CODEC: Complex Document and Entity Collection

Iain Mackie, Paul Owoicho, Carlos Gemmell
University of Glasgow
Glasgow, Scotland, UK
{i.mackie.1,p.owoicho.1,c.gemmell.1}@research.gla.ac.uk

Sophie Fischer, Sean MacAvaney, Jeffrey Dalton
University of Glasgow
Glasgow, Scotland, UK
{sophie.fischer,sean.macavaney,jeff.dalton}@glasgow.ac.uk

ABSTRACT

CODEC is a document and entity ranking benchmark that focuses on complex research topics. We target essay-style information needs of social science researchers, i.e. 'How has the UK's Open Banking Regulation benefited Challenger Banks?'. CODEC includes 42 topics developed by researchers and a new focused web corpus with semantic annotations including entity links. This resource includes expert judgments on 17,509 documents and entities (416.9 per topic) from diverse automatic and interactive manual runs. The manual runs include 387 query reformulations, providing data for query performance prediction and automatic rewriting evaluation.

CODEC includes analysis of state-of-the-art systems, including dense retrieval and neural re-ranking. The results show the topics are challenging with headroom for document and entity ranking improvement. Query expansion with entity information shows significant gains on document ranking, demonstrating the resource's value for evaluating and improving entity-oriented search. We also show that the manual query reformulations significantly improve document ranking and entity ranking performance. Overall, CODEC provides challenging research topics to support the development and evaluation of new entity-centric search methods.

CCS CONCEPTS

• **Information systems** → **Information retrieval**.

KEYWORDS

Document Ranking; Entity Retrieval; Query Reformulation

1 INTRODUCTION

Researchers spend considerable time exploring sources to understand key arguments, concepts and facts about a specific topic. For example, surveys show that many legal researchers, recruitment professionals, and healthcare researchers require high-recall Boolean or structured queries over domain-specific collections [33]. In contrast, CODEC focuses on researchers within social sciences (history, economics, and politics) to develop complex topics that can be satisfied by web documents and entities within standard knowledge bases (i.e. Wikipedia).

Figure 1 shows an example topic, where a financial researcher wants to understand *How has the UK's Open Banking Regulation benefited Challenger Banks?*. The researcher would use a standard commercial search engine to identify valuable information, reformulating queries to investigate varying dimensions of the topic, e.g. 'Open Banking products', 'Challenger Banks fundraising', etc. Through this iterative process, the researcher is trying to identify useful sources (i.e. documents) and understand the critical concepts (i.e. entities). CODEC[1] is a dataset that seeks to benchmark document and entity ranking on complex long-form essay questions.

Domain	Economics
Query	How has the UK's Open Banking Regulation benefited challenger banks?
Narrative	UK's Open Banking regulation, which has parallels to the EU's second payment service directive (PSD2), went live in January 2018. This piece of legislation "will require banks to open their payments infrastructure and customer data assets to third parties". As a result, banks no longer have a monopoly on user data if clients grant permission. Challenger banks are small, recently created retail banks that compete directly with the longer-established banks in the UK. Specifically, seeking market share from the "big four" UK retail banks (Barclays, HSBC, Lloyds Banking Group, and NatWest Group). The banks distinguish themselves from the historic banks by modern financial technology practices, such as online-only operations, that avoid the costs and complexities of traditional banking. The largest UK-operating challenger banks include Atom Bank, Revolut, Starling, N26, and Tide. Relevant documents and entities will discuss how challenger banks have used open banking to develop new products or capture market share from traditional retail banks in the UK.

Figure 1: Example CODEC topic: economics-1.

Studies show that a large proportion of information needs are about entities or relate to entities [14, 18]. This is particularly true for essay-style questions across social sciences, where the information need generally focuses on key events (e.g. *How close did the world come to nuclear war during the Cuban Missile Crisis?*), people (e.g. *How did Colin Kaepernick impact the political discourse about racism in the United States?*), or things (e.g. *What technological challenges does Bitcoin face to becoming a widely used currency?*). Previous work demonstrates that incorporating entity-based information improves ad-hoc retrieval, with the most notable improvements made on complex topics [10].

