

Rethinking Search: Making Experts out of Dilettantes

Donald Metzler
metzler@google.com

Google Research
Mountain View, California, USA

Dara Bahri
dbahri@google.com
Google Research

Mountain View, California, USA

Yi Tay
yitay@google.com

Google Research
Mountain View, California, USA

Marc Najork
najork@google.com
Google Research

Mountain View, California, USA

ABSTRACT

When experiencing an information need, users want to engage with an expert, but often turn to an information retrieval system, such as a search engine, instead. Classical information retrieval systems do not answer information needs directly, but instead provide references to (hopefully authoritative) answers. Successful question answering systems offer a limited corpus created on-demand by human experts, which is neither timely nor scalable. **Large pre-trained language models, by contrast, are capable of directly generating prose that may be responsive to an information need, but at present they are dilettantes rather than experts** – they do not have a true understanding of the world, they are prone to hallucinating, and crucially they are incapable of justifying their utterances by referring to supporting documents in the corpus they were trained over. This paper examines how ideas from classical information retrieval and large pre-trained language models can be synthesized and evolved into systems that truly deliver on the promise of expert advice.

CCS CONCEPTS

• **Information systems** → **Retrieval models and ranking.**

KEYWORDS

retrieval, ranking, language models, multi-task learning, few-shot learning

ACM Reference Format:

Donald Metzler, Yi Tay, Dara Bahri, and Marc Najork. . Rethinking Search: Making Experts out of Dilettantes. In *Proceedings of . ACM*, New York, NY, USA, 12 pages.

1 INTRODUCTION

Given an information need, users often turn to search engines for help. Such systems point them in the direction of one or more relevant items from a large corpus. This is appropriate for navigational and transactional intents (e.g. home page finding or online shopping) but less than ideal for informational needs, where users seek answers to questions they may have [8]. Classical information retrieval (IR) systems do not directly answer information needs, but instead provide references to (hopefully authoritative) content.

The very fact that ranking is a critical component of this paradigm is a symptom of the retrieval system providing users a selection of potential answers, which induces a rather significant cognitive burden on the user. The desire to return *answers* instead of ranked lists of results was one of the motivating factors for developing question answering systems. While there has been a great deal of research into QA systems, large-scale practical success has been somewhat limited. The original vision of question answering was to provide the aforementioned human expert response (i.e., ask a question using natural language and get an answer in natural language). Question answering systems have only delivered on the question part. Their responses, however, are either traditional lists of relevant documents, snippets extracted from documents, or answers created by human editors (e.g., Yahoo! Answers, Naver, Quora). While these solutions go beyond the experience afforded by classical IR systems, they suffer from a number of issues, including those related to coverage (e.g., answers are only provided for a small fraction of all possible questions) and authoritativeness (e.g., answers are often crowdsourced from both high and low quality sources).

When it comes to natural language understanding, there has been significant progress over the past decade that has largely been fueled by the deep learning movement. Early advances, which have had wide-ranging impact across many research disciplines, include word embeddings (which capture word similarity) [56, 65], advances in sequence modeling (which capture morphological and grammatical phenomena), and large pre-trained language models (LMs) [10, 54, 69] (which can capture information about the relationship between entities). Improvements in these technologies are driven by ever-expanding data sets and model sizes, which allows such models to encode more and more knowledge about the world and demonstrate impressive capabilities to generalize via zero- and few-shot learning.

Unlike traditional IR or question answering systems, state-of-the-art pre-trained LMs [10, 20, 70] are capable of directly generating prose that may be responsive to an information need. However, such models are *dilettantes* – they do not have a true understanding of the world, they are prone to hallucinating, and crucially they are incapable of justifying their utterances by referring to supporting documents in the corpus they were trained over. This paper argues that many of these limitations result from the fact that such models fail to bridge the gap between sequences of terms and documents

(and all the important meta-information associated with documents like provenance, authorship, authoritativeness, polarity, etc.).

Given the significant recent progress developing information retrieval, question answering, and pre-trained language modeling capabilities, now is an opportune time to take a step back to try to envision what possibilities the future might hold in terms of synthesizing and evolving these technologies into the next generation of IR systems that can help us get one step closer to truly human expert quality responses.

This paper envisions how experts can be created by leveraging pre-trained LMs. Of course, actual experts have a "true understanding" of a given topic. Building such experts would likely require developing an artificial general intelligence, which is beyond the scope of this paper. Instead, by "expert" we specifically mean that the system is capable of producing results that are of human expert quality (with or without actual "understanding"). To achieve this, this paper explores how ideas from classical IR and large pre-trained LMs can be synthesized and evolved into systems that deliver on the promise of expert response quality.

To move well beyond the current state-of-the-art, the fundamental assumptions that underlie modern IR systems need to be questioned. One of the key assumptions that this paper takes a critical look at is whether search indexes as we know them today are absolutely necessary or do they perhaps impose unnecessary and artificial restrictions on systems.

The inverted index has served as the workhorse of most modern search engines over the past several decades [16]. Such indexes encode term frequencies, term positions, document structure information, various forms of document metadata (e.g., document length), etc. They allow users to query the system using a mix of terms, phrases, and other so-called "advanced search operators" (e.g., "title:"). On the other hand, inverted indexes treat words as uninterpreted tokens, they do not capture their semantics. Specifically, the index is oblivious of morphology (instead, current IR systems perform stemming or lemmatization prior to indexing or retrieval), term similarity (instead, queries are expanded with synonyms prior to retrieval), or grammar (the closest thing to a LM captured by the index is word frequency distributions).

Over the past few years, advances in representation learning resulted in a shift away from traditional inverted indexes towards dense vector-based indexes (or hybrid inverted + vector-based indexes) [28, 40, 41, 44, 45, 49, 102]. These indexes encode semantically-rich document representations that primarily help improve recall by overcoming the vocabulary mismatch problem that is known to plague inverted indexing-based systems.

Many language understanding advances have already been successfully leveraged by IR researchers. For example, representation learning has been used for retrieval purposes, large pre-trained LMs are being leveraged for scoring, etc. These efforts have yielded significant improvements across a range of tasks.

Despite all of this progress, today's cutting edge IR systems are not fundamentally different than classical IR systems developed many decades ago. Indeed, a majority of today's systems boil down to: (a) building an efficient queryable index for each document in the corpus, (b) retrieving a set of candidates for a given query, and

(c) computing a relevance score for each candidate. This *index-retrieve-then-rank* blueprint has withstood the test of time and has rarely been challenged or seriously rethought.

This paper envisions a unified model-based approach to building IR systems that eliminates the need for indexes as we know them today by encoding all of the knowledge for a given corpus in a model that can be used for a wide range of tasks. As the remainder of this paper shows, once everything is viewed through a model-centric lens instead of an index-centric one, many new and interesting opportunities emerge to significantly advance IR systems. If successful, IR models that synthesize elements of classical IR systems and modern large-scale NLP models have the potential to yield a transformational shift in thinking and a significant leap in capabilities across a wide range of IR tasks, such as document retrieval, question answering, summarization, classification, recommendation, etc.

2 RELATED WORK

This section provides a brief survey of research related to document retrieval, question answering, knowledge bases, and large pre-trained LMs, as those are the research directions that are most relevant and closely aligned to the envisioned system.

2.1 Document Retrieval

Document retrieval is the quintessential IR task, and as such has a rich history. Rather than undertake a comprehensive literature review here, we instead focus on three specific lines of important recent research that have culminated in the current state-of-the-art document retrieval systems.

The first such line of research is *learning to rank*, which was propelled by the commercial success of search engines and easy access to large volumes of user interaction data. This movement represented a transformational leap beyond traditional TF.IDF-based IR systems. There is a vast and continually growing body of literature focused on this topic. Interested readers are encouraged to see [48] and [50] for more details.

The next line of research is *neural-based re-ranking models*. This line of research can be thought of as a specific application of neural networks to the problem of learning to rank. These models typically take documents retrieved in some way (e.g., from a traditional inverted index or dense vector index) and use neural network-based models to score or rank documents. Some examples of such models include Deep Relevance Matching Model (DRMM) [32], DUET [60], Kernel-based Neural Ranking Model (KNRM) [101], Position-Aware Convolutional-Recurrent Relevance (PACRR) [34], and Context-Aware PACRR (co-PACRR) [35]. This is a highly active area of research, with continuous progress as a result of newer, better modeling architectures, novel uses of data, etc. For more information on this topic, interested readers should see [59] and [62].

