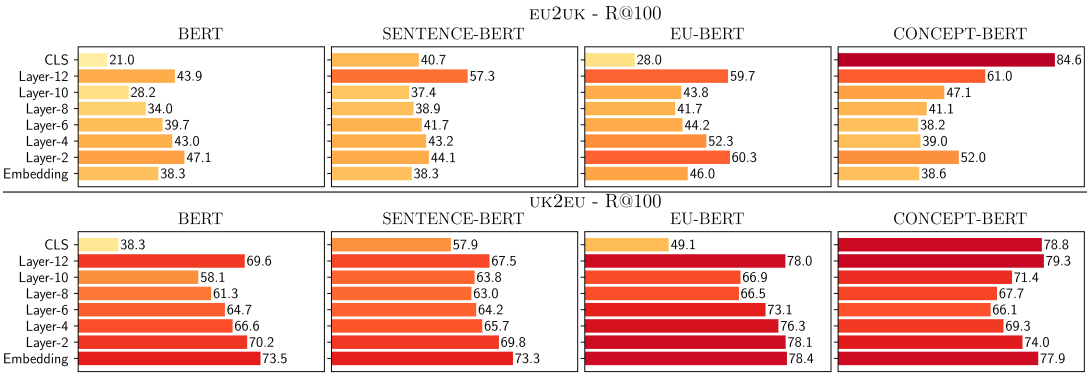


Figure 5.5: Heatbars showing R@100 (on development data) for text representations extracted from different layers of the various BERT-based pre-fetchers we experimented with.

5.6.6 Extracting representations from BERT

Recently there has been a lot of research on understanding the effectiveness of the different layers of BERT across several tasks (Liu et al., 2019a; Hewitt and Manning, 2019; Jawahar et al., 2019; Goldberg, 2019; Kovaleva et al., 2019; Lin et al., 2019). Figure 5.5 shows heatbars comparing representations extracted from different layers of the various BERT-based pre-fetchers we experimented with.[10] EU-BERT and C-BERT, which have been adapted for the legal domain, perform much better than BERT and S-BERT, which were trained on generic corpora. An interesting observation is that the `[cls]` token is a powerful representation only in C-BERT where it was trained to predict EUROVOC concepts. Also, in UK2EU the embedding layer produces the best representations in all BERT variants

[10]Recall that a text can be represented by its `[cls]` token embedding or by the centroid of its token embeddings, which can be extracted from any of the 12 layers of BERT.

except C-BERT, where the embedding layer achieves comparable results to the top-2 representations (`[cls]`, Layer-12). This is an indication that the context in this dataset is not as important as in EU2UK.

5.6.7 Implementation details for neural methods

All neural models were implemented using the TENSORFLOW 2 framework, as in the previous chapter. Hyper-parameters were tuned on development data, using early stopping and the Adam optimizer (Kingma and Ba, 2015) with a fixed learning rate of 1e-3. For neural re-rankers (DRMM, PACRR, and their variants), hyper-parameters were selected by grid-searching the following sets, and selecting the values: CNN hidden units {16, 32}, MLP units {10, 20}, batch size {8, 16, 32}. Concerning C-BERT, we set the dropout rate to 0.1 and grid-search for learning rate {2e-5, 3e-5, 4e-5, 5e-5}, as suggested by Devlin et al. (2019), while batch size was set to 4 due to GPU memory limitations.

5.7 Experimental results

5.7.1 Pre-fetching results

Table 5.4 shows R@100 on the test datasets for the various pre-fetchers considered. On EU2UK, C-BERT is the best method by a large margin, followed by S-BERT and EU-BERT, verifying our assumption that the concept classification task is a good proxy for obtaining rich representations with respect to IR. Both S-BERT and EU-BERT are better than BERT for different reasons. EU-BERT was adapted to the legal domain and is, therefore, able to capture the nuances of the legal language. S-BERT was trained to produce representations suitable for comparing texts with cosine similarity, a task highly related to IR. Nonetheless, having been trained on generic corpora with small texts it performs much worse than C-BERT. Interestingly, BM_{25} is comparable to both S-BERT and EU-BERT despite its simplicity. As expected, combining C-BERT with BM_{25} further improves the results. In UK2EU R@100 is much higher compared to EU2UK probably because of the shortest queries. Also, as discussed in Section 5.6.6, the contextual information is not so critical in this dataset, thus we expect the context unaware BM_{25} and W2V-CENT to perform well. Indeed, BM_{25} achieves the best results followed closely by C-BERT and EU-BERT, while W2V-CENT outperforms S-BERT and BERT. Again the ENSEMBLE improves the results.

Method	EU2UK	UK2EU
	R@100	R@100
BM_{25} (Robertson et al., 1995)	57.5	93.7
W2V-CENT (Brokos et al., 2016)	50.6	88.2
BERT (Devlin et al., 2019)	54.0	85.1
S-BERT (Reimers and Gurevych, 2019)	57.7	84.8
EU-BERT (ours)	57.6	90.1
C-BERT (ours)	83.8	92.9
ENSEMBLE (BM_{25} + C-BERT)	**86.5**	**95.0**

Table 5.4: Pre-fetching results across test datasets.

5.7.2 Re-ranking results

Table 5.5 shows the ranking results on test data for EU2UK and UK2EU. We also report results for BM_{25}, C-BERT, ENSEMBLE and an ORACLE, which re-ranks the top-k documents returned by the pre-fetcher placing all relevant documents at the top. On EU2UK ENSEMBLE performs better than the other two pre-fetchers. Interestingly, neural re-rankers fall short on improving performance and are comparable (or even identical) with ENSEMBLE in most cases, possibly because very similar documents may be relevant or not (Section 5.4.2, Table 5.3), leading to *contradicting* supervision.[11] As we hypothesized (Section 5.5.2), re-rankers over-utilize the pre-fetcher score when calculating document relevance, as a defense mechanism (bias) against contradicting supervision, which eventually leads to the degeneration of the re-ranker's term matching mechanism. Inspecting the corresponding weights of the models, we observe that indeed $w_p >> w_s$ across all methods. This effect seems more intense in BERT-based re-rankers (C-BERT + DRMM or PACRR), especially those that fine-tune C-BERT, possibly because these models perform term matching considering sub-word units, instead of full words. In other words, relying on the neural relevance score (s_r) is catastrophic. Similar observations can be made for UK2EU. In both datasets all methods have a large performance gap compared to the ORACLE, indicating that there is still large room for improvement, possibly utilizing information beyond text, such as the time of publication or relations between regulations and law.

Filtering by year: We have already highlighted the difficulties imposed to our datasets by the frequently amended EU directives (Section 5.4.2, Table 5.3). Also, recall that each EU directive defines a deadline (typically 2 years) for the transposition to take place. On the other hand, as we observe in Figure 5.6, EU directives may already be transposed by earlier legislative acts of member states (the member states act in a proactive manner), or the states may delay the transposition for political reasons. In effect, the relevance of

[11] By *contradicting* supervision we mean similar training query-document pairs with opposite labels.