COmplex Document and Entity Collection (CODEC) supports two distinct tasks: document ranking and entity ranking. Document ranking is the task, given an information need Q, to return a relevance-ranked list of documents $[D_1, ..., D_N]$ from a document corpus C_D. Entity ranking is the task, given an information need

[1] available at https://github.com/grill-lab/CODEC and *ir-datasets* [25]

SIGIR '22, July 11–15, 2022, Madrid, Spain

ACM ISBN 978-1-4503-8732-3/22/07…$15.00
https://doi.org/10.1145/3477495.3531712

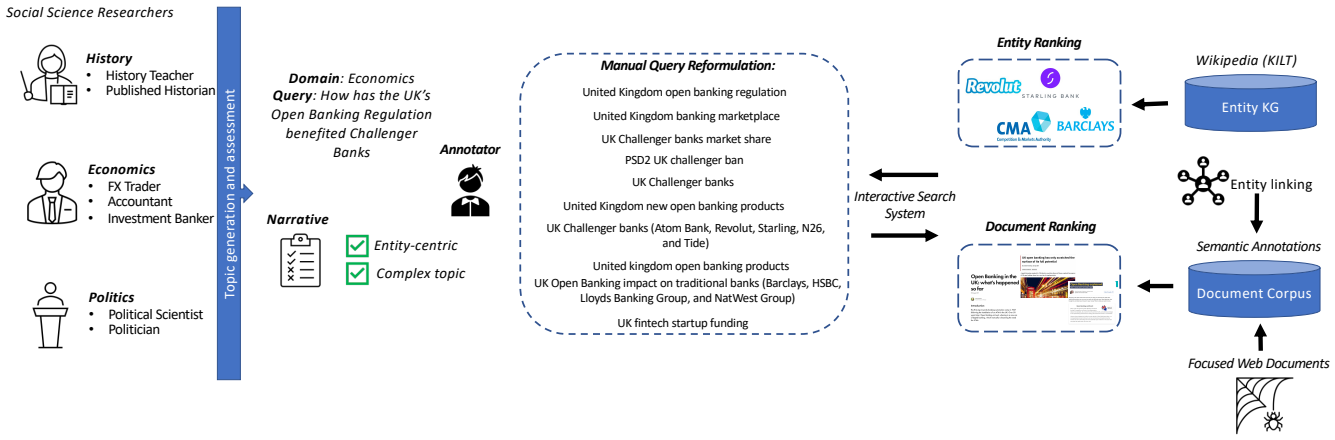

Figure 2: Overview of CODEC. Social science researchers develop topics based on new complex topic criteria. Annotators assess initial pooled runs before using an interactive search system to issue manual queries and explore the topic. Documents are from CODEC's focused web document collection and entities from KILT's [30] version of Wikipedia.

Q, to return a relevance-ranked list of entities $[E_1, .., E_N]$ from an entity knowledge base KB_E. Entity links between the documents and entities provide structured connections between both tasks. We also release the manual query reformulations from the researcher annotating the topic with mapped relevance judgments. Figure 2 shows the setup of these tasks. Although tasks can be undertaken independently, CODEC has aligned document and entity judgments to allow joint exploration of these tasks. This setup will allow researchers to leverage CODEC to target two key limitations of current neural ranking models:

(1) Entity-centric representations: Prior work has shown that ad-hoc retrieval [10, 40] and entity ranking [12] can be improved by leveraging entity information. Yet, current state-of-the-art methods rely solely on text representations and lack the understanding of entities and their relationships. For example, dense retrieval underperforms BM25 on even simple entity queries [34]. CODEC provides aligned document and entity judgments that are unified through entity-linking connections. For example, we demonstrate simple entity-based query expansion statistically improves MAP and Recall@1000 for document retrieval over strong initial retrieval systems. CODEC enables entity-centric ranking models to be developed on complex topics.

(2) Complex topics: Compared to easier general factoid or short keyword queries [16], CODEC curates long natural-language queries where relevant topic information spans many entities and documents. Figure 1 shows an example topic query and narrative. CODEC queries average 12.5 words in length, considerably longer than many datasets, i.e. TREC Deep Learning [8] averages 5.8 words. We also deliberately do not release shorter keyword 'title' queries, such as TREC CORE [3] and TREC Robust [37], to challenge end-to-end ranking on long, natural language queries.

We develop new complex topic criteria to produce topics that require deep knowledge and research to identify the key facts and arguments. Using topic narratives as a proxy for topic complexity, these contain on average 143.4 words and 23.7 explicitly mentioned entities. Annotators also required, on average, 9.2 manual query reformulations to reasonably understand these topics.

These query reformulations are released to allow development of query expansion and query reformulation on complex topics. We show that query expansion using the query reformulations significantly improves MAP and Recall@1000 on document ranking and MAP, NDCG@10 and Recall@1000 on entity ranking. Additionally, we find the best manual reformulation performs better than the original query.

Figure 2 shows the CODEC dataset creation process that is designed for entity-centric ranking over complex topics. Social science experts across history (history teacher, history scholar), economics (trader, accountant, investment banker) and politics (political scientist, politician) generate 42 topics based on the developed criteria.

Experts also suggest 24 focused research websites for the corpus. Topics across history, economics and politics are selected because (1) this allows a broad range of complex topics and (2) there is sufficient topic overlap to share a single web document corpus. After reviewing several standard document collections, none had enough focused content for these topics, with history topics having poorest coverage. Thus, web content from Common Crawl is used with website-specific parsers to create a curated document corpus of around 700k heterogeneous web documents (blogs, news, interviews, etc.). Wikipedia via KILT [30] is the entity knowledge base (KB) for entity ranking. Entity linking is run over the document corpus using REL [36], which is an effective end-to-end entity linker.