The third and final line of research is *representation learning*. The goal of representation learning is to encode queries and/or documents into (often dense) vector representations. These representations can be used for a variety of purposes including retrieval, for example via efficient k -nearest neighbor search. There have been many such approaches proposed in the literature [28, 40, 41,

44, 45, 49, 102]. One of the key benefits of these approaches over term-based representations is that the encodings often capture rich semantics and provide a way of overcoming the well-known vocabulary mismatch problem. However, one of the shortcomings of these approaches is that the recall improvements they bring often come at the cost of reduced precision compared to term-based representations.

The culmination of these three lines of research represent the current state-of-the-art retrieval systems. Today’s state-of-the-art systems often rely on a combination of term-based (i.e., retrieval over an inverted index) and semantic (i.e., retrieval over an index of dense vector representations) retrieval to generate an initial set of candidates. This set of candidates is then typically passed into one or more stages of re-ranking models, which are quite likely to be neural network-based learning-to-rank models. As mentioned previously, the index-retrieve-then-rank paradigm has withstood the test of time and it is no surprise that advanced machine learning and NLP-based approaches are an integral part of the indexing, retrieval, and ranking components of modern day systems.

2.2 Question Answering

Early research into question answering systems primarily focused on ranking and retrieval, whereby models are trained to learn a relevance score between a user question and candidate answers [75, 81, 92, 104]. Due to the nature of how the task is defined, systems typically rely on advances in short text matching [73], paraphrase identification [33], and entailment detection [63, 84]. More recently, a large number of neural network-based models have been designed for question answer matching and have made significant progress on the problem [75, 85, 93].

As time goes on, there has been a slight shift in how the question answering problem is expressed. The development of new neural network-based modules, pointer networks [90], and consequently the Match-LSTM with answer pointers [94] have unlocked the potential for highly effective extractive question answering. Instead of ranking question-answer pairs, the new pointer mechanism enables the extraction of answer spans within passages. As such, this spurred significant growth in the number of models proposed for QA tasks of this sort [71, 88]. Likewise, a surge of neural network-based models, primarily attention-based [86, 97, 105], have been developed for tackling this question answering task variant.

The typical setup of machine reading comprehension (MRC) systems involve a query and a context (passage). In the real world, these passages do not appear out of thin air, i.e., ultimately they have to be retrieved from somewhere. This motivated another variant of the QA problem which is commonly referred to as open domain question answering [21, 25, 39]. Here, the goal is to first retrieve the relevant passages such that an MRC model can extract the correct answer [15]. To this end, there have been multiple innovations on this front, such as jointly learning or modeling interactions between the retrieval system and the MRC model [17, 95]. Hence, retrieval still remains a core component of QA systems, especially when the corpus is large [40].

The final class of QA systems are generative ones. In retrieval and/or span-based QA systems, it is always assumed some notion of ground truth exists in either the provided passages or amongst the

answer candidates. Generative QA systems shift this burden to the generative model whereby the only assumption is that the answer exists in the generator’s output vocabulary. Historically, this has been thought of as significantly more challenging than extractive or retrieval-based QA [43, 80, 87]. Today, large pre-trained encoder-decoder (seq2seq) models such as T5 [70] and BART [47] have demonstrated that state-of-the-art QA performance can be achieved via generative models.

2.3 Explicit Knowledge Bases

During the early 2000s, research momentum around the Semantic Web [5] and pre-existing “old-style” AI research gave rise to graph-structured knowledge bases, including Freebase [6], WikiData [91], the Google Knowledge Graph [77], and Microsoft Satori [68]. A knowledge base is typically realized as a set of triplets – a pair of *entities* and a *predicate* relating them. The triplets induce a graph structure, with entities as the nodes and predicates as labeled edges. Knowledge graphs are well-suited to represent factoids (e.g. “Thomas Edison invented the light bulb”), and query algebras over the graph structure make it possible to form short chains of relations. Originally assembled and curated by hand, there has been much research on automatically extracting knowledge graph triplets from Web pages, including Yago [78], NELL [12, 58], and Knowledge Vault [22]. Google leverages its Knowledge Graph when generating “Knowledge Panels” (cards containing a collection of factoids directly embedded in the results page) in response to a factoid-seeking query. These direct answers bring us some of the way towards our vision of expert advice; however, they are limited by the size of the graph, which only represents a fraction of the information contained in the Web corpus, and the inability to provide nuanced answers (by definition, answers are limited to factoids).

2.4 Pre-Trained Language Models

Over the past few years, pre-trained LMs have had a significant impact on the field of NLP. Models such as like BERT [20], RoBERTa [51], XLNet, T5 [70], BART [47] GPT-2 [69], GPT-3 [10], and Meena [1] are state-of-the-art for most (if not all) NLP tasks. The key idea behind pre-trained LMs is to first pre-train using one or more generative tasks, after which one may simply apply the pre-trained model to downstream tasks by fine-tuning their parameters. To this end, language modeling [10], masked language modeling [20], and encoder-decoder based generation [70] have proven to be highly effective pre-training approaches.

One of the core reasons why large pre-trained LMs are so successful is that they learn highly effective contextual representations. Research on learning contextual representations dates back to early work of learning semantic word vectors whereby models like Skip-Gram [55, 56] and GloVe [65] helped revolutionize the field of NLP. Subsequently, pre-trained models such as CoVe [52] and ELMo [66] have also demonstrated the benefits of more sophisticated pre-training objectives and model architectures.

Today, pre-trained LMs are generally based on Transformer models [89]. Unlike predecessors that are trained largely on recurrent neural network models [52, 66], the Transformer’s ability to be parallelized efficiently enables practitioners and researchers to greatly

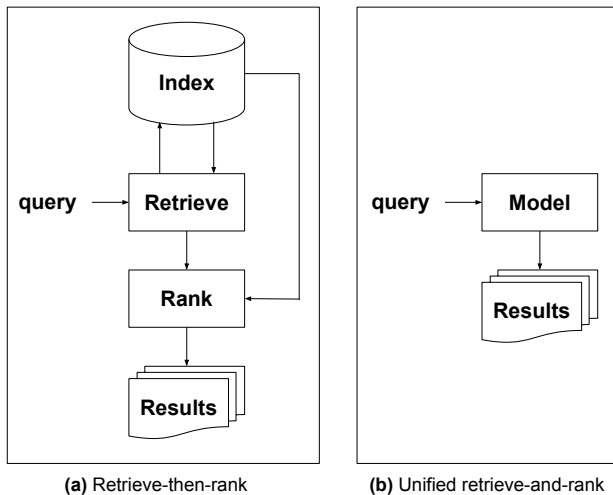


Figure 1: High-level schematics of the traditional index-retrieve-then-rank (left) and model-based (right) paradigms.

scale these models. Large-scale models have shown to generalize better, as shown in the zero- and few-shot experiments of [10], and achieve significantly better performance [70]. Many of the largest models are billion-scale, with the largest T5 model reaching 11 billion parameters and GPT-3 reaching 175 billion parameters. Very recently, Switch Transformers [26] broke through the *trillion* parameter ceiling. As such, it is common to associate pre-trained LMs with size, which has resulted in such models being referred to as large LMs (LLMs) or giant LMs (GLMs).

Pre-trained LMs such as GPT-3 have demonstrated impressive text generation capabilities. In fact, some of the text synthesized by such models are indistinguishable from text written by humans [37].

3 MODEL-BASED INFORMATION RETRIEVAL

We begin the more technical portion of the paper by posing the following questions:

- What would happen if we got rid of the notion of the index altogether and replaced it with a large pre-trained model that efficiently and effectively encodes all of the information contained in the corpus?
- What if the distinction between retrieval and ranking went away and instead there was a single response generation phase?

Recent breakthroughs in natural language understanding (e.g., BERT), large-scale language modeling, few-shot learning, and multi-task learning (e.g., T5) provide supporting evidence that these questions are not as far-fetched as they may have been just a couple of years ago. Indeed, the confluence of these advances has created a unique opportunity to meaningfully explore answers to these questions.

This section describes a modeling approach that synthesizes aspects of modern IR systems and NLP models. The approach, referred to as *model-based information retrieval*, is meant to replace

the long-lived "retrieve-then-rank" paradigm by collapsing the indexing, retrieval, and ranking components of traditional IR systems into a single unified model. With model-based IR, indexing is replaced with model training, while retrieval and ranking are replaced with model inference. See Figure 1 for a high-level schematic of these two paradigms.

It is of course important to acknowledge that models are already used everywhere in modern IR systems. The important distinction between the systems of today and the envisioned system is the fact that a unified model replaces the indexing, retrieval, and ranking components. In essence, it is referred to as model-based because there is nothing but a model.

