Easing Legal News Monitoring with Learning to Rank and BERT

Luis Sanchez[1,2], Jiyin He[2], Jarana Manotumruksa[1], Dyaa Albakour[2], Miguel Martinez[2], and Aldo Lipani[1]

[1] University College of London, London,
{luis.izquierdo18,j.manotumruksa,aldo.lipani}@ucl.ac.uk
[2] Signal AI, London, research@signal-ai.com

Abstract. While ranking approaches have made rapid advances in the Web search, systems that cater to the complex information needs in professional search tasks are not widely developed, common issues and solutions typically rely on dedicated search strategies backed by ad-hoc retrieval models. In this paper we present a legal search problem where professionals monitor news articles with constant queries on a periodic basis. Firstly, we demonstrate the effectiveness of using traditional retrieval models against the Boolean search of documents in chronological order. In an attempt to capture the complex information needs of users, a learning to rank approach is adopted with user specified relevance criteria as features. This approach, however, only achieves mediocre results compared to the traditional models. However, we find that by fine-tuning a contextualised language model (e.g. BERT), significantly improved retrieval performance can be achieved, providing a flexible solution to satisfying complex information needs without explicit feature engineering.

Keywords: professional search · complex information needs · BERT.

1 Introduction

In information retrieval (IR), there has been a long standing interest in professional search, as demonstrated by various TREC tracks dedicated to a diverse range of professional domains [4, 8]. Unlike traditional Web search, an important characteristic of professional search is the complex information needs of the professional users. For instance, a professional user may ask for information within certain time range, written in a professional quality [24].

Although there have been ongoing discussions and studies calling for search systems addressing common issues faced by professional search, solutions typically rely on dedicated databases or specialised search strategies that are backed by ad-hoc retrieval models [23, 20]. Meanwhile, although traditional retrieval models as well as learning to rank (L2R) approaches have made rapid advances in Web search, retrieval models that cater to the diverse requirements in professional search tasks are not widely developed.

In this paper, we study a case in the context of *legal professional search* and investigate how different retrieval approaches can be employed to address the

complex needs of professional users. The work task of our users is to monitor a number of legal topic of their interest in news and select articles to be included in a report periodically according to a set of clearly defined criteria, ranging from topical relevance to language quality. Like in many other professional search senarios, our users setup their searches against a news stream with complex Boolean queries [20] where results are ranked in chronological order; and they deem recall an important metric as they do not want to miss relevant articles.

While Boolean queries are often preferred by professional searchers due to their needs of having results which are "efficient, trustable, explainable and accountable"[16, 24], it falls short in addressing the complex relevance criteria beyond matching terms. Traditional retrieval models such as BM25 and Language Models (LM) capture topical relevance. As a first step going beyond the Boolean search practice, we answer the following research question:

RQ1. Do traditional IR models help our users in identifying relevant documents more effectively compared to the Boolean search practice?

Further, in order to satisfy users' complex information needs beyond topicality, it seems natural to encode indicators of different criteria as features, and combine them with a L2R approach. Therefore, our next research question is:

RQ2. Can we provide better results by adopting a L2R approach to satisfy users' complex information needs beyond topicality?

However, feature engineering for every criterion can be time-consuming and may not be convenient when switching from one use case to another. Recently, pre-trained contextualised language models (e.g. BERT [10]) have effectively addressed various NLP tasks [10, 25] eliminating the need of feature engineering. This leads the investigation of the follow up research question:

RQ3. Can we further improve the quality of the search results by fine-tuning a pre-trained language model on our search task?

Our contributions can be summarised as follows:

1. Unlike simulation based studies such as TREC tasks where the information needs, relevance criteria and judgements are set by different parties rather than the actual users, the complex information needs in our study come from real users, who also define the relevance criteria and provide judgements. Our study not only reveals the practical challenges for professional search systems, but also demonstrates possible solutions to effectively address these challenges, and;
2. We also contribute to the generic solution to professional search (i.e. search with complex information needs). Our experiments show the potential of employing pre-trained contextualised language models to learn relevance criteria without handcrafted features, which leads to a flexible solution that adapts to varying complex needs.

2 Methodology

In this case study, our users have three relevance criteria: *topical relevance*, *factual information*, and *language quality*. Specifically, the topicality of retrieved articles must be associated with a specific legal area[3]; only factual articles are considered relevant; and articles written in technical language and linguistically accurate are preferred.