Experienced IR experts (the authors) produce 6,186 document (147.3 per topic) and 11,323 entity (269.6 per topic) relevance judgments. We use a two-stage assessment process: (1) assessing pooled runs of BM25 [32], BM25 with RM3 expansion [1], ANCE [41], MonoT5 [29], commercial search engine, and entity linkers [5, 39]. Then (2) allowing assessors to formulate a series of manual queries (average 9.2 queries per topic) to search and annotate different aspects of the topics. This assessment process is intended to simulate

the research process of complex topics on document and entity ranking.

We evaluate strong document and entity ranking systems using CODEC, including sparse retrieval, dense retrieval, and neural Language Model (LM) re-ranking. The results show significant headroom for improvement within the current first-pass and re-ranking search systems on these complex topics. The best performing document ranking system has MAP under 0.35 and NDCG under 0.5, while the best-performing entity ranking system has MAP under 0.25 and NDCG under 0.45. Comparing these measures to comparable systems on TREC Deep Learning [8], where NDCG@10 is 0.7 and MAP is 0.55, highlights that CODEC complex topics are challenging for current systems.

Analysis shows a positive correlation between document relevance and the occurrence of the most relevant entities. These findings support the complementary relationship between the document and entity ranking tasks and motivate the development of future models that leverage both [10, 12]. We explore this directly by leveraging CODEC entity judgments within document ranking via query expansion, which improves document ranking by a statistically significant amount.

CODEC is a valuable resource for IR researchers, supporting the development of new neural methods that leverage entity-centric representations. The key contributions of CODEC are:

- **Test collection:** We release a test collection to benchmark complex research topics on document and entity ranking. We produce new guidelines for complex topics and have social sciences experts generate 42 new topics. We curate a new document corpus from 24 focused sources covering diverse social science domains across history, economics, and politics. We release 6,186 document and 11,323 entity judgments.
- **Analysis of system performance:** We study the behaviour of strong systems (sparse retrieval, dense retrieval, and LM re-ranking) that highlights failures and provide headroom for improvement on complex topics. We highlight specific failures due to models lacking the ability to encode or utilize entities and their relationships.
- **Aligned entity and document tasks:** We design the dataset to have aligned document and entity judgments to allow the development of new entity-centric ranking models. We show that document relevance is positively correlated with the proportion of most relevant entities contained within the document. We show that we can improve document ranking using entity query expansion.
- **Query reformulation:** A two-stage assessment process allows the assessment of strong pooled runs, followed by manual exploration of the topic using multiple live search systems. These query reformulations (387 queries in total) are also released. We show that the best query reformulation outperforms the original query. Additionally, query expansion that uses all query reformulations improves document and entity ranking.

2 RELATED WORK

We provide an overview of the related literature across document ranking, entity ranking, and query reformulation.

2.1 Document ranking

Document ranking is the task of retrieving a ranked list of relevant documents from a corpus given a specified information need. In recent years, pre-trained language models [20, 24, 29] and dense retrieval systems [17, 41] have been shown to improve document ranking. However, findings have shown failures of neural retrieval on even simple entity queries [34].

Within the field of domain-specific research (i.e. legal, recruitment, and healthcare), high-recall Boolean or structured queries over domain-specific collections are standard [33]. CODEC instead focuses on social sciences (history, economics, and politics) and open-ended essay-style topics that more closely align with web or newswire search. For example, TREC Robust [37] and TREC CORE [3], provide a similar style of natural language queries. However, CODEC topics are more current and provide extensive narratives, i.e. CODEC narratives average 143.4 words versus TREC CORE's 44.0.

MS MARCO [28] is a family of passage and document test collections consisting of web queries, passages or documents, with sparse relevance judgments. However, MS MARCO's annotation technique means that the queries tend to be artificially easy and exhibit undesirable qualities like the 'maximum passage bias' [20]. TREC Deep Learning [8] extends MS MARCO with dense judgments to provide a more useful benchmark, and DL-HARD [26] develops a more challenging subset with annotations and metadata. CODEC differs from these datasets in terms of length of queries, i.e. TREC Deep Learning averages 5.8 words compared to CODEC's 12.5 words. CODEC provides a new focused document corpus that provides good coverage of complex social science topics. Additionally, aligned entity and document judgments will allow researchers to explore the related task of entity ranking.