This represents a fundamentally different way of thinking about IR systems. Within the index-retrieve-then-rank paradigm, modeling work (e.g., query understanding, document understanding, retrieval, ranking, etc.) is done on top of the index itself. This results in modern IR systems being comprised of a disparate mix of heterogeneous models (e.g., one model used to learn document representations, another for document understanding, and yet another for ranking). Within the model-based information retrieval paradigm, the model and the index are one. Everything that was previously developed on top of the index is now integrated directly into a single, unified model. The model itself is built from the corpus, just like indexes are built from the corpus, but the encoded information is expected to be much more complex and able to solve a wider range of tasks.

For example, for question answering tasks our envisioned model is able to synthesize a single answer that incorporates information from many documents in the corpus, and it will be able to support assertions in the answer by referencing supporting evidence in the corpus, much like a properly crafted Wikipedia entry supports each assertion of fact with a link to a primary source. This is just one of many novel tasks that this type of model has the potential to enable.

The following sub-sections dive deeper into some of the fundamental building blocks that are necessary for this model-based approach to be possible.

3.1 Beyond Language Models

Large-scale pre-trained LMs have proven to be useful for a wide range of NLP and IR tasks. However, such models fundamentally work on the term-level. Common natural language tasks, like masked language modeling, typically take a sequence of terms as input and produce one or more terms as output.

As the literature clearly demonstrates, there are many ways to represent queries and documents with such models. However, nearly all previously proposed work first tokenizes queries and/or documents into sequences of terms that are then passed as input to some model. The output of the model can then be used in a variety of ways. For example, embeddings can be used as learned representations, generated terms can be used to augment an inverted index, the models can be fine-tuned and used for ranking, and so on.

This approach is obviously quite useful, but it does have a number of limitations. LMs that are purely learned over sequences of terms have no way of explicitly modeling relationships between terms and documents. LMs essentially learn *assertions* ("The sky is blue.")

from the corpus they are trained over but fail to learn *associations* between terms and individual documents. This is why we refer to pre-trained LMs as dilettantes – they are perceived to know a lot but their knowledge is skin deep.

Given that such models only know about sequences of terms, it is not possible to provide higher-level entities (like document ids) as input or expect document ids to be produced as output without making some changes to the underlying model. To replace indexes with a single, unified model, it must be possible for the model itself to have knowledge about the universe of document identifiers, in the same way that traditional indexes do. One way to accomplish this is to move away from traditional LMs and towards *corpus models* that jointly model term-term, term-document, and document-document relationships.

Of course, modern LMs are trained over a corpus of word sequences, and therefore can be considered rudimentary corpus models. However, since such models are agnostic to higher-level corpus structure and document properties, they fall far short of being faithful models of a corpus.

Corpus models, as defined, can take as input a sequence of terms and output one or more terms or one or more document ids. Similarly, the model could take as input a mix of terms and document ids and output one or more terms and/or document ids. By explicitly encoding the associations between terms and documents, the model suddenly becomes able to "natively" retrieve (and score) documents without the need for a traditional index.

How to actually build corpus models that are both efficient (at training and inference time) and effective is an open research question that spans multiple research communities. There are many obvious things that can be tried, such as adding a sentinel token to the vocabulary for each document identifier, but then the question immediately becomes how can one meaningfully pre-train such a model? Another option is to connect document identifiers to input or output sequences of terms using a separate model or procedure, which might be more scalable but is less unified as it would likely need to be done "outside" of the unified model itself. An option is to investigate early work in learning document representations, i.e., doc2vec or paragraph2vec [56] that learns embeddings for documents by infusing document identifiers in the pre-training stage. However, this raises additional questions of how to incrementally update the index. Should additional training stages be incorporated so that a model may learn new document-term associations?

Another research challenge is how to effectively scale the number of document identifier tokens. Document identifiers have to be allocated as extra ids in the output layers of the language model, which can substantially increase the number of model parameters. Clearly, when there is no upper bound on the total number of documents, as is often the case in dynamic corpora, this quickly becomes a concern. Some options include representing document identifiers as sequences of subwords (or characters), factorizing the id space in some way, or storing identifiers in some form of structured memory module.

Overall, this is an important and potentially highly impactful, but long-overlooked, line of research that could benefit both the IR and NLP communities.

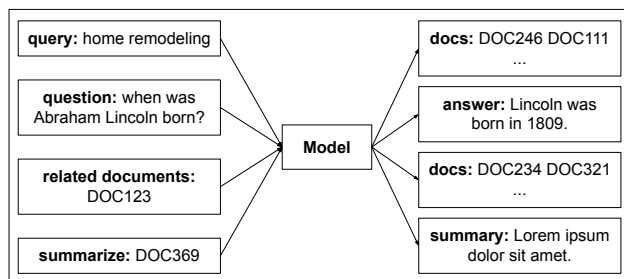


Figure 2: Example of how a single unified model can be leveraged to solve a wide range of IR tasks. This example shows a model that handles document retrieval, question answering, related document retrieval, and document summarization tasks.

3.2 Multi-Task Learning: Towards A Single Model for all Information Retrieval Tasks

We envision using the same corpus model as a multi-task learner for multiple IR tasks. To this end, once a corpus model has been trained, it can of course be used for the most classical of all IR tasks – document retrieval. However, by leveraging recent advances in multi-task learning, such a model can very likely be applied to a diverse range of tasks.

By leveraging a multi-task text-to-text corpus model with appropriately defined pre-training objectives and fine-tuning tasks, one can envision a "one giant model" approach to IR that can be used for document retrieval, question answering, summarization, and new tasks such as the aspirational task of providing expert advice that was described in the introduction.

The T5 model [70] and follow-ups demonstrated that it is possible to achieve state-of-the-art performance across multiple tasks with a single unified model. The key idea is to leverage task conditioning via a task identifier that tells the model which task it is supposed to perform. The T5 model has been shown to achieve state-of-the-art on several challenging language understanding benchmarks. Hence, it is expected that a sufficiently high quality corpus-based model trained in a similar manner would be capable of equally strong performance across multiple tasks of interest.

Figure 2 demonstrates what this might look like from a practical perspective where the input to such a unified model is a task-prefixed request and the output is a response that satisfies the request. In this figure, note that the model is able to perform tasks defined over mixtures of terms and document identifiers. Such a setup would provide significantly more flexibility than the pure term-based LMs that are widely used today.

The tasks in this figure include:

- **Document retrieval.** The input is a query string and the output is one or more relevant document identifiers.
- **Question answering.** The input is a question in natural language and the output is a natural language response.
- **Related document retrieval.** The input is a document identifier and the output is a one or more relevant document identifiers.

- **Document summarization.** The input is a document identifier and the output is a summary of the document.

These are obviously only for illustrative purposes and there is no limit to the types of tasks that could potentially be incorporated into such a model.

Moving towards "one giant model" for all of IR opens up a number of potentially interesting and impactful research directions that span machine learning, NLP, and IR.

3.3 Zero- and Few-Shot Learning

Another advantage of large pre-trained models is their ability to perform well in zero- and few-shot learning situations. This is particularly appealing for tasks where limited training data is available. Indeed, many real-world IR systems have access to very little in the way of labeled data. For this reason, being able to generalize well based on a small number of labeled examples has the potential to yield significant practical utility.

Zero- and few-shot learning is common in document ranking tasks. For example, *ad hoc* retrieval can be thought of as a zero-shot learning task since no examples of relevant documents for the given query are actually provided to the system. Furthermore, relevance feedback can be thought of as few-shot learning, since the user manually provides labels for one or more documents that the system can use to improve its ranking.

Building upon the general unified modeling paradigm developed in the previous sub-sections, these tasks can easily be defined as follows:

Ad Hoc Retrieval (zero-shot).

- **Input:** *query*
- **Output:** $reldoc_1, \dots, reldoc_n$

where *query* is a query string and $reldoc_i$ are document identifiers.

Pseudo-relevance feedback (few-shot).

- **Input:** $(query_1, doc_1), \dots, (query_n, doc_n)$ *query*
- **Output:** $reldoc_1, \dots, reldoc_n$

where $(query_i, doc_i)$ are pairs of query strings and document identifiers that have been labeled as relevant in some way and $reldoc_i$ are document identifiers. In this way, the labeled query/document pairs are provided as context to the system to enable few-shot learning for the current *query*.

Beyond document retrieval, unified models can be used in a few-shot learning setting for other tasks, including query understanding and document understanding. For example,

Query Understanding (few-shot).

- **Input:** $(query_1, intent_1), \dots, (query_n, intent_n)$ *query*
- **Output:** *intent*

where $(query_i, intent_i)$ are pairs of query strings and intents (categories) that have been labeled in some way. These are passed as context to the model, which then uses them to generalize to identify the best *intent* associated with *query*.

Document Understanding (few-shot).

- **Input:** $(doc_1, label_1), \dots, (doc_n, label_n)$ *doc*
- **Output:** *label*

where $(doc_i, label_i)$ are pairs of document identifiers and document labels. The model then takes *doc* as input and generates *label* as output.

