WestSearch Plus: A Non-factoid Question-Answering System for the Legal Domain

Gayle McElvain*
gayle.mcevlain@capitalone.com
Capital One
McLean, Virginia, USA

George Sanchez, Sean Matthews, Don Teo, Filippo Pompili, Tonya Custis
{first}.{last}@tr.com
Thomson Reuters
St. Paul, Minnesota, USA & Toronto, ON, Canada

ABSTRACT

We present a non-factoid QA system that provides legally accurate, jurisdictionally relevant, and conversationally responsive answers to user-entered questions in the legal domain. This commercially available system is entirely based on NLP and IR, and does not rely on a structured knowledge base. WestSearch Plus aims to provide concise one sentence answers for basic questions about the law. It is not restricted in scope to particular topics or jurisdictions. The corpus of potential answers contains approximately 22M documents classified to over 120K legal topics.

CCS CONCEPTS

• **Information systems → Question answering**; **Expert search**; *Query log analysis*; *Query reformulation*; • **Computing methodologies → Natural language processing**; **Artificial intelligence**; *Discourse, dialogue and pragmatics.*

KEYWORDS

question answering; legal question answering

ACM Reference Format:
Gayle McElvain and George Sanchez, Sean Matthews, Don Teo, Filippo Pompili, Tonya Custis. 2019. WestSearch Plus: A Non-factoid Question-Answering System for the Legal Domain. In *Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '19), July 21–25, 2019, Paris, France.* ACM, New York, NY, USA, 4 pages. https://doi.org/10.1145/3331184.3331397

1 INTRODUCTION

Question Answering (QA) in the Legal domain requires a system that is precise and legally accurate, while also providing adequate recall across different jurisdictions (governing law may vary between jurisdictions). Such a system needs to be robust to the different information needs that arise across different legal specialties and practice areas, as well as to differences in language use between individual attorneys and the statutes of a particular jurisdiction.

Attorneys are continually trying to optimize their research time, trying to provide value for their clients in terms of doing the most and best research possible in the allowed number of billable hours. As such, it is important for attorneys to find the most relevant information for their clients' matters in the quickest manner possible. We present WestSearch Plus, a Question Answering system for legal questions that allows attorneys to quickly find the most salient points of law, related case law, and statutory law appropriate to their jurisdiction.

Our system aims to provide conversationally fluent, concise one sentence answers for basic questions about the law. It is not restricted in scope to particular topics or jurisdictions. Consider also that scoring based on linguistic analysis is computationally expensive relative to the scoring methods used for search and that the entire system must be performant enough to run dynamically from a global search box. Taken together, these factors make search a critical component in the system, functioning as the primary means by which to narrow the universe of potential answers for scoring. Additionally, our system relies on NLP models targeted at the tokens, syntax, semantics, and discourse structure of legal language, in addition to other machine-learned models to classify question and answer intents, identify named entities and legal concepts, and for classifying, ranking, and thresholding the final answer candidates.

The answers provided are taken from a corpus of approximately 22 million editorially-written, one-sentence summaries of US case law classified to over 120K legal topics. These one-sentence summaries of case law (Headnotes) are editorially added to cases as part of the content publishing process. Attorney-editors have been adding Headnotes to cases in this manner to Thomson Reuters content for over 100 years.

2 TRAINING DATA & ANNOTATION TASK

As is typical, our QA system was trained on a large corpus of question-answer pairs. In total, we trained the production system on approximately 200K QA pairs, with an average of 3 correct vs. incorrect judgements for each supplied by attorney-editors.

Our initial set of questions was mined from Westlaw (a legal search engine) query logs. Initial QA pairs were constructed by attorney experts in their attempts to find answers to those questions by using Westlaw. Our answer corpus consists of about 22 million human-written, one-sentence summaries of US court case documents, spanning over 100 years of case law. As such, no passage retrieval, algorithmic summarization, nor NLG derived from the longer case documents is necessary to render the answers.

*Work was done while employed at Thomson Reuters.

SIGIR '19, July 21–25, 2019, Paris, France

ACM ISBN 978-1-4503-6172-9/19/07…$15.00
https://doi.org/10.1145/3331184.3331397

QA Pairs were given four labels (**A**, **C**, **D**, or **F**) by attorney-editors. Both **A** and **C** labels are factually correct answers, but **A** answers are ideal and more pragmatically correct (sound like a natural answer to the question and give all and only the necessary amount of information). **D** and **F** answers are incorrect, but with **D**s being less egregiously wrong than **F**s.

3 LOGICAL SYSTEM ARCHITECTURE

Logically, the QA system has four main components: Question Analysis, Query Generation & Federated Search, Answer Analysis, and Question-Answer Pair Scoring.