2.2 Entity ranking

Entity ranking is the task of retrieving a ranked list of relevant entities from an entity knowledge base given a specified information need. Past studies have shown that entity ranking improves by leveraging mentions in text passages to create a topic-specific text-entity graph [12]. Transformer-based embeddings have been shown to be a reasonable entity ranking baseline [6], with a strong performance on the related tasks such as entity linking [38]. Entity ranking closely relates to entity aspect linking, where the task is to identify the fine-grained semantics of the entity that relates to a mention in a contextual passage [27, 31]. Incorporation of entity aspects has also been shown to improve entity ranking [6].

INEX 2009 XML Entity Ranking Track [11] focuses on entity ranking and entity list completion from Wikipedia XML documents. The queries are generally factoid in nature, i.e. 'Italian Nobel prize winners' and 'Formula 1 drivers that won the Monaco Grand Prix'. In contrast, CODEC asks entity ranking systems to rank important named entities or general concepts on complex topics, 'What technological challenges does Bitcoin face to becoming a widely used currency?' Where relevant entities include [Cyberattack] and [Transaction Cost], as well as the explicitly mentioned [Bitcoin].

DBpedia-Entity [4] and DBpedia-Entity v2 [15] are test collections for entity search over the structured DBpedia knowledge base. These encompass four slightly different entity search tasks, i.e. named entity search, ad-hoc entity ranking, list completion, and natural language entity-based QA. The key difference is CODEC uses a free-text based entity KG (Wikipedia), the topics are more open-ended, and there is an aligned document ranking task.

TREC CAR [13] is a passage and free-text entity ranking dataset built from Wikipedia and uses Wikipedia titles and headings as keyword queries. TREC CAR is the closest setup to CODEC, and the sparse entity and document relevance judgements could provide a useful pre-training step. CODEC differs based on the complex natural language queries, heterogeneous text corpus (instead of solely Wikipedia), and focus on document ranking.

2.3 Query Reformulation

Culpepper et al. [9] highlight that users routinely reformulate queries to satisfy an information need, and show the high variance of retrieval performance across these query variants. In fact, Culpepper et al. [9] find that query formulations have a comparable effect to the actual topic complexity in terms of overall system performance. CODEC provides over 387 manual query reformulations to support research in this space, i.e. query performance prediction and automatic query reformulation.

Liu et al. [22] show the benefits for human and automatic query reformulations on document ranking systems, with human reformulations being the most effective. Analysis on CODEC supports these findings that the best reformulations improve document and entity ranking. ORCAS is a click dataset that aligns with TREC Deep Learning [7], which is useful for identifying clusters of related queries or related documents. However, CODEC provides manual query reformulation on a single information need, providing more fine-grained topic-specific query reformulations.

3 CODEC

CODEC is a test collection that provides two tasks: document ranking and entity ranking. A complex topic is defined as an essay-style question where essential information can span across multiple entities and documents (see Section 3.2 for more detail).

This dataset benchmarks a researcher who is attempting to find supporting entities and documents that will form the basis of a long-form essay discussing the topic from various perspectives. The researcher would explore the topic to (1) identify relevant sources and (2) understand key concepts.

3.1 Task Definition

CODEC supports both document ranking and entity ranking tasks. Document ranking systems have to return a relevance-ranked list of documents $[D_1, ..., D_N]$, from a document corpus C_D, for a given natural language query Q. Entity ranking systems havef to return a relevance-ranked list of entities $[E_1, ..., E_N]$, from an entity knowledge base KB_E, for a given natural language query Q. Document ranking uses CODEC's new document corpus and entity ranking uses KILT as the entity KB. For the experimental setup, we provide four pre-defined 'standard' folds for k-fold cross-validation to allow parameter tuning. Initial retrieval or re-ranking of provided baseline runs can both be evaluated using this test collection. CODEC setup encourages exploration of the related entity and document ranking tasks; however, both tasks can also be undertaken in isolation.

Table 1: Topic Statistics across 42 CODEC topics.

	Total	Avg. Length
Query (Words)	524	12.5
Query (Entities)	102	2.4
Narrative (Words)	6,021	143.4
Narrative (Entities)	994	23.7

3.2 Topic Generation

CODEC provides complex topics which intend to benchmark the role of a researcher. Understanding these topics requires deep knowledge and investigation to identify the key documents and entities.

Social science experts from history (history teacher, published history scholar), economics (FX trader, accountant, investment banker) and politics (political scientists, politician) helped to generate interesting and factually-grounded topics. The authors develop the following criteria for complex topics:

- **Open-ended essay-style**: Satisfying the information need of this topic comprehensively would require a long-form essay-style response. Factoid questions or questions that only require a short answer are not suitable.
- **Natural language question**: The query should be long, natural language-based. Keyword queries are not suitable.
- **Multiple points of view**: It is preferable if the topic elicits debate and multiple points of view. A good response would thus require an understanding of each of these dimensions.
- **Concern multiple key entities**: It is preferable if the topic concerns multiple key entities (people, things, events, etc.).
- **Complexity**: It is preferable if the topic requires an educated adult to undertake significant research to understand it.
- **Knowledge**: It is preferable if the topic requires deep knowledge to understand satisfactorily.