As these examples show, having a unified multi-task model that understands the connections between sequences of terms and document identifiers opens up a wide range of straightforward and powerful use cases, even when there is only limited labeled data available, in an extremely straightforward manner.

3.4 Response Generation

Using a T5-like setup or more generally any encoder-decoder model, it is possible to leverage the model to generate a wide range of potential output types. As described in the previous sub-sections, these outputs could be sequences of terms, document identifiers learned as part of a unified corpus model, query intent or document category labels learned as a result of fine-tuning or few-shot learning, and so on.

An aspirational goal would be a retrieval system that, given the query "what are the health benefits and risks of red wine," would give you a coherent and authoritative answer laying out the evidence for both benefits and risks. However, in today's world of search engines and pre-trained LMs, you end up with less-than-ideal responses. Figure 3 shows the responses by a modern day search engine (left) and modern pre-trained LM for this example query. The search engine identifies a single relevant Web page and extracts a snippet that attempts to, but fails to sufficiently answer the question. On the other hand, the pre-trained LM returns a coherent, focused response that seemingly answers the question but provides no context for the answer or any sense of how authoritative or comprehensive it is. The system envisioned in this paper would instead be able to produce a response like the one on the right. This response provides references to source material making it much easier to highlight the authoritativeness of the content. This simple example shows how deeper connections between sequences of words and documents can be useful.

There are many possible ways to approach this problem. The model itself can understand both terms and documents and their relationships and be trained to generate content with proper citations. This alone is a major challenge in terms of how to define the task, where labeled (or weakly labeled) data might be sourced from, how to evaluate the output, etc. Another possibility is to use a standard pre-trained LM and add a *learning to cite* task on top of it that can be used to artificially ground synthesized text to articles in the corpus. This solution has a number of drawbacks, including the fact that the generation and citation processes are disjoint and hence may lead to incongruous outputs. On the other hand, jointly performing generation and citation will likely be more challenging but is likely to yield better results. A potential approach could perhaps be in similar spirit to learning a mixture of distributions [53, 74] and adaptively learning to toggle between generating document identifiers and raw tokens.

There are also a number of other major challenges associated with generating high quality responses. A response is high quality if it exhibits the following properties:

- **Authoritative.** Responses should generate content by pulling from highly authoritative sources. This is another reason

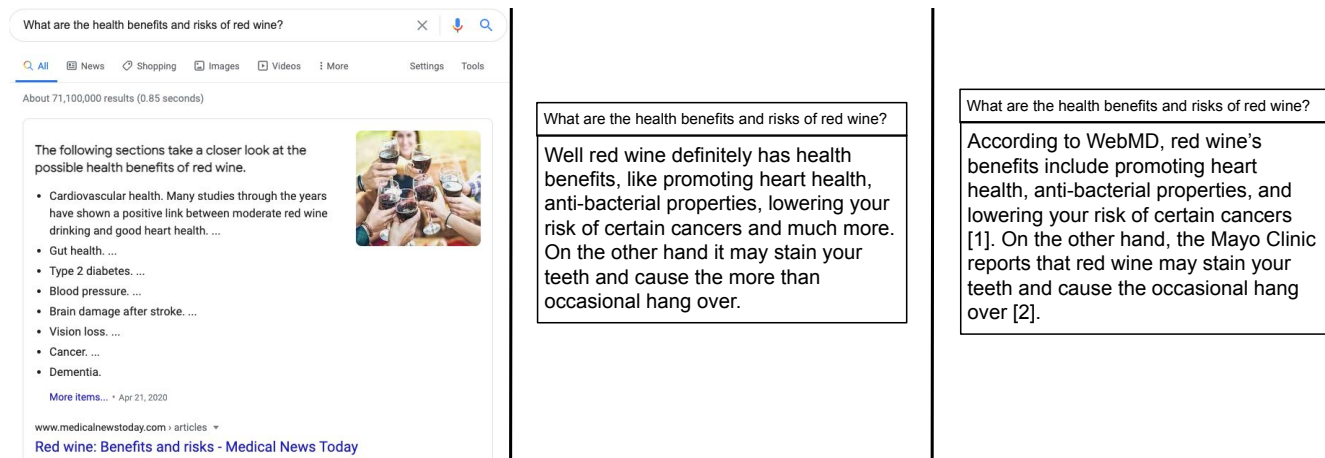


Figure 3: Example Web search (left), pre-trained language model (middle), and expert (envisioned system; right) responses for the query "What are the health benefits and risks of red wine?".

why establishing more explicit connections between sequences of terms and document metadata is so crucial. If all of the documents in a corpus are annotated with an authoritativeness score, that score should be taken into account when training the model, generating responses, or both.

- **Transparent.** Whenever possible, the provenance of the information being presented to the user should be made available to them. Is this the primary source of information? If not, what is the primary source?
- **Unbiased.** Pre-trained LMs are trained to maximize their predictive power on their training data, and thus they may reflect societal biases in that data [4, 36, 76]. To address those risks, designers of systems that employ pre-trained LMs may consider different training objectives [98] and also surround the model with additional safeguards against biased system responses.
- **Diverse perspectives.** Generated responses should represent a range of diverse perspectives but should not be polarizing. For example, for queries about controversial topics, both sides of the topic should be covered in a fair and balanced way. This obviously has close tie-ins with model bias.
- **Accessible.** Written in terms that are understandable to the user. For example, responses that discuss complex medical issues should be written in as-plain-as-possible terms. Another example is authoritative content that may only be written in a certain language that is different than the one that the user issued their query in. In this situation, the system should provide a translated version of the response to the user.

This list is obviously not exhaustive but hopefully drives home the point that extreme care must be taken to ensure that synthesized responses are indeed high quality. Doing so will require a significant amount of research across multiple disciplines. Even simply defining a measure of synthesized answer quality that takes into account all of these factors (and more) is itself an important but difficult research challenge. Building these principles into the model will be even more challenging.

3.5 Reasoning Capabilities

One key advantage of adopting a model-based index is that we can leverage the modularity of neural networks for composing new modules with specialized forms of inductive bias. While most pre-trained LMs today are based on the Transformer model architecture, such models may be augmented by composing them with one or more additional neural modules. For example, we may imbue the model with reasoning-like capabilities by allowing the model to attend over an external memory. To this end, neural modules that provide a memory-like inductive bias to enable memory lookups [57, 99] or content addressing (e.g., differentiable neural computers [30]) may be explored. Other forms of inductive biases such as multi-hop reasoning [2, 13, 107] might also be useful.

Within the context of neural retrieval from an encoder-decoder model, it may also be possible to incorporate relational inductive biases [2, 18] to model relationships amongst candidate documents and terms in the decoder. For instance, when learning to output document identifiers, the decoder learns what is already being partially generated and performs self-attention across the partial generation. While a simple way of thinking of this is through the lens of listwise learning-to-rank, there is significantly more flexibility in incorporating relational reasoning as a neural component where the model learns this in a data-driven manner. Conversely, it is not possible to develop systems that exhibit these reasoning-like properties with traditional indexes and classical IR models.

While the ability to reason is a nice characteristic for such models to have, it may also result in unfavorable outcomes. Specifically, large sequence-to-sequence neural network-based models are prone to hallucination [46]. Hallucination has the potential to generate novel truthful outputs, but it also has the potential for to result in strange, untrue, or downright offensive outputs as well. This is an issue that is prevalent across all modern pre-trained LMs and one that will need to be addressed in some way (e.g., via the system outputting logical explanations) before their outputs can be trusted in a similar way as a human expert.

3.6 Arithmetic, Logical, Temporal, and Geographical Reasoning

It is well established that modern search engines can handle queries that deal with some form of arithmetic reasoning. For example, converting across currencies, i.e., ‘*36,500 USD to pounds*’, *2am PST to GMT-2*’ and ‘*how far is California from New York City*’. Current search engines behave as if they have some sense of order, time, logic, and even geographical distance. While there has been recent work that investigates these aspects in the context of neural network models [72], it remains challenging to develop neural models that can reliably and accurately deliver on these types of reasoning capabilities. Moreover, at a foundational level, LMs that can handle numerical or temporal reasoning can be quite crucial even for document understanding, as some form of numerical commonsense may be required to fundamentally understand the content of documents.

3.7 Unifying Modalities in a Single Model

Another key advantage of the model-based paradigm is that it allows multiple modalities to be unified within a single model. Documents traditionally contain a significant amount of metadata and/or media content, such as images, videos, and audio. Traditionally, image search and document search leverage very different indexes. Having a unified model capable of handling multiple modalities can bridge this gap.