Method	EU2UK					UK2EU				
	w_p	w_s	R@20	nDCG@20	RP	w_p	w_s	R@20	nDCG@20	RP
BM$_{25}$	-	-	45.8	34.4	25.5	-	-	87.5	66.8	**49.4**
C-BERT (ours)	-	-	55.7	37.9	21.8	-	-	79.7	53.0	33.1
ENSEMBLE (BM$_{25}$ + C-BERT)	-	-	54.1	43.1	29.6	-	-	88.0	**67.7**	49.3
+ DRMM	+1.1	-0.8	**59.9** (± 3.2)	41.7 (± 2.4)	24.3 (± 2.9)	+1.3	-0.8	86.3 (± 1.1)	61.6 (± 1.1)	40.1 (± 1.5)
+ PACRR	+4.2	+0.6	54.3 (± 0.2)	**43.3** (± 0.2)	**30.1** (± 0.4)	+4.0	+0.1	88.0 (± 0.0)	**67.7** (± 0.0)	49.3 (± 0.0)
+ C-BERT-DRMM *(frozen)*	+3.3	-1.6	57.9 (± 3.4)	43.1 (± 0.3)	27.3 (± 2.2)	+3.5	-1.0	88.3 (± 0.4)	67.3 (± 0.6)	48.5 (± 1.3)
+ C-BERT-PACRR *(frozen)*	+4.6	+0.9	54.1 (± 0.0)	43.1 (± 0.0)	29.6 (± 0.0)	+2.9	-0.9	**89.6** (± 0.4)	66.5 (± 0.5)	46.0 (± 0.9)
+ C-BERT-DRMM *(tuned)*	+1.9	-0.5	54.1 (± 0.0)	43.1 (± 0.0)	29.6 (± 0.0)	+1.2	+0.5	88.0 (± 0.0)	**67.7** (± 0.0)	49.3 (± 0.0)
+ C-BERT-PACRR *(tuned)*	+1.8	-0.6	54.1 (± 0.0)	43.1 (± 0.0)	29.6 (± 0.0)	+2.0	+2.1	88.0 (± 0.0)	**67.7** (± 0.0)	49.3 (± 0.0)
+ ORACLE	-	-	86.5	87.7	86.5	-	-	95.0	95.3	95.0
Applying date filtering on top of predictions										
Year range	±5 years					±15 years				
ENSEMBLE (BM$_{25}$ + C-BERT)	-	-	76.6	54.6	37.1	-	-	**86.2**	**68.2**	**50.0**
+ DRMM (*pre-filtering*)	+1.1	-0.8	**81.4**	**56.5**	35.4	+1.3	-0.8	85.3	62.6	42.3
+ DRMM (*post-filtering*)	+1.1	-0.8	75.7	49.2	31.1	+1.3	-0.8	83.6	63.5	44.2
+ PACRR (*pre-filtering*)	+4.2	+0.6	76.6	54.8	**37.6**	+4.0	+0.1	**86.2**	**68.2**	**50.0**
+ PACRR (*post-filtering*)	+4.2	+0.6	74.2	52.9	36.5	+4.0	+0.1	85.5	67.6	49.6

Table 5.5: Re-ranking results across test datasets. The upper zone shows the results of neural re-rankers on top of the best pre-fetchers with respect to (w_s, w_p). It also reports re-ranking results of the best pre-fetchers. The lower zone reports the re-ranking results after applying temporal filtering.

a document to a query depends both on the textual content and the time the laws were published. Thus, we filter out documents that are outside a predefined distance (in years) from the query in two ways, *pre-filtering* and *post-filtering*. Pre-filtering is applied to the pre-fetcher, i.e., prior to re-ranking, while post-filtering is applied after the re-ranking. Note that our main goal is to improve re-ranking. We thus apply the filtering scheme to the ENSEMBLE, DRMM and PACRR. The lower zone of Table 5.5 shows the results of the whole process. In EU2UK, the hardest out of the two datasets, the time filtering has a positive impact, improving the results by a large margin. On the other hand, filtering seems to have a minor effect in UK2EU.

5.7.3 EU2UK ≠ UK2EU

Across experiments, we observe that best practices vary between the EU2UK and UK2EU datasets. EU2UK benefits from C-BERT representations, while in UK2EU context-unaware and domain-agnostic BM$_{25}$ has comparable or better performance than C-BERT. Similarly, we observe that time filtering further improves the performance in EU2UK, while we have a contradicting effect in UK2EU. Given the overall results, we conclude the two datasets have quite different characteristics. Thus, it is important to consider both EU2UK and UK2EU independently, although one may initially consider them to be symmetric.

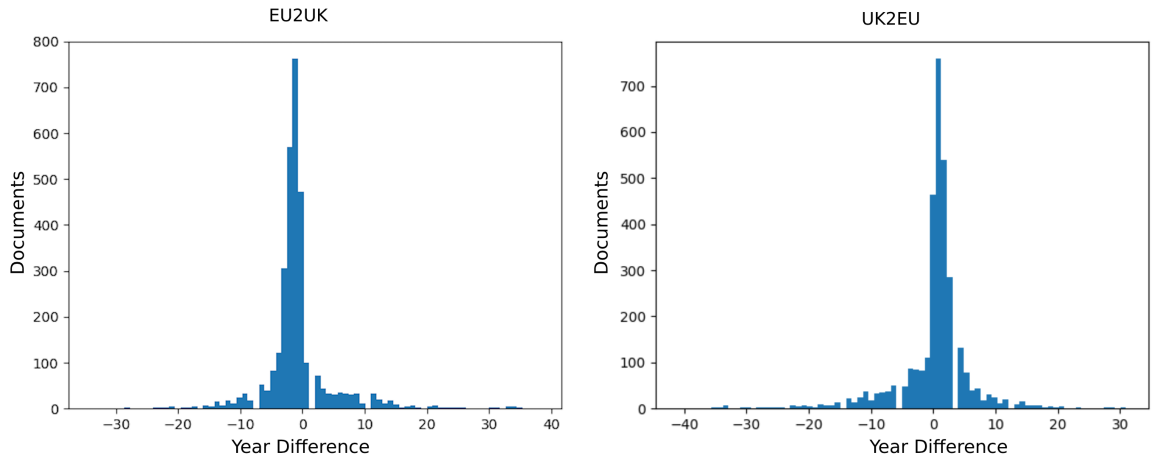


Figure 5.6: Relevant documents according to their chronological difference (in years) with the query.

5.8 Conclusions

In this chapter, we discussed the challenging application of regulatory information retrieval (REG-IR), a document-to-document information retrieval (DOC2DOC IR) task, where both queries and documents are extremely long, comparing to standard IR benchmarks. Given the nature of the data (long documents), we showed that BM_{25}, a standard pre-fetching method, needs careful tuning, choosing parameters out of the current standards of conventional ad-hoc retrieval. Considering various BERT-based models as pre-fetching alternatives, we also found that fine-tuning BERT in an in-domain classification task leads to a vast improvement in one out of two datasets. Alongside BM_{25}, these are the two best pre-fetchers, that can be further exploited with an ENSEMBLE to provide the best pre-fetching document retrieval performance. Interestingly, we showed that neural re-rankers improve performance only marginally, if at all, while they vastly rely on the pre-fetching scores, provided by the ENSEMBLE method. Based on these findings, we hypothesize that document relevance is affected also by other factors, thus we examined and empirically confirmed that the time dimension is also a very important aspect in one out of two datasets.

In future work, we would like to experiment with more tasks and datasets to have a better and more universal understanding of different aspects in DOC2DOC IR. In this direction, we would like to experiment with the interesting task of case law retrieval in ECtHR cases (Chapter 4). Moreover, as we identified the limitations of current neural re-ranking methods, we believe that there are three very interesting directions: (a) document de-noising, where sentence selections methods, like those described in Section 4.5.3, could be employed to replace naive statistical methods, like TF-IDF scores; (b) neural re-rankers

comparing paragraph instead of word embeddings; and (c) consideration of additional information, such as the relations (e.g., amendments, references, etc.) between documents or better use of information like the time dimension, which was used in a naive fashion in our study.