The domain experts write 42 topics with minimal post-processing from the authors to align styles or correct spelling or grammatical errors. There is an equal number of topics per target domain, i.e. 14 history topics, 14 economics topics, and 14 politics topics. Each topic contains a query and narrative. The query is the question the researcher seeks to understand by exploring documents and entities, i.e., the text input posed to the search system. The narratives provide an overview of the topic (key concepts, arguments, facts, etc.) and allow non-domain experts to understand the topic. Due to the complexity of these topics, the narratives are not completely comprehensive but provide a useful starting point for annotators. We also review pooled runs to assess whether topics are too easy (i.e. lots of highly ranked relevant documents) or do not align with the corpora (i.e. not enough relevant documents or entities to satisfy the information need).

Table 1 shows the average number of words and entities in topic queries and narratives. Entity statistics are calculated by running GENRE [5] over the queries and narratives. An average of 12.5 words and 2.4 entities per query supports long natural language

Table 2: Distribution of Top 15 Websites in Document Corpus.

	Count
reuters.com	172,127
forbes.com	147,399
cnbc.com	100,842
britannica.com	93,484
latimes.com	88,486
usatoday.com	31,803
investopedia.com	21,459
bbc.co.uk	21,414
history.state.gov	9,187
brookings.edu	9,058
ehistory.osu.edu	8,805
history.com	6,749
spartacus-educational.com	3,904
historynet.com	3,811
historyhit.com	3,173
TOTAL	**729,824**

queries that include entities. Narratives provide a good proxy for the complexity of the underlying information need, and 143.4 words and 23.7 entities support this complexity.

3.3 Document Corpus

We want CODEC document corpus to have enough high-quality coverage of current social science topics.

3.3.1 Focused Content. We perform initial exploration on standard document collections (MS MARCO, TREC Washington Post, etc.) with CODEC topics but find critical coverage gaps within required research content. History topics have particularly low coverage and would require augmentation from historical authority sites. This motivates building upon a subset of Common Crawl to create a new focused document corpus for the target domains. Table 2 shows the distribution of documents.

We leverage our domain experts, who recommend suitable seed websites or sections of websites. The pool contains a mixture of clearly specialized websites (i.e. economicsdiscussion.net, history.com, brookings.edu) and several general newswire websites (bbc.co.uk, latimes.com, etc.). Social science experts requested up-to-date newswire websites for contextualizing current economic and political topics. We also run the topics through a commercial search engine to ensure appropriate coverage and that each domain has enough representation.

3.3.2 Corpus Generation Pipeline. The document corpus pipeline takes the focused seed websites and uses Common Crawl and URL pattern matching to extract 300GB of HTML across recent crawls in 2021 (CC-MAIN-2021-[21,17,10,14]). We develop 24 custom BeautifulSoup HTML parsers to extract text and metadata while removing any advertising and formatting. This creates documents with fields:

- **id**: Unique identifier is the MD5 hash of URL.
- **url**: Location of the webpage (URL).
- **title**: Title of the webpage if available.

Table 3: Entity Links on Document Corpus.

	Corpus Total	Document Mean
Entity Links	27,482,650	37.7
Entity Candidates	144,127,482	197.5

- **contents**: The text content of the webpage after removing any unnecessary advertising or formatting. New lines provide some structure between the extracted sections of the webpage, while still easy for neural systems to process.

We then run multiple filtering stages to ensure the documents are of suitable length and unique. First, the extracted text has to contain at least 30 words, approximately a paragraph. Second, we identify that several websites contain the same (or very similar) webpage hosted on different URLs. Thus, we run a de-duplication step by grouping webpages from the same website that (1) have the same *title* and (2) cosine similarity between document tokens greater than 95%. We solely include the document with the longest *contents* in the final corpus. This removes 96,900 duplicates and results in a final corpus containing 729,824 documents. The corpus is released in jsonlines format.

3.3.3 Entity Linking. We run the REL [36] entity linker over the entire 729,824 document corpus to provide structured connections between documents and entities. REL is a light-weight neural entity linker that allows easy deployment and strong performance. We use the suggested setup for mention detection, i.e. Flair [2] which is a Named Entity Recognition (NER) model based on contextualized word embeddings. We use REL's pre-trained model for candidate selection that uses a 2019-07 version of Wikipedia (i.e. closely aligns with the 2019/08/01 Wikipedia version for entity KB). For each document we provide a list of entity links containing fields:

- **mention**: Text spans in document that is linked to entity.
- **prediction**: Top predicted entity link (Wikipedia title).
- **prediction_kilt**: We map *prediction* entity link to KILT id to align with entity KB and entity judgments.
- **candidates**: Top-k entity link candidates (Wikipedia title).
- **candidates_kilt**: We map *candidates* entity links to KILT ids to align with entity KB and entity judgments.
- **conf_ed**: Score of Flair NER model.
- **score**: Scores of REL candidate selection model.