3.1 Question Analysis

The first step in processing a question is to infer its basic linguistic structure. This involves machine learning algorithms trained to predict parts of speech, NP and VP chunks, syntactic dependency relations, and semantic roles. The QA system employs open source models,[1] [2] trained on annotated sentences from varied domains [3].

To detect named entities and legal concepts in both questions and answers, we use a combination of gazetteer lookup taggers and statistical taggers trained with Conditional Random Fields.[3]

We classify a question's semantic intent to a set of predetermined semantic frames. A semantic frame is a coherent structure of related concepts, where the relationships between concepts tend to be realized according to prototypical patterns. This notion of frame is borrowed from Frame Semantics [4] and related notions in theories of Construction Grammar [5]. The process for identifying semantic frames was informed by editorial guidelines used to author the one-sentence summaries comprising the answer corpus. The frame of a question and its ideal answer should always be one in the same.

At runtime, questions are classified to a particular frame (or as "out-of-frame"). A Neural Network question classifier is trained on labeled (question, frame) pairs for each frame [1].

3.2 Query Generation & Federated Search

Search is a critical component in our QA system, functioning as the primary means by which to narrow the universe of potential answers for scoring. Linguistic analysis and feature generation for all questions against 22M potential answers is computationally expensive relative to the scoring methods used for search.

While no search strategy can precisely identify all possible answers to a question, the likelihood of retrieving a correct answer is increased by running multiple searches against different search engines. We execute three types of queries for each question against different search indices in two different search engines: 1) Natural language searches, derived from the question text; 2) Structured semantic searches, derived from the text, entities, and semantic frame information of the question; 3) More-like-this relevance feedback searches, derived from highly-ranked candidate answers.

A default natural language search strategy is applied to all incoming questions. This type of search is run against answer indices created in both a proprietary search engine and Elasticsearch. We chose these two search engines because they provide very different results. The proprietary engine has been fine tuned over several years with many features that enhance legal search in particular. Elasticsearch is open source, and does not have any such fine-tuning. Together, these two search engines give us an overall better candidate pool of answers in terms of recall.

The question answering system leverages semantic search strategies for questions belonging to known frames. This is one way the system uses frame classification on both questions and answers. The questions must be classified at runtime, but candidate answers can be classified offline and stored in a separate index. This enables search to target the particular subset of answers evoking the same semantic frame as the question. Depending on the frame of the question, multiple queries may be generated in order to target specific frame elements. This is accomplished with frame-specific template queries that have placeholders for specific frame elements. Recognized entities and legal concepts in the question replace placeholders in frame-specific template queries to produce fully formed queries for execution against a search engine.

Finally, More-like-this search is a relevance feedback strategy used to widen the pool of potential answers after an initial set of candidates have been scored. The main contribution of this strategy is to expand coverage for specific legal jurisdictions. It involves searching for answers that closely match high scoring answers from outside the user's jurisdiction.

Document vectors constructed over the answer corpus are used to measure the semantic similarity between top-ranked answer candidates and more-like-this answer candidates. The approach used to generate document embeddings is based on the Paragraph Vectors model, also known as doc2vec, proposed by [7]. Models were trained on the answer corpus using an implementation provided by [6].

3.3 Answer Analysis

Search produces a pool of candidate answers for deeper analysis. The answer analysis stage mirrors the linguistic analyses produced for questions as detailed above. Unlike with questions, however, all answer analyses can be precomputed and stored to optimize performance.

Although the basic procedure and outcome of applying entity recognition and semantic frame classifications to answers is the same as for questions, different models for both must be trained on the answers due to their more complex syntactic structure. Because answers are harder to read than questions, the creation of adequate training data for the answer frame classifiers is more time consuming: individual (answer, frame) labels take much longer for attorney-editors to produce, and due to greater linguistic variation between answers than questions, more labels are also required to train a reliable model.

To reduce the amount of training data required, a two-step procedure leveraging search-based classification was used. In the first stage, queries were written by attorney-editors to identify a high-recall subset of the corpus containing answers associated with each frame. Then one maximum entropy classifier per frame is trained on the corresponding search results. Manual judgments indicating whether or not each document matching the query is actually an

[1] https://opennlp.apache.org/
[2] https://emorynlp.github.io/nlp4j/
[3] http://www.chokkan.org/software/crfsuite