Chapter 6

Conclusions, Limitations and Future Work

We presented a thorough analysis of several legal tasks (contract element extraction, obligation extraction, neural judgment prediction and rationale extraction, legal information retrieval) across contracts, legislation, and court cases from various jurisdictions, namely US, EU and UK, in English.

Our project targeted two main research questions; first and foremost on the adaptability of neural methods that have been proposed for relevant NLP tasks in other domains and how they are affected by legal language, writing, and structure; and second on providing explanations of the decisions(predictions) of neural models. Considering the first research question, we highlighted several cases where either legal language affects a model's performance (Chapter 2.4) or suitable modeling of the document structure is needed (Chapters 2.5 and 4.4). To this end, we pre-trained and used in-domain word representations and neural language models, while we also proposed new methods with state-of-the-art performance, e.g., X-BILSTM-ATT for obligation extraction (Chapter 2.5), BERT-LWAN for legal topic classification (Chapter 3), and HIER-BERT for legal judgment prediction (Chapter 4). With respect to model explainability, we initially experimented with saliency (attention) heat-maps and highlighted their limitation as a means for the explanation of a model's decisions in legal judgment prediction, where we further studied rationale extraction techniques as a prominent methodology for explainability (Chapter 4). In lack of publicly available annotated datasets, in order to experiment with deep learning methods, we curated and published five datasets for various legal tasks (contract element extraction, legal topic classification, legal judgment prediction and rationale extraction, legal information retrieval), while we also published legal word embeddings (LAW2VEC) and a legal pre-trained language model (LEGAL-BERT) for further re-use to assist legal text processing research.

In the following paragraphs, we present our detailed conclusions, limitations and directions for future work per chapter.

In Chapter 2, we investigated two applications of information extraction for contracts. In contract element extraction, we found that BILSTM-based models lead to state-of-the-art results, even comparing to pre-trained TRANSFORMER-based models, e.g., BERT, that currently dominate the NLP literature. We linked this finding with the lack of inhered re-currency in the TRANSFORMERs architecture. Moreover, we observed that in-domain knowledge, as expressed in language and captured by in-domain WORD2VEC embeddings and LEGAL-BERT, leads to performance improvements in two out of three cases we considered. In the second task, obligation extraction, we found that correct modeling of the text structure with hierarchical LSTMs leads to the best results with vast performance improvements in the list item categories, but also in the rest of the categories (stand-alone sentences). This finding highlights the importance of context in the form of inter-sentence relations in this sentence classification task. In future work, we would like to further investigate the impact of the lack of recurrency in TRANSFORMER-based models in contract element extraction with a qualitative analysis by identifying examples to better understand this phenomenon. With respect to classification performance, an obvious direction would be to train a BERT encoder, specialized solely on contractual text that could possibly benefit the overall performance. Similarly, it would be interesting to explore the use of TRANSFORMER-based models in the second task of obligation extraction by employing a similar method to hierarchical BERT (HIER-BERT), discussed in Section 4.4.3.

In Chapter 3, we presented an extensive study of Large Scale Multi-label Text Classification (LMTC) considering the task of tagging EU legislation with EUROVOC concepts, to answer three understudied questions on (1) the competitiveness of PLT-based methods against neural models, (2) the use of the label hierarchy in neural methods, and (3) the benefits from neural transfer learning. A condensed summary of our findings is that (1) TF-IDF PLT-based methods are definitely worth considering, but are not always competitive, while ATTENTION-XML, a neural PLT-based method that captures word order, is robust across datasets; (2) transfer learning leads to state-of-the-art results, especially combined with a label-wise attention mechanism. Considering datasets from other domains (MIMIC-III, AMAZON13K), we found that no single method is best across all domains and label groups (all, few, zero) as the language, the size of documents, and the label assignment strongly vary with direct implications in the performance of each method. In future work, we would like to further exploit information from the label hierarchy in TRANSFORMER-based methods. In follow up work (Manginas et al., 2020), we found that mapping label hierarchy levels across BERT layers further improves the

classification performance in EURLEX57K. In another direction, we would like to further investigate few and zero-shot learning in LMTC, especially in BERT models that are currently unable to cope with zero-shot labels. BERT uses sub-word units, instead of full tokens, and its sub-word units are contextualized and probably richer than typical WORD2VEC word embeddings; thus, they could benefit zero-shot capable LWANs. In a completely different set-up, one could follow the recent work of Wenpeng Yin and Roth (2019) that models zero-shot classification as a natural language inference task, considering a document-label pair at the same time.

In Chapter 4,we discussed the challenging application of legal judgment prediction. Contrary to our findings in LMTC with the EURLEX57K dataset (Section 3.6), in ECtHR cases one has to consider the full text of the documents (court cases). As court cases are extremely long and modern pre-trained transformer-based are limited to processing up to 512 tokens at most, we proposed a new state-of-art method HIER-BERT to tackle this issue, while effectively modeling the natural document structure, i.e., list of paragraphs. For the first time, we considered three downstream tasks (binary and multi-label classification, case importance prediction), highlighting the escalated difficulties in the latter two cases that have not been considered in the literature. Further on, we studied the extent of model bias in demographic information, where both quantitative (classification performance drop) and qualitative (attention to demographics) evidence highlighted that model bias exists, but only slightly affects the predictions of the models on average. Considering the limitations of studying soft attention scores as a means for explainability, we followed recent work in rationale extraction and introduced a new task, paragraph-level rationale extraction, where a model has to extract the most relevant paragraphs of a court case with respect to the alleged violations in the case. For this purpose, we proposed a new model, similar to HIER-BERT, using hard attention masking with additional regularizers (HIERBERT-HA). We studied several alternatives following the literature (Lei et al., 2016; Yu et al., 2019), and proposed a new regularizer, which led to state-of-the-art performance in both faithfulness and rationale quality measures proposed by DeYoung et al. (2020). In the future, we would like to consider a pipe-lined framework, where alleged violation prediction could be a first module, providing primary information for the more difficult task of judgment prediction in the realistic and challenging multi-label setup (predicting particular article violations). In the same direction, additional system modules trained to predict case law relevant to a given case, could also provide valuable information for the aforementioned task, which is also part of the court reasoning considering the facts, the law and the case law on par. In the case of explainability, which is a core feature, to provide formal explanations of the overall system's decisions, more structured explanations closer to human reasoning should be constructed. In this direction, modeling

the inter-paragraph relations is a crucial characteristic to provide factual patterns, i.e., paragraph X1 in light of paragraph X2 and X3 leads to conclusion Y, instead of factual masks that are currently extracted by the methods described in Section 4.5.3. Last but not least, a more extensive in-depth analysis on model bias identification and model debiasing would be an important step towards improving fairness, which may also lead to improvements in faithfulness and rationale quality, as the model will then be unable to rely on biased spurious information.

In Chapter 5, we discussed the challenging application of regulatory information retrieval (REG-IR), a document-to-document information retrieval (DOC2DOC IR) task, where both queries and documents are extremely long, comparing to standard IR benchmarks. Given the nature of the data (long documents), we showed that BM_{25}, a standard pre-fetching method, needs careful tuning, choosing parameters out of the current standards of conventional ad-hoc retrieval. Considering various BERT-based models as pre-fetching alternatives, we also found that fine-tuning BERT in an in-domain classification task leads to a vast improvement in the most difficult of the two datasets. Alongside BM_{25}, these are the two best pre-fetchers, that can be further exploited with a linear combination of their scores (ENSEMBLE) to provide the best pre-fetching document retrieval performance. Interestingly, we showed that neural re-rankers improve performance only marginally, if at all, while they vastly rely on the pre-fetching scores, provided by the ENSEMBLE method. Based on these findings, we hypothesize that document relevance is affected also by other factors, thus we examined and empirically confirmed that the time dimension is also a very important aspect in the most difficult of the two datasets, where the document pool is much larger and temporal filtering substantially reduces candidate relevant documents. In future work, we would like to experiment with more tasks and datasets to have a better and more universal understanding of different aspects in DOC2DOC IR. In this direction, we would like to experiment with the interesting task of case law retrieval in ECtHR cases. Moreover, as we identified the limitations of current neural re-ranking methods, we believe that there are three very interesting directions: (a) document de-noising, where sentence selections methods, like those described in Section 4.5.3, could be employed to replace naive statistical methods, like TF-IDF scores; (b) neural re-rankers comparing paragraph instead of word embeddings; and (c) consideration of additional information, such as the relations (e.g., amendments, references, etc.) between documents or better use of information like the time dimension, which was used in a naive fashion in our study.