There has been a great deal of recent interest in vision-based Transformers [24] and vision-based T5 [14]. Such models may provide a means for exploring multi-modal grounding as a way of enabling effective text and image representations. This typically involves having a separate encoder for each modality, making it rather straightforward to integrate into existing models.

Other modalities of interest include tabular data and traditional features (e.g., document metadata) that are passed to standard machine learning models. These supplementary features may be generated from another network (e.g., embeddings) or handcrafted by experts. Here, it is an open research question how to integrate these auxiliary features into these pre-trained models.

3.8 Leveraging Document and Corpus Structure

Successful modern IR systems fully leverage all of the rich structure associated with documents. For example, terms that appear in the title or anchor text are often treated as more important for Web search applications. Today’s modern pre-trained LMs fail to take this rich document structure into account. How to successfully model and leverage rich document structure is an interesting direction of future research that could provide significant benefit to IR-centric applications of pre-trained LMs.

In an open corpus such as the web, not all documents are equally authoritative or trustworthy. There are many known techniques for estimating the authority or veracity of a Web page, from fact-checking claims within a single page [38] to aggregating quality signals at the logical domain level [23]. Incorporating such documents authority signals, along with other signals such as polarity or toxicity [100] of the content, into a giant language model is crucial to ameliorating biases that such models are prone to learn from unvetted documents “in the wild”.

Furthermore, many corpora have some form of explicit or implicit graph structure associated with them [9]. Modern IR systems leverage such graphs in a number of ways, such as computing graph-based measures like PageRank [7], identifying hubs and authorities [42], and so on. There are many ways that this structure can also be leveraged within the proposed framework. For example, the graph structure can be used for cross-document pre-training (i.e., finding similar documents and packing them into the same sequence for pre-training [11]), thereby enabling the model to benefit from long-range dependencies and cross-document language understanding. Another potential way to leverage the graph structure is to define a co-training task that predicts if there is an edge between two documents. How to best leverage corpus structure within pre-trained LMs is an important and interesting open research question.

3.9 Scaling to Multiple Languages

Another key research question is whether it is possible to model all documents across languages within a single model. Practically, this has implications both in terms of vocabulary and model capacity. If this can be effectively addressed, the envisioned model would be able to support cross-lingual generalization and be applied to tasks like cross-language IR [61]. Early work has shown that this is already possible, given the success of multilingual T5 [103] and other multilingual pre-trained models [67]. Despite these early success, many important research challenges remain, such as determining the optimal proportions of training data from each language to use to effectively learn balanced representations.

3.10 On the Challenges of Scale

The modeling efforts outlined in this paper can be generally regarded as incredibly resource intensive. There are multiple perspectives to the challenge of scale of this overall endeavour.

Firstly, there is question of model capacity and exactly how large of a model would be required to fit multiple tasks, billions of documents, document identifiers across a dozen of languages. We postulate that models need to go beyond a billion parameters to be effective in having enough capacity. However, models with large parameter footprints are difficult to serve in practice.

Secondly, documents are generally long, easily spanning a thousand or more subwords. Additionally, modeling multiple documents can incur additional substantial costs. The problem of scale within the context of long document understanding with pre-trained LMs is a well-established problem [3, 83]. Dozens of models have been proposed to solve this issue but still sacrifice performance for memory efficiency [82].

We believe that there are multiple research directions that enable us to scale. In terms of modeling capacity, a potential solution is to leverage models with a large number of parameters (e.g., trillions of parameters) but maintain the computation cost and inference time of a model an order of magnitude smaller. One good example of a recent model along these lines is the Switch Transformer [26]. Models that leverage dynamic and conditional computation to select and activate certain sub-networks may be key to allow such systems to scale. These models also fit into the overall paradigm of modeling

multiple tasks in a single network - since intuitively a model should only select the relevant sub-network for certain tasks.

Complementing the challenges around scaling model size are those around *sample efficiency*. For the model to be truly knowledgeable, they must be trained over a diverse distribution of data. However, naively training on the oceans of data available in the wild in their entirety may not be feasible. Techniques that can “condense” or “distill” massive training datasets into smaller, more manageable ones, with little loss in information will likely need to be employed [96, 106].

3.11 Incremental Learning

There are a large number of research and engineering challenges associated with keeping such models up-to-date in the presence of potentially highly dynamic corpora. For example, it is an open question as to how models can be built in a manner such that it is efficient (and effective) to add new documents to the model. “On-line” or “incremental” learning explores the problem of updating machine learned models as new data arrives sequentially in a way that does not harm performance on previous data, a phenomenon known “catastrophic forgetting” [27]. The “continual learning” setup generalizes this and studies models and techniques for learning on new tasks without forgetting old ones. While many methods have been proposed (see [19, 64] for a survey), it has mostly been studied on toy datasets and synthetic setups in a low-parameter count regime. Investigating whether current methods work on large generative models trained with colossal datasets remains an open and important research direction.

Even more interestingly and more challenging is the problem of having models “forget” everything that they know about a document that was removed from the corpus. This becomes even more challenging in situations where privacy or legal reasons require that all traces of a deleted piece of content be removed from a system, which is a typical requirement when building practical IR systems.

3.12 Model Interpretability, Controllability, and Robustness

Since the operating mechanism of classical term-based IR systems is transparent to designers, how the system will behave on test queries is often predictable. Deviations from desired behavior are easier to debug and can even be fixed by manually adding new rules, although such manual interventions and hard-coded rules are hard to scale. In contrast, it is well-known that modern deep neural networks suffer from interpretability issues, and addressing them is an active line of research (e.g. [79]; see [31] for a survey). Furthermore, even after an issue with the model has been identified, it is often unclear what modeling knobs one should turn to fix the model’s behavior. A desiderata then is that the model should be both interpretable and debuggable as well as *controllable*, i.e. the model designer should know how to control the behavior of the trained model, e.g. by modifying training data or tuning hyper-parameters in the loss function. Of equal importance is robustness. For example, should the search user make the benign typo: “the” → “teh” in an otherwise good query, we expect that the model’s response will not change drastically. Crucially, we would like the model to be well-behaved for queries it may not have seen before, including

adversarial examples [29] that can occur not due to malicious intent by the user but rather by bad luck, which is inevitable in search systems serving a large user base.

3.13 Putting It All Together

If all of these research ambitions were to come to fruition, the resulting system would be a very early version of the system that we envisioned in the introduction. That is, the resulting system would be able to provide expert answers to a wide range of information needs in a way that neither modern IR systems, question answering systems, or pre-trained LMs can do today.

Some of the key benefits of the model-based IR paradigm described herein include:

- It abstracts away the long-lived, and possibly unnecessary, distinction between “retrieval” and “scoring”.
- It results in a unified model that encodes all of the knowledge contained in a corpus, eliminating the need for traditional indexes.
- It allows for dozens of new tasks to easily be handled by the model, either via multi-task learning or via few-shot learning, with minimal amounts of labeled training data.
- It allows seamless integration of multiple modalities and languages within a unified model.

4 CONCLUSIONS

This paper envisions an ambitious research direction that doubles down on the synthesis between modern IR and NLP to deliver on the long-promised goal of providing human expert quality answers to information needs. Specifically, the paper makes the case for developing retrieval systems that combine the best elements of document retrieval systems and pre-trained language models. To accomplish this, a so-called model-based information retrieval framework is proposed that breaks away from the traditional index-retrieve-then-rank paradigm by encoding the knowledge contained in a corpus in a unified model that replaces the indexing, retrieval, and ranking components of traditional systems. It was argued that if successful, such a unified model can be used to solve a wide range of tasks (via multi-task learning), can easily adapt to new low resource tasks and corpora (via zero- and few-shot learning), and can be used to synthesize high quality responses that go well beyond what today’s search and question answering systems are capable of.

There are a number of interesting and difficult research and engineering challenges that must be solved before the envisioned system can be realized. These challenges span the IR, NLP, and machine learning research disciplines, and will require interdisciplinary research to be successful. Some of the major challenges include modeling (moving from LMs to corpus model), training (pre-training objectives, fine-tuning task definitions), response generation (authoritativeness, bias mitigation), and scalability (indexing and serving).

REFERENCES

- [1] Daniel Adiwardana, Minh-Thang Luong, David R So, Jamie Hall, Noah Fiedel, Romal Thoppilan, Zi Yang, Apoorv Kulshreshtha, Gaurav Nemade, Yifeng Lu, and Quoc V Le. 2020. Towards a Human-like Open-Domain Chatbot. *arXiv preprint arXiv:2001.09977* (2020).