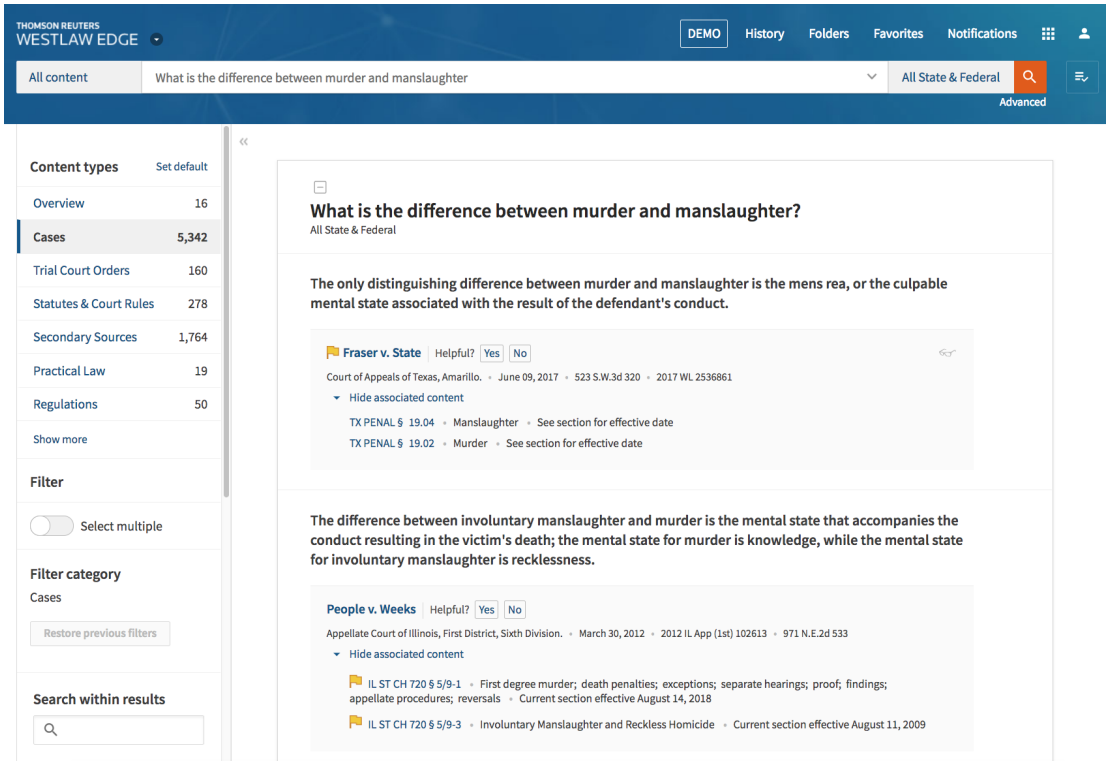

Figure 1: QA System screenshot, displaying two candidate answers in the 'teaser' view on top of WestSearch results.

instance of the frame are then used as training data for the classifiers. This requires less training data overall, because each subset of the data has less variation, and the task domain is narrower.

3.4 Question-Answer Pair Scoring

Feature scoring functions for the syntactic analyses of a question-answer pair primarily measure overlap and alignment between the question and the answer sentence. Different features compute the alignment between noun phrases, dependency relations, and verb phrases. Various word embedding models trained on both open domain corpora and legal corpora are employed to measure semantic similarity within these structures.

Positional features capture the intuition that concepts in the question are likely to occur more closely together in correct answers than in incorrect answers. Distance is measured over the syntactic parse tree as well as over token and character offsets.

Answers that read like a natural answer to the question will typically put concepts from the question in English "topic" position near the beginning of the sentence. Highly-rated answers have a strong tendency to exhibit this pattern. Correct answers will also often have question concepts near the root of the answer's syntactic parse tree. Both these tendencies are captured with the topicality features.

All questions and answers are classified to a legal taxonomy with over 120K fine-grained categories. All answers have manually assigned key numbers assigning the point of law to one or more categories. The classification scheme is quite complex, so user questions are generally underspecified relative to the taxonomy. As such, when there are multiple answer candidates with the correct answer to a question, we do not expect that they all belong to the same category. In the same search result, however, there is a tendency for correct answers to have fewer distinct category classifications among them than incorrect answers. This indicates some association between question intent and taxonomic classification, which is leveraged by the system.

In particular, both question and answer candidate are classified to the legal taxonomy and feature scoring functions compare the similarity of those outputs. In addition, the predicted classifications for a question are also compared to the manually assigned categories for each answer.

All of the above features are combined in an ensemble model of weak learners [2]. This supervised model learns by example from labeled question answer pairs. At runtime, each QA pair is considered independently by the model and ranked by a score that represents the probability of that candidate being a correct answer for that question.

The last stage of the system determines whether or not to show an answer based on its probability score. Determining probability score thresholds is a business decision that weighs the relative cost of showing some incorrect answers against the cost of showing customers fewer answers (i.e., answering fewer customer questions).