Bibliography

A. Akbik, T. Bergmann, and R. Vollgraf. Pooled contextualized embeddings for named entity recognition. In *Proceedings of the Annual Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT)*, pages 724–728, Minneapolis, Minnesota, 2019.

N. Aletras et al. Predicting judicial decisions of the European Court of Human Rights: a Natural Language Processing perspective. *PeerJ Computer Science*, 2:e93, 2016.

E. Alsentzer, J. Murphy, W. Boag, W.-H. Weng, D. Jin, T. Naumann, and M. McDermott. Publicly available clinical BERT embeddings. In *Proceedings of the 2nd Clinical Natural Language Processing Workshop*, pages 72–78, Minneapolis, Minnesota, USA, June 2019.

D. Alvarez-Melis and T. Jaakkola. A causal framework for explaining the predictions of black-box sequence-to-sequence models. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 412–421, Copenhagen, Denmark, Sept. 2017.

J. Angwin, J. Larson, S. Mattu, and L. Kirchner. Machine bias: There's software used across the country to predict future criminals. and it's biased against blacks. *ProPublica*, 2016. URL `https://www.propublica.org/article/machine-bias-risk-assessments-in-criminal-sentencing`.

K. Ashley. *Artificial Intelligence and Legal Analytics: New Tools for Law Practice in the Digital Age*. Cambridge University Press, 2017.

K. Asooja, G. Bordea, G. Vulcu, L. O'Brien, A. Espinoza, E. Abi-Lahoud, P. Buitelaar, and T. Butler. Semantic annotation of finance regulatory text using multilabel classification. In *Proceedings of the International Workshop on Legal Domain and Semantic Web Applications*, Portoroz, Slovenia, 2015.

J. L. Ba, J. R. Kiros, and G. E. Hinton. Layer normalization, 2016.

D. Bahdanau, K. Cho, and Y. Bengio. Neural machine translation by jointly learning to align and translate. In *Proceedings of the 5th International Conference on Learning Representations (ICLR)*, San Diego, CA, USA, 2015.

W. Barfield. *The Cambridge Handbook of the Law of Algorithms*. Cambridge Law Handbooks. Cambridge University Press, 2020.

J. Bastings, W. Aziz, and I. Titov. Interpretable neural predictions with differentiable binary variables. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 2963–2977, Florence, Italy, July 2019.

I. Beltagy, K. Lo, and A. Cohan. Scibert: A pretrained language model for scientific text. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2019.

Y. Bengio, R. Ducharme, P. Vincent, and C. Janvin. A neural probabilistic language model. *J. Mach. Learn. Res.*, 3(null):1137–1155, Mar. 2003. ISSN 1532-4435.

K. Bhatia, H. Jain, P. Kar, M. Varma, and P. Jain. Sparse Local Embeddings for Extreme Multi-label Classification. In C. Cortes, N. D. Lawrence, D. D. Lee, M. Sugiyama, and R. Garnett, editors, *Advances in Neural Information Processing Systems 28*, pages 730–738. Curran Associates, Inc., 2015.

V. K. Bhatia. Analysing genre: Language use in professional settings. *London Longman*, 16, 12 1994.

C. Biagioli, E. Francesconi, A. Passerini, S. Montemagni, and C. Soria. Automatic semantics extraction in law documents. In *Proceedings of the 10th International Conference on Artificial Intelligence and Law (ICAIL)*, pages 133–140, Bologna, Italy, 2005.

D. M. Bikel, R. Schwartz, and R. M. Weischedel. An algorithm that learns what's in a name. *Machine Learning*, 34(1–3):211–231, 1999.

R. Binns, M. Van Kleek, M. Veale, U. Lyngs, J. Zhao, and N. Shadbolt. 'it's reducing a human being to a percentage': Perceptions of justice in algorithmic decisions. In *CHI*, pages 377:1–377:14, 2018.

P. Bojanowski, E. Grave, A. Joulin, and T. Mikolov. Enriching Word Vectors with Subword Information. *arXiv preprint arXiv:1607.04606*, 2016.

G. Brokos, P. Malakasiotis, and I. Androutsopoulos. Using centroids of word embeddings and word mover's distance for biomedical document retrieval in question answering. In

Proceedings of the 15th Workshop on Biomedical Natural Language Processing (BioNLP 2016), at the 54th Annual Meeting of the Association for Computational Linguistics (ACL 2016), pages 114–118, Berlin, Germany, 2016.

D. Cer, M. Diab, E. Agirre, I. Lopez-Gazpio, and L. Specia. SemEval-2017 task 1: Semantic textual similarity multilingual and crosslingual focused evaluation. In *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)*, pages 1–14, Vancouver, Canada, 2017.

I. Chalkidis and I. Androutsopoulos. A deep learning approach to contract element extraction. In *Proceedings of the 30th International Conference on Legal Knowledge and Information Systems (JURIX)*, pages 155–164, Luxembourg, 2017.

I. Chalkidis and D. Kampas. Deep Learning in Law: Early Adaptation and Legal Word Embeddings Trained on Large Corpora. *Artificial Intelligence and Law*, 27(2):171–198, June 2019. ISSN 0924-8463.

I. Chalkidis, I. Androutsopoulos, and A. Michos. Extracting contract elements. In *Proceedings of the 16th International Conference on Artificial Intelligence and Law (ICAIL)*, pages 19–28, London, UK, 2017.

I. Chalkidis, I. Androutsopoulos, and A. Michos. Obligation and Prohibition Extraction Using Hierarchical RNNs. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (ACL)*, Melbourne, Australia, 2018.

I. Chalkidis, I. Androutsopoulos, and N. Aletras. Neural Legal Judgment Prediction in English. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics (ACL)*, Florence, Italy, 2019a.

I. Chalkidis, M. Fergadiotis, P. Malakasiotis, N. Aletras, and I. Androutsopoulos. Extreme Multi-Label Legal Text Classification: A case study in EU Legislation. In *Proceedings of the Annual Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT)*, Minneapolis, USA, 2019b.

I. Chalkidis, M. Fergadiotis, P. Malakasiotis, and I. Androutsopoulos. Large-Scale Multi-Label Text Classification on EU Legislation. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics (ACL)*, Florence, Italy, 2019c.

I. Chalkidis, M. Fergadiotis, P. Malakasiotis, and I. Androutsopoulos. Neural Contract Element Extraction Revisited. In *Proceedings of the Document Intelligence Workshop of*

the 33rd Conference on Neural Information Processing Systems (NeurIPS), Vancouver, Canada, 2019d.

I. Chalkidis, M. Fergadiotis, S. Kotitsas, P. Malakasiotis, N. Aletras, and I. Androutsopoulos. An empirical study on large-scale multi-label text classification including few and zero-shot labels. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 7503–7515, Online, 2020a.