We release the full 18GB of entity links in jsonlines format. This will allow researchers to use entity links within document and entity ranking easily. Table 3 shows breakdown of the 27.5m entity links (37.7/document) and 144.1m entity candidates (197.5/document).

3.4 Entity KB

CODEC uses KILT's [30] Wikipedia KB for the entity ranking task, which is based on the 2019/08/01 Wikipedia snapshot. KILT contains 5.9M preprocessed articles which are freely available to use. The entity pages are primarily text-based with minimal structure to indicate headings or passages, i.e. very similar to Document Corpus. KILT is selected for CODEC's KB because it aligns with related knowledge-grounded tasks (fact-checking, open-domain QA, entity linking, etc.). KILT also provides inter-entity entity links based on

Wikipedia mentions, which could be helpful when identifying how related entities are to each other.

3.5 Relevance Criteria

We perform relevance assessment on a graded scale (between 0 and 3) using developed guidelines to ensure a consistent assessment process. Guidelines take inspiration from those of HC4 [19] and are adapted for our tasks (full guidelines online).

3.5.1 Document Criteria. The key question for document relevance is: *How valuable is the most important information in this document?*

- **Very Valuable (3):** The most valuable information in the document would be found in the lead paragraph of a report written on the topic. This includes central topic-specific arguments, evidence, or knowledge. This does not include general definitions or background.
- **Somewhat valuable (2):** The most valuable information in the document would be found in the body of such a report. This includes valuable topic-specific arguments, evidence, or knowledge.
- **Not Valuable (1):** Although on topic, the information contained in the document might only be included in a report footnote or omitted entirely. This consists of definitions or background information.
- **Not Relevant (0):** Not useful or on topic.

3.5.2 Entity Criteria. The key question for entity relevance is: *How valuable is understanding this entity to contextualize document knowledge?*

- **Very Valuable (3):** This entity would be found in the lead paragraph of a report written on the topic. It is absolutely critical to understand what this entity is for understanding this topic.
- **Somewhat valuable (2):** The entity would be found in the body of such a report. It is important to understand what this entity is for understanding this topic.
- **Not Valuable (1):** Although on topic, this entity might only be included in a report footnote or omitted entirely. It is useful to understand what this entity is for understanding this topic.
- **Not Relevant (0):** This entity is not useful or on topic.

3.6 Assessment Process

CODEC uses a 2-stage assessment approach to balance adequate coverage of current systems while allowing annotators to explore topics using iterative search systems.

3.6.1 Initial Run Assessment. We generate pools from runs using state-of-the-art sparse and dense retrieval methods. For document runs we use top-100 BM25 [32], BM25 using RM3 expansion [1], ANCE [41], BM25 re-ranked with MonoT5 [29], BM25 using RM3 expansion re-ranked with MonoT5, and ANCE re-ranked by MonoT5. We also use a commercial search engine where the top-100 search results are limited to the 24 corpus websites, and the URLs are mapped back to document ids. Pyserini [21] is used for BM25 and BM25 with RM3 expansion with default parameters. We use MS Marco fine-tuned versions of ANCE and MonoT5.

Table 4: Judgment distribution across 42 topics.

Judgment	Document Ranking	Entity Ranking
0	2,353	7,053
1	2,210	2,241
2	1,207	1,252
3	416	777
TOTAL	**6,186**	**11,323**

For entity runs, we also use a pool of the top-100 results from BM25, BM25 using RM3 expansion, ANCE, BM25 re-ranked with MonoT5, BM25 with RM3 re-ranked with MonoT5, and ANCE re-ranked with MonoT5. We use ELQ [39], which is an end-to-end entity linking model for questions, to produce an entity run on the queries. GENRE [5], sequence-to-sequence entity linking model, is used to produce an entity run using the narrative. We again use a commercial search engine where top-100 search results are limited to Wikipedia and URLs mapped back to document ids.

We devise a weighting ratio for document and entity pooling based on an analysis of several topics across domains. This process takes (1) top-k for each initial system run, then (2) intersection across specified sub-groups, before (3) sampling until the required threshold is reached. The pooling method provides an initial 60 documents and entities for annotators to assess, which provides a reasonable starting point for annotation before the topic exploration stage.

Experienced IR annotators (the authors) judge the top 60 documents before doing the same for the top 60 entities. Documents are deliberately judged before entities to provide the annotator with the necessary topic knowledge to assess entity relevance.