We first explore the effectiveness of traditional models in satisfying the users' needs. We include four models: TF-IDF, BM25, unigram Language Model (LM) with Jelinek-Mercer and Dirichlet smoothing, applied to three fields of the news articles (title, summary, and content). As for query, we extract the keywords from the complex Boolean queries that our users created and concatenate them as a long query (typically ∼100 words), where negation terms were ignored.

In order to estimate the relevance of a document with respect to the combined relevance criteria described above, we employ a L2R approach and encode these criteria as features. We devise 28 features (see Table 1) as follows.

Topical relevance. We model topical relevance using the outputs of traditional retrieval models, as usually done in the literature [17].

Factual information. We model factual information with three types of features: (1) Subjectivity: it measures the degree of subjectivity of an article, which is directly related to the "factuality" of the article. (2) Modality: it shows the degree of certainty of the statements of an article by looking at the verb tense in which the article is written. (3) Sentiment: it provides the degree of negativity or positivity of the language used–while not directly related to the factuality dimension, there can be entanglement between the subjective and opinionated dimensions [13]. We employ a lexicon based approach to compute these features [22, 15]. We also include the number of lexicons assessed in an article as a normalisation factor for articles of different lexicon sizes.

Language quality. Since the content our users request are technical and sometimes hard for non-expert to read, we employ readability scores as features, which measure the ease with which a reader can understand a written text.

Apart from devising task specific features, we exploit a pre-trained contextualised language model to automatically learn the complex relevance criteria. By fine tuning the model on our search task we expect to associate these language features with the relevance judgements. We use BERT [18], which shows the state-of-the-art performance on a wide range of NLP tasks [10, 25]. Inspired by the work of MacAvaney et al. [18], we employ BERT in its regression form (known as Vanilla BERT in [18]). Specifically, the input consists of a query-document pair, and the output is a predicted relevance score. For document input, we use (i) a combined title and summary field (referred to as BERT on summary), and (ii) the content of the article (referred to as BERT on content).

[3] Information regarding the specific legal domain cannot be disclosed due to a non-disclosure agreements that we have with the legal professionals.

Type	Features	Description
Topicality	Retrieval model scores (12)	TF-IDF, BM25, LM (J-M), and LM (Dir) applied on an article's title, define summary, and content.
Factual information	Subjectivity (1)	Degree of objectivity vs. subjectivity
	Modality (1)	Degree of certainty of the statements.
	Sentiments (4)	Negative, positive, neutral, and compound scores [15].
	Lexicons (1)	# of the lexicon's vocabulary that appear in the article.
Language quality	Readability (9)	Kincaid index[12], Readability Index [21], Coleman-Liau index [7], Flesch index [11], Gunning Fog index [14], Lasbarhets index [6], McLaughlins SMOG index [19], John Aderson's index [5], and Dale-Chall index [9].

Table 1: Features for Learning to Rank

3 Experimental Setup

Dataset. The dataset we use to evaluate our retrieval approaches comes from the interaction data of legal professionals with a news monitoring system over a one year period. The users monitor a specific legal topic by querying the news stream with that topic periodically and all the retrieved results are tagged with a relevance judgment for later usage. Given this context, we group the data into equal intervals corresponding to the report creation times and evaluate the retrieved results per interval. The initial ranked lists were generated by using the Boolean query created by the users, and ranked in a chronological order. We apply the alternative ranking approaches as a re-ranking task. In total the dataset consists 206 queries and 60,512 labelled news articles, among which 2,872 (21%) are marked as relevant. By grouping the searches into the equal interval and removing sessions with no relevant articles, we obtain 1,774 search sessions (i.e. query-results pairs). The average number of relevant articles per-session varies from 1.5 to 4.4 articles depending on the queries. We randomly split the dataset into training (80%), validation (10%), and test (10%) sets. The same setup holds for both traditional retrieval models as well as for L2R approaches. We use the validation set to tune the models' parameters.

Evaluation measures. We use Mean Average Precision (MAP) to train and measure the performance of the retrieval models. Since recall is important in this user task, we use two recall oriented metrics: R@3 (given the small number of relevant documents per search); and average Search Length (SL) which measures the amount of effort a user needs to find *all* relevant documents.

Features. The topical features take the scores generated by the traditional IR models with their optimal parameter settings. For language usage features, we use an implementation of the CLiPS *pattern-en* module for subjectivity and modality [22]; and VADER [3] to compute sentiment scores. The 9 readability features are computed using the Python Readability package [2].