I. Chalkidis, M. Fergadiotis, P. Malakasiotis, N. Aletras, and I. Androutsopoulos. LEGAL-BERT: The muppets straight out of law school. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 2898–2904, Online, 2020b.

I. Chalkidis, M. Fergadiotis, N. Manginas, E. Katakalou, and P. Malakasiotis. Regulatory Compliance through Doc2Doc Information Retrieval: A case study in EU/UK legislation where text similarity has limitations. In *16th Conference of the European Chapter of the Association for Computational Linguistics (EACL 2021)*, Online, 2021a.

I. Chalkidis, D. Tsarapatsanis, M. Fergadiotis, P. Malakasiotis, N. Aletras, and I. Androutsopoulos. Paragraph-level Rationale Extraction through Regularization: A case study on European Court of Human Rights Case. In *Proceedings of the Annual Conference of the North American Chapter of the Association for Computational Linguistics (NAACL 2021)*, Online, 2021b.

S. Chang, Y. Zhang, M. Yu, and T. S. Jaakkola. A Game Theoretic Approach to Class-wise Selective Rationalization. In *Advances in Neural Information Processing Systems (NeurIPS)*, Vancouver, Canada, 2019.

E. Charniak. *Introduction to Deep Learning*. The MIT Press, 2019. ISBN 0262039516, 9780262039512.

J. Cheng, L. Dong, and M. Lapata. Long short-term memory-networks for machine reading. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 551–561, Austin, Texas, Nov. 2016. doi: 10.18653/v1/D16-1053.

W.-T. M. Chiang, M. Hagenbuchner, and A. C. Tsoi. The wt10g dataset and the evolution of the web. In *Special Interest Tracks and Posters of the 14th International Conference on World Wide Web*, WWW '05, page 938–939, New York, NY, USA, 2005. Association for Computing Machinery. ISBN 1595930515.

K. Clark, U. Khandelwal, O. Levy, and C. D. Manning. What does BERT look at? an analysis of BERT's attention. In *Proceedings of the 2019 ACL Workshop BlackboxNLP:*

Analyzing and Interpreting Neural Networks for NLP, pages 276–286, Florence, Italy, Aug. 2019.

C. Clarke, N. Craswell, and I. Soboroff. Overview of the trec 2004 terabyte track. In *TREC*, 2004.

N. Cristianini and J. Shawe-Taylor. *An Introduction to Support Vector Machines and other kernel-based learning methods*. Cambridge University Press, 2000.

M. Curtotti and E. Mccreath. Corpus based classification of text in Australian contracts. In *Proceedings of the Australasian Language Technology Association Workshop*, pages 18–26, Melbourne, Australia, 2010.

M. Cutts. *Oxford guide to plain English*. Oxford University Press, 2009.

J. Devlin, M. Chang, K. Lee, and K. Toutanova. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *Proceedings of the Annual Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT)*, 2019.

J. DeYoung, S. Jain, N. F. Rajani, E. Lehman, C. Xiong, R. Socher, and B. C. Wallace. ERASER: A benchmark to evaluate rationalized NLP models. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4443–4458, Online, July 2020.

F. Doshi-Velez and B. Kim. Towards a rigorous science of interpretable machine learning. *arXiv*, 2017.

C. Dozier, R. Kondadadi, M. Light, A. Vachher, S. Veeramachaneni, and R. Wudali. Named entity recognition and resolution in legal text. In E. Francesconi, S. Montemagni, W. Peters, and D. Tiscornia, editors, *Semantic Processing of Legal Texts*, number 6036 in Lecture Notes in AI, pages 27–43. Springer, 2010.

J. Dressel and H. Farid. The accuracy, fairness, and limits of predicting recidivism. *Science Advances*, 4(10), 2018. URL `https://advances.sciencemag.org/content/4/1/eaao5580`.

Y. Fan, J. Guo, Y. Lan, J. Xu, C. Zhai, and X. Cheng. Modeling Diverse Relevance Patterns in Ad-Hoc Retrieval. In *The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval*, SIGIR '18, page 375–384, New York, NY, USA, 2018. Association for Computing Machinery. ISBN 9781450356572.

R. Fergus. *Contract Law*. Round Hall, 2006.

N. E. I. Gado, E. Grall-Maës, and M. Kharouf. Linear discriminant analysis for large-scale data: Application on text and image data. In *2016 15th IEEE International Conference on Machine Learning and Applications (ICMLA)*, pages 961–964, 2016.

P. Gage. A new algorithm for data compression. *C Users Journal*, 12(2):23–38, 1994.

X. Gao, M. P. Singh, and P. Mehra. Mining business contracts for service exceptions. *IEEE Transactions on Services Computing*, 5:333–344, 2012.

F. Gers and J. Schmidhuber. Lstm recurrent networks learn simple context-free and context-sensitive languages. *IEEE transactions on neural networks*, 12 6:1333–40, 2001.

X. Glorot and Y. Bengio. Understanding the difficulty of training deep feedforward neural networks. In *Proceedings of the International Conference on Artificial Intelligence and Statistics*, pages 249–256, Sardinia, Italy, 2010.

Y. Goldberg. *Neural Network Methods in Natural Language Processing.* Morgan and Claypool Publishers, 2017.

Y. Goldberg. Assessing BERT's Syntactic Abilities. *CoRR*, abs/1901.05287, 2019.

I. Goodfellow, Y. Bengio, and A. Courville. *Deep Learning.* MIT Press, 2016.

I. J. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio. Generative adversarial nets. In *arXiv*, 2014.

A. Graves, N. Jaitly, and A. Mohamed. Hybrid speech recognition with deep bidirectional LSTM. In *Proceedings of the IEEE Workshop on Automatic Speech Recognition and Understanding*, pages 273–278, Olomouc, Czech Republic, 2013.

A. Grover and J. Leskovec. node2vec: Scalable Feature Learning for Network. *KDD, ACM*, page 855–864, 2016.

J. Guo, Y. Fan, Q. Ai, and W. B. Croft. A deep relevance matching model for ad-hoc retrieval. In *Proceedings of the 25th ACM International on Conference on Information and Knowledge Management*, pages 55—-64, Indianapolis, IN, USA, 2016.

R. Haigh. *Legal English.* Routledge, 2018.

D. J. D. J. Harris. *Harris, O'Boyle, and Warbrick : Law of the European Convention on Human Rights.* Oxford University Press, 4rd edition. edition, 2018. ISBN 019878516.

I. Hasan, J. Parapar, and R. Blanco. Segmentation of legislative documents using a domain-specific lexicon. In *Proceedings of the 19th International Conference on Database and Expert Systems Application*, pages 665–669, Turin, Italy, 2008.

K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 770–778, 2016. doi: 10.1109/CVPR.2016.90.

J. Hewitt and C. D. Manning. A structural probe for finding syntax in word representations. 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 4129–4138, Minneapolis, Minnesota, June 2019.

F. Hill, A. Bordes, S. Chopra, and J. Weston. The goldilocks principle: Reading children's books with explicit memory representations, 2016.

S. Hochreiter and J. Schmidhuber. Long short-term memory. *Neural Computation*, 9(8): 1735–1780, 1997.

Z. Hu, X. Li, C. Tu, Z. Liu, and M. Sun. Few-shot charge prediction with discriminative legal attributes. In *Proceedings of the International Conference on Computational Linguistics (COLING)*, pages 487–498, 2018.

P.-S. Huang, H. Zhang, R. Jiang, R. Stanforth, J. Welbl, J. Rae, V. Maini, D. Yogatama, and P. Kohli. Reducing sentiment bias in language models via counterfactual evaluation. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 65–83, Online, Nov. 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.findings-emnlp.7. URL `https://www.aclweb.org/anthology/2020.findings-emnlp.7`.