3.6.2 Topic Exploration. After the initial runs are assessed, annotators are allotted between two and three hours to use live search systems to explore key dimensions of topics to find relevant documents or entities. Annotators need to construct a minimum of 6 new manual query reformulations. Figure 2 shows the query reformulations for the economics-1 topic. Annotators are encouraged to run these queries through a commercial search engine for spell checking and evaluate whether the results are on topic.

The live search systems use a hybrid BM25, BM25 with RM3 expansion and ANCE for initial retrieval, with re-ranking from MonoT5. This system returns the top 50 documents and top 50 entities to the assessor. Similar to how a researcher would use commercial search systems to explore a topic iteratively, annotators do not need to assess all returned documents and entities. Annotators are encouraged to scan returned result lists using the title and keyword highlighting to decide whether the document or entity is worth considering before annotating. This process is designed to identify the highly-relevant documents and entities not currently returned by baseline systems. Annotators are encouraged to keep searching until they cannot find new relevant documents or entities.

3.6.3 Judgments. Table 4 shows the distribution of judgments across the 42 judged topics, which includes 6,186 document judgments (147.3 per topic) and 11,323 entity judgments (269.6 per topic). *Highly Valuable (3)* only makes up 7% of document judgments and

7% of entity judgments. CODEC also releases the manual query reformulations, with the topic exploration phase providing around 74% of overall judgements. There are 387 additionally issued queries overall (9.2 per topic), which can be used to explore query performance prediction or system improvement via query reformulations.

3.6.4 Evaluation. We provide TREC-style query-relevance files with graded relevance judgments (0-3) for entity and document evaluation. The official measures for both tasks include MAP and Recall@1000 with binary relevance above 1 (i.e. relevance mappings: *0=0.0, 1=0.0, 2=1.0, 3=1.0*), and NDCG@10 with custom weighted relevance judgments (i.e. relevance mappings: *0=0.0, 1=0.0, 2=1.0, 3=2.0*). We deliberately gear measures toward the most key documents and entities (i.e. relevance scores of 2 or 3) to prioritise systems ranking these higher vs more tangential but on-topic information (i.e. relevance score of 1).

MAP assumes the user wants to find many relevant documents or entities, exposing ranking order throughout the run. On the other hand, NDCG10 with custom scaling to overweight critical information aim to provide a clear signal of whether systems highly rank the essential documents and entities. Due to recall being important for research-based tasks, Recall@1000 show missed information.

4 EXPERIMENTAL RESULTS

We conduct an in-depth analysis of sparse, dense and neural re-ranking systems on CODEC across document and entity tasks. Document ranking shows a neural re-ranker with query expansion is the best performing system, and entity ranking is particularly challenging for neural systems in a zero-shot setting. We highlight critical system failures, including models lacking (1) the ability to filter based on entities and relationships and (2) identify latent dimensions of the topic. Using CODEC's aligned document and entity judgments, we show that an entity-based query expansion technique significantly outperforms other systems. We also demonstrate how manual query reformulations can improve system performance. Firstly, showing the best query reformulation significantly outperforms the original query. Secondly, we demonstrate that a reformulation-based query expansion technique significantly outperforms other systems.

4.1 Systems

For sparse retrieval methods, the full text of entities and documents are both indexed using Pyserini [21], with Porter stemming, and stopwords removed. We use the released 'standard' four folds for cross-validation on sparse baselines and release the tuned parameters for each fold. We optimise **BM25** [32] for MAP via parameter grid search of $k1$ (between 0.1 and 5.0 with step of 0.2) and b (between 0.1 and 1.0 with step of 0.1).

For **BM25+RM3** [1], we use tuned $k1$ and b fold parameters for BM25, and optimise RM3 for MAP via parameter grid search of fb_terms (between 5 and 95 with step of 5), fb_docs (between 5 and 20 with step of 5), and $original\_query\_weight$ (between 0.2 and 0.8 with step of 0.1).

ANCE [41] is a dense retrieval model that constructs harder negative samples using the Approximate Nearest Neighbor (ANN) index. We use an MS Marco fined-tune ANCE model and Pyserini's wrapper for easy indexing. Following the methodology in the ANCE paper, **ANCE+FirstP** takes the first 512 BERT tokens of each document to represent that document. While **ANCE+MaxP** shards the document into a maximum of four 512-token shards with no overlap, and the highest-scoring shard represents the document. For entity ranking, we solely used ANCE+FirstP due to computational overhead. Using the first paragraph of Wikipedia to represent an entity is common practice in entity linking [39].

T5 [29] is state-of-the-art LM re-ranker that casts text re-ranking into a sequence-to-sequence setting and has shown impressive results. We use Pygaggle's [21] MonoT5 model, which is fine-tuned using MS Marco. The model is not fine-tuned specifically on CODEC and is used in a transfer-learning setup because of the size and scope of the current benchmark. For document and entity ranking, we employ a max-passage approach similar to Nogueira et al. [29] to re-rank initial retrieval runs (BM25, BM25+RM3, ANCE-FirstP, ANCE-MaxP). The document is sharded in 512 tokens shards with a 256 overlapping token window (maximum 12 shards per document), and the highest scored shard is taken to represent the document.