- [2] Akari Asai, Kazuma Hashimoto, Hannaneh Hajishirzi, Richard Socher, and Caiming Xiong. 2020. Learning to Retrieve Reasoning Paths over Wikipedia Graph for Question Answering. In *Proceedings of the 8th International Conference on Learning Representations (ICLR '20)*.
- [3] Iz Beltagy, Matthew E Peters, and Arman Cohan. 2020. Longformer: The Long-Document Transformer. *arXiv preprint arXiv:2004.05150* (2020).
- [4] Emily M. Bender, Timnit Gebre, Angelina McMillan-Major, and Shmargaret Shmitchell. 2021. On the Dangers of Stochastic Parrots: Can Language Models Be Too Big?. In *Proceedings of the 2021 ACM Conference on Fairness, Accountability, and Transparency (FAccT '21)*. 610–623.
- [5] Tim Berners-Lee, James Hendler, and Ora Lassila. 2001. The Semantic Web. *Scientific American* 284, 5 (May 2001), 34–43.
- [6] Kurt Bollacker, Colin Evans, Praveen Paritosh, Tim Sturge, and Jamie Taylor. 2008. Freebase: A Collaboratively Created Graph Database for Structuring Human Knowledge. In *Proceedings of the 2008 ACM SIGMOD International Conference on Management of Data (SIGMOD '08)*. 1247–1250.
- [7] Sergey Brin and Lawrence Page. 1998. The Anatomy of a Large-Scale Hypertextual Web Search Engine. In *Proceedings of the 7th International Conference on World Wide Web (WWW '98)*. 107–117.
- [8] Andrei Broder. 2002. A Taxonomy of Web Search. *SIGIR Forum* 36, 2 (September 2002), 3–10.
- [9] Andrei Broder, Ravi Kumar, Farzin Maghoul, Prabhakar Raghavan, Sridhar Rajagopalan, Raymie Stata, Andrew Tomkins, and Janet Wiener. 2000. Graph structure in the Web. *Computer Networks* 33, 1–6 (2000), 309–320.
- [10] Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language Models are Few-Shot Learners. In *Proceedings of the 34th Conference on Neural Information Processing Systems (NeurIPS '20)*.
- [11] Avi Caciularu, Arman Cohan, Iz Beltagy, Matthew E Peters, Arie Cattan, and Ido Dagan. 2021. Cross-Document Language Modeling. *arXiv preprint arXiv:2101.00406* (2021).
- [12] Andrew Carlson, Justin Betteridge, Richard C. Wang, Estevam R. Hruschka, and Tom M. Mitchell. 2010. Coupled Semi-Supervised Learning for Information Extraction. In *Proceedings of the 3rd ACM International Conference on Web Search and Data Mining (WSDM '10)*. 101–110.
- [13] Jifan Chen, Shih-ting Lin, and Greg Durrett. 2019. Multi-hop Question Answering via Reasoning Chains. *arXiv preprint arXiv:1910.02610* (2019).
- [14] Jaemin Cho, Jie Lei, Hao Tan, and Mohit Bansal. 2021. Unifying Vision-and-Language Tasks via Text Generation. *arXiv preprint arXiv:2102.02779* (2021).
- [15] Christopher Clark and Matt Gardner. 2018. Simple and Effective Multi-Paragraph Reading Comprehension. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics, Volume 1 (Long Papers) (ACL '18)*. 845–855.
- [16] Bruce Croft, Donald Metzler, and Trevor Strohman. 2009. *Search Engines: Information Retrieval in Practice* (1st ed.). Addison-Wesley Publishing Company, USA.
- [17] Rajarshi Das, Shehzaad Dhuliawala, Manzil Zaheer, and Andrew McCallum. 2019. Multi-step Retriever-Reader Interaction for Scalable Open-domain Question Answering. In *Proceedings of the 7th International Conference on Learning Representations (ICLR '19)*.
- [18] Nicola De Cao, Wilker Aziz, and Ivan Titov. 2019. Question Answering by Reasoning Across Documents with Graph Convolutional Networks. 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) (NAACL-HLT '19)*. 2306–2317.
- [19] Matthias De Lange, Rahaf Aljundi, Marc Masana, Sarah Parisot, Xu Jia, Ales Leonardis, Gregory Slabaugh, and Tinne Tuytelaars. 2021. A continual learning survey: Defying forgetting in classification tasks. *IEEE Transactions on Pattern Analysis and Machine Intelligence* (2021).
- [20] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers) (NAACL-HLT '18)*. 4171–4186.
- [21] Bhuwan Dhingra, Kathryn Mazaitis, and William W Cohen. 2017. Quasar: Datasets for Question Answering by Search and Reading. *arXiv preprint arXiv:1707.03904* (2017).
- [22] Xin Dong, Evgeniy Gabrilovich, Jeremy Heitz, Wilko Horn, Ni Lao, Kevin Murphy, Thomas Strohmann, Shaohua Sun, and Wei Zhang. 2014. Knowledge Vault: A Web-Scale Approach to Probabilistic Knowledge Fusion. In *Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD '14)*. 601–610.
- [23] Xin Luna Dong, Evgeniy Gabrilovich, Kevin Murphy, Van Dang, Wilko Horn, Camillo Lugaresi, Shaohua Sun, and Wei Zhang. 2015. Knowledge-Based Trust: Estimating the Trustworthiness of Web Sources. *Proc. VLDB Endow.* 8, 9 (May 2015), 938–949.
- [24] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. 2020. An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale. *arXiv preprint arXiv:2010.11929* (2020).
- [25] Matthew Dunn, Levent Sagun, Mike Higgins, V Ugur Güney, Volkan Cirik, and Kyunghyun Cho. 2017. SearchQA: A New Q&A Dataset Augmented with Context from a Search Engine. *arXiv preprint arXiv:1704.05179* (2017).
- [26] William Fedus, Barret Zoph, and Noam Shazeer. 2021. Switch Transformers: Scaling to Trillion Parameter Models with Simple and Efficient Sparsity. *arXiv preprint arXiv:2101.03961* (2021).
- [27] Robert M French. 1999. Catastrophic forgetting in connectionist networks. *Trends in Cognitive Sciences* 3, 4 (1999), 128–135.
- [28] Luyu Gao, Zhu Yun Dai, Tongfei Chen, Zhen Fan, Benjamin Van Durme, and Jamie Callan. 2021. Complement Lexical Retrieval Model with Semantic Residual Embeddings. In *Proceedings of the 43rd European Conference on IR Research (ECIR '21)*. 146–160.
- [29] Ian J Goodfellow, Jonathon Shlens, and Christian Szegedy. 2015. Explaining and Harnessing Adversarial Examples. In *Proceedings of the 3rd International Conference on Learning Representations (ICLR '15)*.
- [30] Alex Graves, Greg Wayne, Malcolm Reynolds, Tim Harley, Ivo Danihelka, Agnieszka Grabska-Barwińska, Sergio Gómez Colmenarejo, Edward Grefenstette, Tiago Ramalho, John Agapiou, Adrià Puigdomènech Badia, Karl Moritz Hermann, Yori Zwols, Georg Ostrovski, Adam Cain, Helen King, Christopher Summerfield, Phil Blunsom, Koray Kavukcuoglu, and Demis Hassabis. 2016. Hybrid computing using a neural network with dynamic external memory. *Nature* 538, 7626 (2016), 471–476.
- [31] Riccardo Guidotti, Anna Monreale, Salvatore Ruggieri, Franco Turini, Fosca Giannotti, and Dino Pedreschi. 2018. A Survey of Methods for Explaining Black Box Models. *Comput. Surveys* 51, 5 (2018), 1–42.
- [32] Jiafeng Guo, Yixing Fan, Qingyao Ai, and W. Bruce Croft. 2016. A Deep Relevance Matching Model for Ad-Hoc Retrieval. In *Proceedings of the 25th ACM International Conference on Information and Knowledge Management (CIKM '16)*. 55–64.
- [33] Hua He, Kevin Gimpel, and Jimmy Lin. 2015. Multi-Perspective Sentence Similarity Modeling with Convolutional Neural Networks. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing (EMNLP '15)*. 1576–1586.
- [34] Kai Hui, Andrew Yates, Klaus Berberich, and Gerard de Melo. 2017. PACRR: A Position-Aware Neural IR Model for Relevance Matching. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing (EMNLP '17)*. 1049–1058.
- [35] Kai Hui, Andrew Yates, Klaus Berberich, and Gerard de Melo. 2018. Co-PACRR: A Context-Aware Neural IR Model for Ad-Hoc Retrieval. In *Proceedings of the 11th ACM International Conference on Web Search and Data Mining (WSDM '18)*. 279–287.
- [36] Ben Hutchinson, Vinodkumar Prabhakaran, Emily Denton, Kellie Webster, Yu Zhong, and Stephen Craig Denuyl. 2020. Social Biases in NLP Models as Barriers for Persons with Disabilities. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics (ACL '20)*. 5491–5501.
- [37] Daphne Ippolito, Daniel Duckworth, Chris Callison-Burch, and Douglas Eck. 2020. Automatic Detection of Generated Text is Easiest when Humans are Fooled. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics (ACL '20)*. 1808–1822.
- [38] Shan Jiang, Simon Baumgartner, Abe Ittycheriah, and Cong Yu. 2020. Factoring Fact-Checks: Structured Information Extraction from Fact-Checking Articles. In *Proceedings of The Web Conference 2020 (WWW '20)*. 1592–1603.
- [39] Mandar Joshi, Eunsol Choi, Daniel S Weld, and Luke Zettlemoyer. 2017. TriviaQA: A Large Scale Distantly Supervised Challenge Dataset for Reading Comprehension. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics, Volume 1 (Long Papers) (ACL '17)*. 1601–1611.
- [40] Vladimir Karpukhin, Barlas Oğuz, Sewon Min, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-tau Yih. 2020. Dense Passage Retrieval for Open-Domain Question Answering. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP '20)*. 6769–6781.
- [41] Omar Khattab and Matei Zaharia. 2020. ColBERT: Efficient and Effective Passage Search via Contextualized Late Interaction over BERT. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '20)*. 39–48.
- [42] Jon M. Kleinberg. 1999. Authoritative Sources in a Hyperlinked Environment. *J. ACM* 46, 5 (Sept. 1999), 604–632.
- [43] Tomáš Kočický, Jonathan Schwarz, Phil Blunsom, Chris Dyer, Karl Moritz Hermann, Gábor Melis, and Edward Grefenstette. 2018. The NarrativeQA Reading Comprehension Challenge. *Transactions of the Association for Computational Linguistics* 6 (2018), 317–328.