Models. As L2R approach we use the LambdaMART implementation from RankLib [1]. We apply a linear normalisation to our features as implemented by the library; each feature is normalised according to its minimum and maximum values. BERT is fine tuned using our labelled data as described by MacAvaney

	MAP			R@3			SL		
Boolean	0.421			0.469			6.66		
Method	T	S	C	T	S	C	T	S	C
TF-IDF	0.518	0.509	0.501	0.580	0.579	0.591	5.632	5.800	5.859
BM25	0.517	0.520	0.546	0.585	0.576	0.594	5.643	5.622	**5.303**
J-M	0.534	0.522	0.551	0.599	0.589	0.588	5.616	5.638	5.660
Dir	0.531	0.521	**0.556**	**0.615**	0.586	0.594	5.465	5.627	5.595

Table 2: Traditional models vs. Boolean search. Models are run on Title (T), Summary (S), and Content (C). All differences are statistically significant compared to the Boolean search results (paired t-test with p-value < 0.01).

et al. [18]. The input of BERT is the concatenation of a [CLS] token, the query, a [SEP] token, and the document. A document is capped when longer than 512 tokens. The output of BERT is the vector representations for each input token. We use the BERT-base uncased version and each vector has a dimension of 764. For fine-tuning we stack a linear-layer on top of BERT, which takes as input the output vector for the [CLS] token. We rank documents according to the score output by the linear-layer.

4 Results & Discussion

Regarding RQ1, Table 2 lists the results of traditional retrieval models (TF-IDF, BM25 and Language Model (LM) with Jelinek-Mercer smoothing (J-M) and Dirichlet smoothing (Dir)) compared to that from the Boolean search with a chronological order, i.e. the working practice of our professional users. We see that all retrieval models significantly outperform the Boolean search results for all measures. This suggests that without further effort in terms of feature engineering and model fitting, traditional models already improve the ranking quality by capturing topical relevance. Further, we see that different fields may be best for one model but not for the other, suggesting that their combination in a L2R approach may be beneficial. Hereafter, we choose Dir on content, which has the best MAP score, as a baseline for the remaining experiments.

To address RQ2 and RQ3, Table 3 shows the results of LambdaMART with explicitly encoded features and BERT scores, compared to the Dir baseline.

Firstly, We see LambdaMART with explicitly encoded features has no significant improvements over the baseline. In particular, topical features (i.e. a linear combination of the traditional models and different fields) does not provide better performance compared to Dir. However, among the different type of features, LambdaMART with all features performs the best, suggesting that both types of features are somewhat useful in capturing the relevance criteria. In response to RQ2, these results imply that the L2R approach with explicit feature engineering does not achieve a competitive performance—perhaps, hand-crafted features were not able to match well with the user specified relevance criteria.

Method	MAP	R@3	SL
Dir	0.556	0.596	5.595
LambdaMART (all features)	0.543	0.616	5.205
LambdaMART (topicality)	0.550	0.607	5.400
LambdaMART (language)	0.514	0.572	5.503
BERT on summary	0.739‡	**0.831**‡	**1.746**‡
BERT on content	**0.763**‡	0.777†	2.059‡

Table 3: Results of LambdaMART with feature variants. † indicates a statistical significant difference compared to the baseline Dir with p-value < 0.05 by paired t-test (p-value < 0.01 when ‡).

Next, we observe that BERT based approaches significantly outperform Dir. In particular, in terms of SL, with the baseline a user would need to read on average 5.5 irrelevant documents before finding all relevant documents, while with BERT based models this is reduced to less than 2, providing potentially improved user experience. Moreover, compared to explicit feature engineering, fine tuning BERT seems to have captured the user information needs in an implicit manner. This is encouraging as it not only learns the complex relevance criteria more accurately, but also provides more flexibility as the model can be fine tuned for use cases with different criteria without dedicated feature engineering.

The above results show promising performance of different ranking approaches in terms of off-line IR evaluation, compared to the original Boolean setup. From a user perspective, this means users may be able to confidently stop reading results after seeing certain number of irrelevant results, which would be particularly useful when the result list is long and relevant articles are few. On the other hand, we should also be aware that as the model complexity increases, there is decreasing model explainability and user controllability—the properties of Boolean search appreciated by professional users [16, 24]. Therefore for future work we find it crucial to investigate methods that explain and control complex models such as BERT.