Z. Huang, W. Xu, and K. Yu. Bidirectional LSTM-CRF Models for Sequence Tagging. *Arxiv*, 2015.

K. Hui, A. Yates, K. Berberich, and G. de Melo. PACRR: A position-aware neural IR model for relevance matching. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 1049–1058, Copenhagen, Denmark, Sept. 2017.

K. V. Indukuri and P. R. Krishna. Mining e-contract documents to classify clauses. In *Proceedings of the 3rd Annual ACM Bangalore Conference*, pages 7:1–7:5, Bangalore, India, 2010.

L. Irani, M. Mitchell, D. Robinson, and S. Kannan, editors. *Proceedings of the Conference on Fairness, Accountability, and Transparency (FAccT 2021)*, Online, 2021. Association for Computing Machinery.

A. Jacovi and Y. Goldberg. Towards faithfully interpretable NLP systems: How should we define and evaluate faithfulness? In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4198–4205, Online, July 2020.

S. Jain and B. C. Wallace. Attention is not Explanation. In *Proceedings of the Annual Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT)*, pages 3543–3556, Minneapolis, Minnesota, 2019.

S. Jain, S. Wiegreffe, Y. Pinter, and B. C. Wallace. Learning to faithfully rationalize by construction. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4459–4473, Online, July 2020.

G. Jawahar, B. Sagot, and D. Seddah. What Does BERT Learn about the Structure of Language? In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3651–3657, Florence, Italy, July 2019.

A. E. Johnson, D. J. Stone, L. A. Celi, and T. J. Pollard. MIMIC-III, a freely accessible critical care database. *Nature*, 2017.

N. Kalchbrenner and P. Blunsom. Recurrent continuous translation models. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 1700–1709, Seattle, Washington, USA, Oct. 2013.

N. Kalchbrenner, L. Espeholt, K. Simonyan, A. van den Oord, A. Graves, and K. Kavukcuoglu. Neural Machine Translation in Linear Time. *Arxiv*, 2016.

Y. Kano, M.-Y. Kim, R. Goebel, and K. Satoh. Overview of coliee 2017. In *COLIEE@ ICAIL*, pages 1–8, 2017.

Y. Kano, M.-Y. Kim, M. Yoshioka, Y. Lu, J. Rabelo, N. Kiyota, R. Goebel, and K. Satoh. Coliee-2018: Evaluation of the competition on legal information extraction and entailment. In *JSAI International Symposium on Artificial Intelligence*, pages 177–192. Springer, 2018.

N. Kassner and H. Schütze. Negated and misprimed probes for pretrained language models: Birds can talk, but cannot fly. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7811–7818, Online, July 2020.

D. M. Katz. Quantitative legal prediction-or-how I learned to stop worrying and start preparing for the data-driven future of the legal services industry. *Emory Law Journal*, 62:909, 2012.

S. Khandagale, H. Xiao, and R. Babbar. Bonsai - Diverse and Shallow Trees for Extreme Multi-label Classification. *CoRR*, abs/1904.08249, 2019.

D. Khashabi, S. Chaturvedi, M. Roth, S. Upadhyay, and D. Roth. Looking beyond the surface:a challenge set for reading comprehension over multiple sentences. In *Proceedings of the Annual Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT)*, 2018.

O. Khattab and M. Zaharia. ColBERT: Efficient and Effective Passage Search via Contextualized Late Interaction over BERT, 2020.

M.-Y. Kim, R. Goebel, and S. Ken. Coliee-2015: evaluation of legal question answering. In *Ninth International Workshop on Juris-informatics (JURISIN 2015)*, 2015a.

M.-y. Kim, Y. Xu, and R. Goebel. A Convolutional Neural Network in Legal Question Answering. *Ninth International Workshop on Juris-informatics (JURISIN)*, 2015b.

M.-Y. Kim, R. Goebel, Y. Kano, and K. Satoh. Coliee-2016: evaluation of the competition on legal information extraction and entailment. In *International Workshop on Juris-informatics (JURISIN 2016)*, 2016a.

M.-y. Kim, Y. Xu, Y. Lu, and R. Goebel. Legal Question Answering Using Paraphrasing and Entailment Analysis. *COLIEE Workshop on Juris-informatics (JURISIN)*, 2016b.

Y. Kim. Convolutional neural networks for sentence classification. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1746–1751, Doha, Qatar, Oct. 2014.

D. P. Kingma and J. Ba. Adam: A method for stochastic optimization. In *Proceedings of the 5th International Conference on Learning Representations (ICLR)*, San Diego, CA, USA, 2015.

N. Kiyavitskaya, N. Zeni, T. D. Breaux, A. I. Antón, J. R. Cordy, L. Mich, and J. Mylopoulos. Automating the extraction of rights and obligations for regulatory compliance. In *Proceedings of the 27th International Conference on Conceptual Modeling*, pages 154–168, Barcelona, Spain, 2008.

G. Klambauer, T. Unterthiner, A. Mayr, and S. Hochreiter. Self-normalizing neural networks. In *31th Int. Conf. on Neural Information Processing Systems*, 2017.

S. Kotitsas, D. Pappas, I. Androutsopoulos, R. McDonald, and M. Apidianaki. Embedding Biomedical Ontologies by Jointly Encoding Network Structure and Textual Node

Descriptors. In *Proceedings of the 18th BioNLP Workshop and Shared Task*, pages 298–308, Florence, Italy, 2019.

O. Kovaleva, A. Romanov, A. Rogers, and A. Rumshisky. Revealing the dark secrets of BERT. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 4365–4374, Hong Kong, China, Nov. 2019.

M. Kusner, Y. Sun, N. Kolkin, and K. Weinberger. From word embeddings to document distances. In F. Bach and D. Blei, editors, *Proceedings of the International Conference on Machine Learning*, volume 37 of *Proceedings of Machine Learning Research*, pages 957–966, Lille, France, 2015.

J. Lafferty, A. McCallum, and F. Pereira. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *Proceedings of the International Conference on Machine Learning (ICML)*, Williamstown, MA, USA, 2001.

G. Lample, M. Ballesteros, S. Subramanian, K. Kawakami, and C. Dyer. Neural architectures for named entity recognition. In *Proceedings of the Annual Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT)*, pages 260–270, San Diego, California, 2016.

R. C. Lawlor. What computers can do: Analysis and prediction of judicial decisions. *American Bar Association Journal*, pages 337–344, 1963.

Y. LeCun and Y. Bengio. *Convolutional Networks for Images, Speech, and Time Series*, page 255–258. MIT Press, Cambridge, MA, USA, 1998. ISBN 0262511029.

T. Lei, R. Barzilay, and T. Jaakkola. Rationalizing Neural Predictions. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 107–117, Austin, Texas, 2016.

D. D. Lewis, Y. Yang, T. G. Rose, and F. Li. RCV1: A New Benchmark Collection for Text Categorization Research. *J. Mach. Learn. Res.*, 5:361–397, Dec. 2004. ISSN 1532-4435.

T. C. Lin. Compliance, technology, and modern finance. *Brook. J. Corp. Fin. & Com. L.*, 11:159, 2016.

Y. Lin, Y. C. Tan, and R. Frank. Open sesame: Getting inside BERT's linguistic knowledge. In *Proceedings of the 2019 ACL Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pages 241–253, Florence, Italy, Aug. 2019.

A. Lipani, M. Lupu, A. Hanbury, and A. Aizawa. Verboseness fission for bm25 document length normalization. In *Proceedings of the 2015 International Conference on The Theory of Information Retrieval*, ICTIR '15, page 385–388, New York, NY, USA, 2015. Association for Computing Machinery. ISBN 9781450338332.