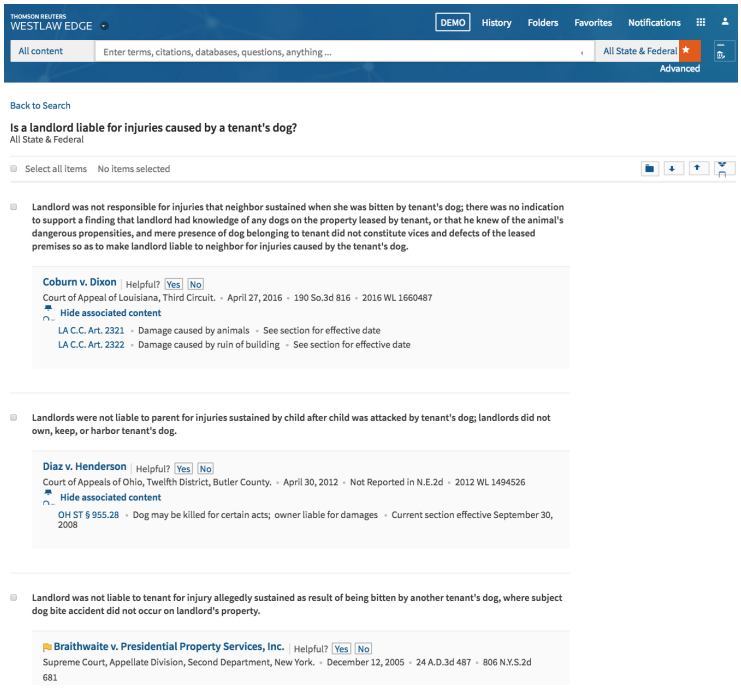

Figure 2: QA System screenshot, displaying the 'Show more' answers page

Thresholds were set using 10-fold cross validation on all graded data. Thresholds were chosen by the business to: 1) maximize *Answered at*[4] metrics for correct answers (90% *Answered at* 3), 2) minimize *Answered at* metrics for **F** answers (1.5% *Answered at* 3), while 3) also balancing the system's coverage (the number of user questions for which answers are shown).

4 DEMO

Our QA system for the legal domain takes a question as input in the global search box. If high-confidence answers are returned by the QA system for the user's question in the user's jurisdiction, they are shown above the search results returned by Westlaw's main search algorithm for that question run as a natural language query.

The user's question and the answers are clearly displayed. For additional context (as a one-sentence summary is often not enough for an attorney to base a case or argument on), options are given to the user to click through to exactly the part of the case from which the answer comes and to any relevant statutes (laws) for additional context. In addition, the user can click on 'Show more' to see more answers (and related content) to their question.

The ability to quickly find answers and to link directly to relevant, more in-depth content from those answers provides attorneys a quick and comprehensive entry point into doing their research that other legal research platforms lack.

Since its launch in July 2018, 40% of Westlaw users have triggered the WestSearch Plus feature. When answers are presented to the user, there is a 52% clickthrough rate to see the full case, statute, or more answers related to their question.

[4]Answered at is defined for each question's answer set such that it is the percentage of questions for which there is at least one of a particular label returned by the system at or above the rank indicated (so, at rank 1, *Answered at* 1 for **A**s is equivalent to Precision at 1).

5 CONCLUSION

We have presented a commercially-released non-factoid QA system that relies extensively on IR and NLP. QA in the Legal domain requires a system that is precise, while also providing adequate recall across different jurisdictions. Our system is robust to the different information needs and different language usage that arise across different legal practice areas and jurisdictions.

REFERENCES

[1] Piotr Bojanowski Armand Joulin, Edouard Grave and Tomas Mikolov. 2017. Bag of tricks for efficient text classification. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics*. EACL, 427–431.

[2] Tianqi Chen and Carlos Guestrin. 2016. Xgboost: A scalable tree boosting system. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. KDD, 785–794.

[3] Jinho D. Choi. 2016. Dynamic Feature Induction: The Last Gist to the State-of-the-Art. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics*. ACL, 271–281.

[4] Charles J. Fillmore. 2006. Frame semantics. Cognitive linguistics: Basic readings. In *Cognitive linguistics: Basic readings*, Dirk Geeraerts (Ed.). Walter de Gruyter, Berin, 34, 373–400.

[5] Adele E. Goldberg. 2006. In *Constructions at work: The nature of generalization in language*. Oxford University Pressr.

[6] Jey Han Lau and Timothy Baldwin. 2016. An Empirical Evaluation of doc2vec with Practical Insights into Document Embedding Generation. In *Proceedings of the 1st Workshop on Representation Learning for NLP*. 78–86.

[7] Quoc Le and Tomas Mikolov. 2014. Distributed representations of sentences and documents. In *Proceedings of the 31st International Conference on Machine Learning*. ICML, 1188–1196.