5 Conclusion

We explored different retrieval approaches to address the complex information needs of professional users in a legal search context. We found that, compared to Boolean search, traditional retrieval models are effective in improving the ranking quality and reducing user effort in finding relevant information (e.g. measured by SL). Learning to rank with explicit feature encoding does not seem to be able to easily improve over traditional models. However, fine-tuning a pre-trained language model (BERT) shows strong improvements over both traditional models and L2R models, with the advantage of not requiring dedicated feature encoding. In particular, our study opens up a number of research questions in the context of professional search: (i) what kind of features allow pre-trained LMs to capture the implicit information needs from users' relevance judgements? (ii)

what are the limitations of pre-trained LMs to capture fine-grained information needs? and; (iii) how does the above depend on the number and quality of the relevance judgements, particularly in the case of niche retrieval tasks?

References

1. Ranklib. https://sourceforge.net/p/lemur/wiki/RankLib/
2. Readability. https://pypi.org/project/readability/
3. Varder. https://github.com/cjhutto/vaderSentiment
4. Guidelines for the 2011 trec medical records track (2011), https://www-nlpir.nist.gov/projects/trecmed/2011/, [Acessed on 2019-08-26]
5. Anderson, J.: Lix and rix: Variations on a little-known readability index. Journal of Reading **26**(6), 490–496 (1983)
6. Björnsson, C.H.: Läsbarhet. Liber (1968)
7. Coleman, M., Liau, T.L.: A computer readability formula designed for machine scoring. Journal of Applied Psychology **60**(2), 283 (1975)
8. Cormack, G.V., Grossman, M.R., Hedin, B., Oard, D.W.: Overview of the trec 2010 legal track. In: Proc. of TREC. vol. 1 (2010)
9. Dale, E., Chall, J.S.: A formula for predicting readability: Instructions. Educational research bulletin pp. 37–54 (1948)
10. Devlin, J., Chang, M.W., Lee, K., Toutanova, K.: Bert: Pre-training of deep bidirectional transformers for language understanding. In: Proc. of NAACL (2019)
11. Farr, J.N., Jenkins, J.J., Paterson, D.G.: Simplification of flesch reading ease formula. Journal of applied psychology **35**(5), 333 (1951)
12. Flesch, R.: A new readability yardstick. Journal of applied psychology **32**(3), 221 (1948)
13. Fuhr, N., Giachanou, A., Grefenstette, G., Gurevych, I., Hanselowski, A., Järvelin, K., Jones, R., Liu, Y., Mothe, J., Nejdl, W., et al.: An information nutritional label for online documents. In: SIGIR Forum. vol. 51, pp. 46–66 (2017)
14. Gunning, R.: The Technique of Clear Writing. McGraw-Hill (1952)
15. Hutto, C.J., Gilbert, E.: Vader: A parsimonious rule-based model for sentiment analysis of social media text. In: Proc. of AAAI (2014)
16. Kim, Y., Seo, J., Croft, W.B.: Automatic boolean query suggestion for professional search. In: Proc. of SIGIR. pp. 825–834. ACM (2011)
17. Liu, T.Y., et al.: Learning to rank for information retrieval. Foundations and Trends in Information Retrieval **3**(3), 225–331 (2009)
18. MacAvaney, S., Yates, A., Cohan, A., Goharian, N.: Cedr: Contextualized embeddings for document ranking. In: Proc. of SIGIR (2019)
19. Mc Laughlin, G.H.: Smog grading-a new readability formula. Journal of reading **12**(8), 639–646 (1969)
20. Russell-Rose, T., Chamberlain, J., Azzopardi, L.: Information retrieval in the workplace: A comparison of professional search practices. Information Processing & Management **54**(6), 1042 – 1057 (2018)
21. Senter, R., Smith, E.A.: Automated readability index. Tech. rep., Cincinnati Univ. Ohio (1967)
22. Smedt, T.D., Daelemans, W.: Pattern for python. Journal of Machine Learning Research **13**(Jun), 2063–2067 (2012)
23. Verberne, S., He, J., Kruschwitz, U., Wiggers, G., Larsen, B., Russell-Rose, T., de Vries, A.P.: First international workshop on professional search. In: Proc. of SIGIR. vol. 52, pp. 153–162. ACM (2019)

24. Verberne, S., He, J., Wiggers, G., Russell-Rose, T., Kruschwitz, U., de Vries, A.P.: Information search in a professional context-exploring a collection of professional search tasks. In: Proc. of SIGIR. pp. 1–5. Paris, France (2019)
25. Yang, Z., Dai, Z., Yang, Y., Carbonell, J., Salakhutdinov, R., V. Le, Q.: Xlnet: Generalized autoregressive pretraining for language understanding (06 2019)