- [44] Saar Kuzi, Mingyang Zhang, Cheng Li, Michael Bendersky, and Marc Najork. 2020. Leveraging Semantic and Lexical Matching to Improve the Recall of Document Retrieval Systems: A Hybrid Approach. *arXiv preprint arXiv:2010.01195* (2020).
- [45] Kenton Lee, Ming-Wei Chang, and Kristina Toutanova. 2019. Latent Retrieval for Weakly Supervised Open Domain Question Answering. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics (ACL '19)*. 6086–6096.
- [46] Katherine Lee, Orhan Firat, Ashish Agarwal, Clara Fannjiang, and David Sussillo. 2018. Hallucinations in Neural Machine Translation. In *Interpretability and Robustness in Audio, Speech, and Language Workshop at NeurIPS 2018*.
- [47] Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Ves Stoyanov, and Luke Zettlemoyer. 2020. BART: Denoising Sequence-to-Sequence Pre-training for Natural Language Generation, Translation, and Comprehension. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics (ACL '20)*. 7871–7880.
- [48] Hang Li. 2014. Learning to Rank for Information Retrieval and Natural Language Processing, Second Edition. *Synthesis Lectures on Human Language Technologies* 7, 3 (2014), 1–121.
- [49] Sheng-Chieh Lin, Jheng-Hong Yang, and Jimmy Lin. 2020. Distilling Dense Representations for Ranking using Tightly-Coupled Teachers. *arXiv preprint arXiv:2010.11386* (2020).
- [50] Tie-Yan Liu. 2009. Learning to Rank for Information Retrieval. *Foundations and Trends in Information Retrieval* 3, 3 (March 2009), 225–331.
- [51] Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. RoBERTa: A Robustly Optimized BERT Pretraining Approach. *arXiv preprint arXiv:1907.11692* (2019).
- [52] Bryan McCann, James Bradbury, Caiming Xiong, and Richard Socher. 2017. Learned in Translation: Contextualized Word Vectors. In *Proceedings of the 31st International Conference on Neural Information Processing Systems (NeurIPS '17)*. 6294–6305.
- [53] Bryan McCann, Nitish Shirish Keskar, Caiming Xiong, and Richard Socher. 2018. The Natural Language Decathlon: Multitask Learning as Question Answering. *arXiv preprint arXiv:1806.08730* (2018).
- [54] Gonzalo Mena, David Belanger, Scott Linderman, and Jasper Snoek. 2018. Learning Latent Permutations with Gumbel-Sinkhorn Networks. In *Proceedings of the 6th International Conference on Learning Representations (ICLR '18)*.
- [55] Tomas Mikolov, Kai Chen, Greg S. Corrado, and Jeffrey Dean. 2013. Efficient Estimation of Word Representations in Vector Space. In *International Conference on Learning Representations (ICLR '13)*.
- [56] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Distributed Representations of Words and Phrases and their Compositionality. In *Proceedings of the 27th International Conference on Neural Information Processing Systems (NeurIPS '13)*. 3111–3119.
- [57] Alexander Miller, Adam Fisch, Jesse Dodge, Amir-Hossein Karimi, Antoine Bordes, and Jason Weston. 2016. Key-Value Memory Networks for Directly Reading Documents. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing (EMNLP '16)*. 1400–1409.
- [58] T. Mitchell, W. Cohen, E. Hruschka, P. Talukdar, B. Yang, J. Betteridge, A. Carlson, B. Dalvi, M. Gardner, B. Kisiel, J. Krishnamurthy, N. Lao, K. Mazaitis, T. Mohamed, N. Nakashole, E. Platanios, A. Ritter, M. Samadi, B. Settles, R. Wang, D. Wijaya, A. Gupta, X. Chen, A. Saparov, M. Greaves, and J. Welling. 2018. Never-Ending Learning. *Commun. ACM* 61, 5 (April 2018), 103–115.
- [59] Bhaskar Mitra and Nick Craswell. 2018. An Introduction to Neural Information Retrieval. *Foundations and Trends in Information Retrieval* 13, 1 (December 2018), 1–126.
- [60] Bhaskar Mitra, Fernando Diaz, and Nick Craswell. 2017. Learning to Match Using Local and Distributed Representations of Text for Web Search. In *Proceedings of the 26th International Conference on World Wide Web (WWW '17)*. 1291–1299.
- [61] Jian-Yun Nie. 2010. *Cross-Language Information Retrieval*. Morgan & Claypool Publishers.
- [62] Kezban Dilek Onal, Ye Zhang, Ismail Sengör Altıngövd, Md. Mustafizur Rahman, P. Senkul, Alex Braylan, Brandon Dang, H. Chang, Henna Kim, Quinten McNamara, A. Angert, E. Banner, Vivek Khetan, Tyler McDonnell, A. T. Nguyen, D. Xu, Byron C. Wallace, M. Rijke, and Matthew Lease. 2017. Neural information retrieval: at the end of the early years. *Information Retrieval Journal* 21 (2017), 111–182.
- [63] Ankur P Parikh, Oscar Täckström, Dipanjan Das, and Jakob Uszkoreit. 2016. A Decomposable Attention Model for Natural Language Inference. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing (EMNLP '16)*. 2249–2255.
- [64] German I Parisi, Ronald Kemker, Jose L Part, Christopher Kanan, and Stefan Wermter. 2019. Continual lifelong learning with neural networks: A review. *Neural Networks* 113 (2019), 54–71.
- [65] Jeffrey Pennington, Richard Socher, and Christopher D Manning. 2014. GloVe: Global Vectors for Word Representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP '14)*. 1532–1543.
- [66] Matthew E Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep Contextualized Word Representations. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers) (NAACL-HLT '2018)*. 2227–2237.
- [67] Telmo Pires, Eva Schlinger, and Dan Garrette. 2019. How Multilingual is Multilingual BERT?. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics (ACL '19)*. 4996–5001.
- [68] Richard Qian. 2013. Understand Your World with Bing. <https://blogs.bing.com/search/2013/03/21/understand-your-world-with-bing>
- [69] Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language Models are Unsupervised Multitask Learners. (2019).
- [70] Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. 2020. Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer. *Journal of Machine Learning Research* 21, 140 (2020), 1–67.
- [71] Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016. SQuAD: 100,000+ Questions for Machine Comprehension of Text. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing (EMNLP '16)*. 2383–2392.
- [72] Qiu Ran, Yankai Lin, Peng Li, Jie Zhou, and Zhiyuan Liu. 2019. NumNet: Machine Reading Comprehension with Numerical Reasoning. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP '19)*. 2474–2484.
- [73] Jinfeng Rao, Linqing Liu, Yi Tay, Wei Yang, Peng Shi, and Jimmy Lin. 2019. Bridging the Gap between Relevance Matching and Semantic Matching for Short Text Similarity Modeling. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP '19)*. 5370–5381.
- [74] Abigail See, Peter J Liu, and Christopher D Manning. 2017. Get To The Point: Summarization with Pointer-Generator Networks. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics, Volume 1 (Long Papers) (ACL '17)*. 1073–1083.
- [75] Aliaksei Severyn and Alessandro Moschitti. 2015. Learning to Rank Short Text Pairs with Convolutional Deep Neural Networks. In *Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '15)*. 373–382.
- [76] Emily Sheng, Kai-Wei Chang, Premkumar Natarajan, and Nanyun Peng. 2019. The Woman Worked as a Babysitter: On Biases in Language Generation. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP '19)*. 3407–3412.
- [77] Amit Singhal. 2012. Introducing the Knowledge Graph: things, not strings. <https://blog.google/products/search/introducing-knowledge-graph-things-not/>
- [78] Fabian M. Suchanek, Gjergji Kasneci, and Gerhard Weikum. 2007. Yago: A Core of Semantic Knowledge. In *Proceedings of the 16th International Conference on World Wide Web (WWW '07)*. 697–706.
- [79] Mukund Sundararajan, Ankur Taly, and Qiqi Yan. 2017. Axiomatic Attribution for Deep Networks. In *Proceedings of the 34th International Conference on Machine Learning (ICML '17)*. 3319–3328.
- [80] Chuanqi Tan, Furu Wei, Nan Yang, Bowen Du, Weifeng Lv, and Ming Zhou. 2018. S-Net: From Answer Extraction to Answer Generation for Machine Reading Comprehension. In *Proceedings of the 32nd AAAI Conference on Artificial Intelligence (AAAI '18)*. 5940–5947.
- [81] Ming Tan, Cicero dos Santos, Bing Xiang, and Bowen Zhou. 2015. LSTM-based Deep Learning Models for Non-factoid Answer Selection. *arXiv preprint arXiv:1511.04108* (2015).
- [82] Yi Tay, Mostafa Dehghani, Samira Abnar, Yikang Shen, Dara Bahri, Philip Pham, Jinfeng Rao, Liu Yang, Sebastian Ruder, and Donald Metzler. 2020. Long Range Arena: A Benchmark for Efficient Transformers. *arXiv preprint arXiv:2011.04006* (2020).
- [83] Yi Tay, Mostafa Dehghani, Dara Bahri, and Donald Metzler. 2020. Efficient Transformers: A Survey. *arXiv preprint arXiv:2009.06732* (2020).
- [84] Yi Tay, Luu Anh Tuan, and Siu Cheung Hui. 2018. Compare, Compress and Propagate: Enhancing Neural Architectures with Alignment Factorization for Natural Language Inference. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing (EMNLP '18)*. 1565–1575.
- [85] Yi Tay, Luu Anh Tuan, and Siu Cheung Hui. 2018. Multi-Cast Attention Networks. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining (KDD '18)*. 2299–2308.
- [86] Yi Tay, Luu Anh Tuan, Siu Cheung Hui, and Jian Su. 2018. Densely Connected Attention Propagation for Reading Comprehension. In *Proceedings of the 32nd Conference on Neural Information Processing Systems (NeurIPS '18)*. 4911–4922.
- [87] Yi Tay, Shuohang Wang, Luu Anh Tuan, Jie Fu, Minh C Phan, Xingdi Yuan, Jinfeng Rao, Siu Cheung Hui, and Aston Zhang. 2019. Simple and Effective Curriculum Pointer-Generator Networks for Reading Comprehension over Long Narratives. In *Proceedings of the 57th Annual Meeting of the Association for*