Z. C. Lipton. The mythos of model interpretability. *Commun. ACM*, 61(10):36–43, Sept. 2018. ISSN 0001-0782.

J. Liu, W.-C. Chang, Y. Wu, and Y. Yang. Deep Learning for Extreme Multi-label Text Classification. In *Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '17, pages 115–124, New York, NY, USA, 2017. ISBN 978-1-4503-5022-8.

N. F. Liu, M. Gardner, Y. Belinkov, M. E. Peters, and N. A. Smith. Linguistic Knowledge and Transferability of Contextual Representation. 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 1073–1094, Minneapolis, Minnesota, June 2019a.

Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer, and V. Stoyanov. Roberta: A robustly optimized BERT pretraining approach. *CoRR*, abs/1907.11692, 2019b.

C. Louizos and M. Welling. Multiplicative normalizing flows for variational bayesian neural networks, 2017.

B. Luo, Y. Feng, J. Xu, X. Zhang, and D. Zhao. Learning to predict charges for criminal cases with legal basis. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 2727–2736, 2017.

X. Ma and E. Hovy. End-to-end sequence labeling via bi-directional LSTM-CNNs-CRF. In *Proceedings of the 54th Annual Meeting of the Association of Computational Linguistics (ACL)*, pages 1064—-1074, Berlin, Germany, 2016.

S. MacAvaney, A. Yates, A. Cohan, and N. Goharian. Cedr: Contextualized embeddings for document ranking. In *Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR'19, page 1101–1104, New York, NY, USA, 2019. ISBN 9781450361729.

N. Manginas, I. Chalkidis, and P. Malakasiotis. Layer-wise guided training for BERT: Learning incrementally refined document representations. In *Proceedings of the Fourth Workshop on Structured Prediction for NLP*, pages 53–61, Online, 2020.

C. D. Manning, P. Raghavan, and H. Schütze. *Introduction to Information Retrieval*. Cambridge University Press, 2009.

J. McAuley and J. Leskovec. Hidden Factors and Hidden Topics: Understanding Rating Dimensions with Review Text. In *Proceedings of the 7th ACM Conference on Recommender Systems*, RecSys '13, pages 165–172, New York, NY, USA, 2013. ISBN 978-1-4503-2409-0.

R. McDonald, G. Brokos, and I. Androutsopoulos. Deep relevance ranking using enhanced document-query interactions. *CoRR*, abs/1809.01682, 2018.

M. Medvedeva, M. Vols, and M. Wieling. Judicial decisions of the European Court of Human Rights: Looking into the crystal ball. In *Proceedings of the Conference on Empirical Legal Studies*, 2018.

E. L. Mencia. Segmentation of legal documents. In *Proceedings of the 12th International Conference on Artificial Intelligence and Law*, pages 88–97, Barcelona, Spain, 2009.

E. L. Mencia and J. Fürnkranzand. An Evaluation of Efficient Multilabel Classification Algorithms for Large-Scale Problems in the Legal Domain. In *Proceedings of the 1st Linguistic Annotation Workshop*, pages 126–132, Halle, Germany, 2007.

T. Mikolov, M. Karafiát, L. Burget, J. Černocký, and S. Khudanpur. Recurrent neural network based language model. In *INTERSPEECH*, 2010.

T. Mikolov, K. Chen, G. Corrado, and J. Dean. Efficient Estimation of Word Representations in Vector Space. In *Proceedings of the International Conference on Learning Representations (ICLR)*, Scottsdale, AZ, 2013a.

T. Mikolov, I. Sutskever, K. Chen, G. Corrado, and J. Dean. Distributed Representations of Words and Phrases and their Compositionality. In *Proceedings of the 26th International Conference on Neural Information Processing Systems*, Stateline, NV, 2013b.

T. Mikolov, W. tau Yih, and G. Zweig. Linguistic regularities in continuous space word representations. In *Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT)*, pages 746–751, Atlanta, GA, 2013c.

A. Morimoto, D. Kubo, M. Sato, and H. Shindo. Legal Question Answering System using Neural Attention, 2017.

J. Mullenbach, S. Wiegreffe, J. Duke, J. Sun, and J. Eisenstein. Explainable Prediction of Medical Codes from Clinical Text. In *Proceedings of the Annual Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT)*, pages 1101–1111, New Orleans, Louisiana, 2018.

W. J. Murdoch, P. J. Liu, and B. Yu. Beyond word importance: Contextual decomposition to extract interactions from lstms. In *Proceedings of International Conference on Learning Representations (ICLR)*, 2018.

D. Nadeau and S. Sekine. A survey of named entity recognition and classification. *Lingvisticae Investigationes*, 30(1):3–26, 2007.

R. Nallapati and C. D. Manning. Legal docket classification: Where machine learning stumbles. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (Conference on Empirical Methods in Natural Language Processing (EMNLP))*, pages 438–446, 2008.

J. Naughton. Why a computer could help you get a fair trial. *Guardian*, 2017. URL `https://www.theguardian.com/technology/commentisfree/2017/aug/13/why-a-computer-could-help-you-get-a-fair-trial`.

J. O'Neill, P. Buitelaar, C. Robin, and L. O. Brien. Classifying Sentential Modality in Legal Language: A Use Case in Financial Regulations, Acts and Directives. In *Proceedings of the 16th International Conference on Artificial Intelligence and Law (ICAIL)*, pages 159–168, London, UK, 2017.

I. Partalas, A. Kosmopoulos, N. Baskiotis, T. Artières, G. Paliouras, É. Gaussier, I. Androutsopoulos, M.-R. Amini, and P. Gallinari. LSHTC: A Benchmark for Large-Scale Text Classification. *CoRR*, abs/1503.08581, 2015.

J. Pennington, R. Socher, and C. D. Manning. GloVe: Global Vectors for Word Representation. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543, Doha, Qatar, 2014.

M. E. Peters, M. Neumann, M. Iyyer, M. Gardner, C. Clark, K. Lee, and L. Zettlemoyer. Deep contextualized word representations. In *Proceedings of the Annual Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT)*, New Orleans, Louisiana, USA, 2018.

Y. Prabhu and M. Varma. FastXML: A Fast, Accurate and Stable Tree-classifier for Extreme Multi-label Learning. In *Proceedings of the 20th ACM SIGKDD International*

Conference on Knowledge Discovery and Data Mining, KDD '14, pages 263–272, New York, NY, USA, 2014.

Y. Prabhu, A. Kag, S. Harsola, R. Agrawal, and M. Varma. Parabel: Partitioned Label Trees for Extreme Classification with Application to Dynamic Search Advertising. In *Proceedings of the 2018 World Wide Web Conference*, WWW '18, pages 993–1002, Republic and Canton of Geneva, Switzerland, 2018.

J. Rabelo, M.-Y. Kim, R. Goebel, M. Yoshioka, Y. Kano, and K. Satoh. A summary of the coliee 2019 competition, 2019.

C. Raffel, N. Shazeer, A. Roberts, K. Lee, S. Narang, M. Matena, Y. Zhou, W. Li, and P. J. Liu. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research*, 21(140):1–67, 2020.

P. Rajpurkar, J. Zhang, K. Lopyrev, and P. Liang. Squad: 100,000+ questions for machine comprehension of text. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2016.

N. Reimers and I. Gurevych. Sentence-BERT: Sentence embeddings using Siamese BERT-networks. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 3982–3992, Hong Kong, China, Nov. 2019.

M. T. Ribeiro, S. Singh, and C. Guestrin. "why should I trust you?": Explaining the predictions of any classifier. *CoRR*, abs/1602.04938, 2016.