- Computational Linguistic (ACL '19)*. 4922–4931.
- [88] Adam Trischler, Tong Wang, Xingdi Yuan, Justin Harris, Alessandro Sordoni, Philip Bachman, and Kaheer Suleman. 2017. NewsQA: A Machine Comprehension Dataset. In *Proceedings of the 2nd Workshop on Representation Learning for NLP (RepL4NLP '17)*. 191–200.
 - [89] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention Is All You Need. In *Proceedings of the 31st Conference on Neural Information Processing Systems (NeurIPS '17)*. 5998–6008.
 - [90] Oriol Vinyals, Meire Fortunato, and Navdeep Jaitly. 2015. Pointer Networks. In *Proceedings of the 29th International Conference on Neural Information Processing Systems (NeurIPS '15)*. 2692–2700.
 - [91] Denny Vrandečić and Markus Krötzsch. 2014. Wikidata: A Free Collaborative Knowledgebase. *Commun. ACM* 57, 10 (Sept. 2014), 78–85.
 - [92] Mengqiu Wang, Noah A Smith, and Teruko Mitamura. 2007. What is the Jeopardy Model? A Quasi-Synchronous Grammar for QA. In *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL '07)*. 22–32.
 - [93] Shuohang Wang and Jing Jiang. 2017. A Compare-Aggregate Model for Matching Text Sequences. In *Proceedings of the 5th International Conference on Learning Representations (ICLR '17)*.
 - [94] Shuohang Wang and Jing Jiang. 2017. Machine Comprehension Using Match-LSTM and Answer Pointer. In *Proceedings of the 5th International Conference on Learning Representations (ICLR '17)*.
 - [95] Shuohang Wang, Mo Yu, Xiaoxiao Guo, Zhiguo Wang, Tim Klinger, Wei Zhang, Shiyu Chang, Gerry Tesauro, Bowen Zhou, and Jing Jiang. 2018. R³: Reinforced ranker-reader for open-domain question answering. In *Proceedings of the 32nd AAAI Conference on Artificial Intelligence (AAAI '18)*. 5981–5988.
 - [96] Tongzhou Wang, Jun-Yan Zhu, Antonio Torralba, and Alexei A Efros. 2018. Dataset Distillation. *arXiv preprint arXiv:1811.10959* (2018).
 - [97] Wenhui Wang, Nan Yang, Furu Wei, Baobao Chang, and Ming Zhou. 2017. Gated Self-Matching Networks for Reading Comprehension and Question Answering. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics, Volume 1 (Long Papers) (ACL '17)*. 189–198.
 - [98] Kellie Webster, Xuezhong Wang, Ian Tenney, Alex Beutel, Emily Pitler, Ellie Pavlick, Jilin Chen, Ed Chi, and Slav Petrov. 2019. Measuring and Reducing Gendered Correlations in Pre-trained Models. *arXiv preprint arXiv:2010.06032* (2019).
 - [99] Jason Weston, Sumit Chopra, and Antoine Bordes. 2015. Memory Networks. In *Proceedings of the 3rd International Conference on Learning Representations (ICLR '15)*.
 - [100] Ellery Wulczyn, Nithum Thain, and Lucas Dixon. 2017. Ex Machina: Personal Attacks Seen at Scale. In *Proceedings of the 26th International Conference on World Wide Web (WWW '17)*. 1391–1399.
 - [101] Chenyan Xiong, Zhuyun Dai, Jamie Callan, Zhiyuan Liu, and Russell Power. 2017. End-to-End Neural Ad-Hoc Ranking with Kernel Pooling. In *Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '17)*. 55–64.
 - [102] Lee Xiong, Chenyan Xiong, Ye Li, Kwok-Fung Tang, Jialin Liu, Paul Bennett, Junaid Ahmed, and Arnold Overwijk. 2021. Approximate Nearest Neighbor Negative Contrastive Learning for Dense Text Retrieval. In *Proceedings of the 9th International Conference on Learning Representations (ICLR '21)*.
 - [103] Linting Xue, Noah Constant, Adam Roberts, Mihir Kale, Rami Al-Rfou, Aditya Siddhant, Aditya Barua, and Colin Raffel. 2020. mT5: A Massively Multilingual Pre-trained Text-to-Text Transformer. *arXiv preprint arXiv:2010.11934* (2020).
 - [104] Yi Yang, Wen-tau Yih, and Christopher Meek. 2015. WikiQA: A Challenge Dataset for Open-Domain Question Answering. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing (EMNLP '15)*. 2013–2018.
 - [105] Adams Wei Yu, David Dohan, Minh-Thang Luong, Rui Zhao, Kai Chen, Mohammad Norouzi, and Quoc V Le. 2018. QANet: Combining Local Convolution with Global Self-Attention for Reading Comprehension. In *Proceedings of the 6th International Conference on Learning Representations (ICLR '18)*.
 - [106] Bo Zhao, Konda Reddy Mopuri, and Hakan Bilen. 2021. Dataset Condensation with Gradient Matching. In *Proceedings of the 9th International Conference on Learning Representations (ICLR '21)*.
 - [107] Chen Zhao, Chenyan Xiong, Corby Rosset, Xia Song, Paul Bennett, and Saurabh Tiwary. 2020. Transformer-XH: Multi-Evidence Reasoning with eXtra Hop Attention. In *Proceedings of the 8th International Conference on Learning Representations (ICLR '20)*.