A. Rios and R. Kavuluru. Few-Shot and Zero-Shot Multi-Label Learning for Structured Label Spaces. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 3132–3142, Brussels, Belgium, 2018.

S. Robertson, S. Walker, S. Jones, M. M. Hancock-Beaulieu, and M. Gatford. Okapi at trec3. *In Overview of the Third Text Retrieval Conference*, pages 109—-126, 1995.

A. Rogers, O. Kovaleva, and A. Rumshisky. A primer in bertology: What we know about how bert works. *arXiv preprint arXiv:2002.12327*, 2020.

S. Sadiq and G. Governatori. *Managing Regulatory Compliance in Business Processes*, pages 265–288. Springer Berlin Heidelberg, Berlin, Heidelberg, 2015. ISBN 978-3-642-45103-4.

S. Serrano and N. A. Smith. Is attention interpretable? In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 2931–2951, Florence, Italy, 2019.

D. Stevenson and N. J. Wagoner. Bargaining in the shadow of big data. *Florida Law Review*, 67:1337, 2015.

E. Strubell, P. Verga, D. Belanger, and A. McCallum. Fast and Accurate Entity Recognition with Iterated Dilated Convolutions. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, Copenhagen, Denmark, 2017.

O.-M. Şulea, M. Zampieri, M. Vela, and J. van Genabith. Predicting the law area and decisions of French Supreme Court cases. In *Proceedings of the Recent Advances in Natural Language Processing (RANLP)*, pages 716–722, 2017.

M. Taylor, H. Zaragoza, N. Craswell, S. Robertson, and C. Burges. Optimisation methods for ranking functions with multiple parameters. In *Proceedings of the 15th ACM International Conference on Information and Knowledge Management*, CIKM '06, page 585–593, New York, NY, USA, 2006. Association for Computing Machinery. ISBN 1595934332.

P. M. Tiersma. *Legal language*. University of Chicago Press, 1999.

A. Trotman, A. Puurula, and B. Burgess. Improvements to bm25 and language models examined. In *Proceedings of the 2014 Australasian Document Computing Symposium*, ADCS '14, page 58–65, New York, NY, USA, 2014. Association for Computing Machinery. ISBN 9781450330008.

G. Tsatsaronis, G. Balikas, P. Malakasiotis, I. Partalas, M. Zschunke, M. R. Alvers, D. Weissenborn, A. Krithara, S. Petridis, D. Polychronopoulos, Y. Almirantis, J. Pavlopoulos, N. Baskiotis, P. Gallinari, T. Artières, A.-C. N. Ngomo, N. Heino, É. Gaussier, L. Barrio-Alvers, M. Schroeder, I. Androutsopoulos, and G. Paliouras. An overview of the BIOASQ large-scale biomedical semantic indexing and question answering competition. *BMC Bioinformatics*, 16(138), 2015.

V. N. Vapnik. *The Nature of Statistical Learning Theory*. Springer-Verlag, 1995.

A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin. Attention Is All You Need. In *Proceedings of the International Conference on Neural Information Processing Systems (NeurIPS)*, Long Beach, USA, 2017.

E. M. Voorhees. The TREC Robust Retrieval Track. *SIGIR Forum*, 39(1):11–20, June 2005. ISSN 0163-5840.

S. T. N. Vu Tran and M. L. Nguyen. Jnlp group: Legal information retrieval with summary and logical structure analysis. In *COLIEE Workshop on Juris-informatics (JURISIN)*, 2018.

B. Waltl, J. Muhr, I. Glaser, G. Bonczek, E. Scepankova, and F. Matthes. Classifying legal norms with active machine learning. In *Proceedings of the 30th International Conference on Legal Knowledge and Information Systems (JURIX)*, pages 11–20, Luxembourg, 2017.

A. Wang, A. Singh, J. Michael, F. Hill, O. Levy, and S. Bowman. GLUE: A multi-task benchmark and analysis platform for natural language understanding. In *Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pages 353–355, Brussels, Belgium, Nov. 2018.

A. Wang, Y. Pruksachatkun, N. Nangia, A. Singh, J. Michael, F. Hill, O. Levy, and S. R. Bowman. SuperGLUE: A stickier benchmark for general-purpose language understanding systems. *arXiv preprint 1905.00537*, 2019.

W. Y. Wang, E. Mayfield, S. Naidu, and J. Dittmar. Historical analysis of legal opinions with a sparse mixed-effects latent variable model. In *ACL*, pages 740–749, 2012.

J. H. Wenpeng Yin and D. Roth. Benchmarking zero-shot text classification: Datasets, evaluation and entailment approach. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing*, 2019.

S. Wiegreffe and Y. Pinter. Attention is not not explanation. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, Hong Kong, China, 2019.

C. Williams. *Tradition and change in legal English: Verbal constructions in prescriptive texts*, volume 20. Peter Lang, 2007.

R. J. Williams. Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Machine Learning*, 8(3–4):229–256, May 1992. ISSN 0885-6125. doi: 10.1007/BF00992696. URL `https://doi.org/10.1007/BF00992696`.

K. Xu, J. Ba, R. Kiros, K. Cho, A. Courville, R. Salakhudinov, R. Zemel, and Y. Bengio. Show, Attend and Tell: Neural Image Caption Generation with Visual Attention. In *Proceedings of the 32nd International Conference on Machine Learning (ICML)*, volume 37 of *Proceedings of Machine Learning Research*, pages 2048–2057, Lille, France, 07–09 Jul 2015. PMLR.

Z. Yang, D. Yang, C. Dyer, X. He, A. Smola, and E. Hovy. Hierarchical attention networks for document classification. In *Proceedings of the Annual Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT)*, pages 1480–1489, San Diego, CA, USA, 2016.

R. You, S. Dai, Z. Zhang, H. Mamitsuka, and S. Zhu. AttentionXML: Extreme Multi-Label Text Classification with Multi-Label Attention Based Recurrent Neural Networks. *CoRR*, abs/1811.01727, 2018.

R. You, Z. Zhang, Z. Wang, S. Dai, H. Mamitsuka, and S. Zhu. AttentionXML: Label Tree-based Attention-Aware Deep Model for High-Performance Extreme Multi-Label Text Classification. In *Advances in Neural Information Processing Systems 32*, pages 5812–5822. Curran Associates, Inc., 2019.

M. Yu, S. Chang, Y. Zhang, and T. Jaakkola. Rethinking cooperative rationalization: Introspective extraction and complement control. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 4094–4103, Hong Kong, China, Nov. 2019.

O. Zaidan, J. Eisner, and C. Piatko. Using "annotator rationales" to improve machine learning for text categorization. In *Human Language Technologies 2007: The Conference of the North American Chapter of the Association for Computational Linguistics; Proceedings of the Main Conference*, pages 260–267, Rochester, New York, Apr. 2007.

Y. Zhang, I. Marshall, and B. C. Wallace. Rationale-augmented convolutional neural networks for text classification. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 795–804, Austin, Texas, 2016.

H. Zhong, G. Zhipeng, C. Tu, C. Xiao, Z. Liu, and M. Sun. Legal judgment prediction via topological learning. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 3540–3549, 2018.

H. Zhong, C. Xiao, C. Tu, T. Zhang, Z. Liu, and M. Sun. How does NLP benefit legal system: A summary of legal artificial intelligence. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 5218–5230, Online, July 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.acl-mai n.466. URL `https://www.aclweb.org/anthology/2020.acl-main.466`.

A. Zubiaga. Enhancing Navigation on Wikipedia with Social Tags. *CoRR*, abs/1202.5469, 2012.