Significance testing is conducted using a paired-t-test approach at a 5% thresholds, which is common within the IR community [35].

4.2 Analysis of Current Systems

The official evaluation measures are calculated on runs to a depth of 1,000 documents and entities (full result tables in Github repository). CODEC performs all evaluation using the *ir_measures* package [23] and provides commands to make evaluation straightforward.

Table 5: Document ranking performance. *Bold* **indicates best system and ($^{\triangle}$) indicates 5% paired-t-test significance against BM25.**

	MAP	NDCG@10	Recall@1000
BM25	0.213	0.322	0.762
BM25+RM3	0.233$^{\triangle}$	0.327	**0.800**$^{\triangle}$
ANCE-MaxP	0.186	0.363	0.689
BM25+T5	0.340$^{\triangle}$	0.468$^{\triangle}$	0.762
BM25+RM3+T5	**0.346**$^{\triangle}$	0.472$^{\triangle}$	**0.800**$^{\triangle}$
ANCE-MaxP+T5	0.316$^{\triangle}$	**0.481**$^{\triangle}$	0.689

4.2.1 Document Ranking. Table 5 shows system performance for document ranking. Based on Recall@1000 of 0.80 the best performing method is BM25+RM3, outperforming BM25 and dense retrieval from ANCE-MaxP. BM25+RM3 adds pseudo-relevant terms to the query based on a first-pass retrieval run; 80-95 terms are optimal. For example, RM3 improves Recall@1000 on economics-18 topic, *Was the crash that followed the dot-com bubble an overreaction considering the ultimate success of the internet?* by 32% by adding or increasing the weight of terms such as [Amazon], [Pets.com], and [crash]. Many of these terms are entities, which supports research based on entity expansion.

An example of a hard topic for initial retrieval is economics-12, *What are the common problems or criticisms aimed at public sector enterprises?*, with Recall@1000 of under 0.55 for all systems. This topic requires a lot of latent knowledge, and analysis of relevant documents shows they contain minimal keyword overlap with the query. This is supported by the annotator having to enter sixteen

wide-ranging queries reformulations to find relevant documents and entities.

T5-MaxP improves all initial retrieval runs, with BM25+RM3+T5 having the highest MAP (0.346) and ANCE-MaxP+T5 having the highest NDCG@10 (0.481). However, an overall MAP of under 0.35 and NDCG@10 under 0.5 leave sufficient headroom for new document ranking systems to improve on complex queries. For comparison, TREC Deep Learning similar systems have an approximate MAP of 0.55 and NDCG@10 of 0.70.

For example, a hard topic of document re-ranking across all systems is politics-22, *What was the role of technology in the Arab Spring?*. BM25+RM3+T5 has a Recall@1000 of 0.8 but an NDCG@10 of only 0.20. By analysing the top-ranked baseline runs, it is clear that models cannot filter documents based on 'technology' (concept) used during the 'Arab Spring' (event). For example, several top-ranked documents discuss Arab startups, science in Islamic World, or Blockchain in politics.

Table 6: Entity ranking performance. *Bold* **indicates best system and ($^{\triangle}$) indicates 5% paired-t-test significance against BM25.**

	MAP	NDCG@10	Recall@1000
BM25	0.181	0.397	0.615
BM25+RM3	**0.209**$^{\triangle}$	**0.412**	**0.685**$^{\triangle}$
ANCE-FirstP	0.076	0.269	0.340
BM25+T5	0.172	0.361	0.615
BM25+RM3+T5	0.179	0.362	**0.685**$^{\triangle}$
ANCE-FirstP+T5	0.136	0.407	0.340

4.2.2 Entity Ranking. Table 6 shows the system performance on the entity ranking task. Performance for entity ranking is lower when compared to document ranking, emphasising that entity ranking is a challenging task within the CODEC setup. The best system is BM25+RM3 with Recall@1000 of 0.685, which has a statistically significant difference compared with BM25. ANCE-FirstP's Recall@1000 of 0.340 is significantly worse than other initial retrieval methods. This could be partially driven by ANCE only using the first passage (and not the whole Wikipedia page) or entity ranking being different from MS Marco document ranking fine-tuning.

T5 re-ranking zero-shot does not improve well-tuned sparse retrieval systems but improves ANCE-FirstP initial retrieval run. The best end-to-end retrieval system is BM25 with RM3 expansion, with a MAP of 0.209 and an NDCG@10 of 0.412.

Analysis of system failures shows key concepts proved particularly hard to retrieve. For example, in topic economics-2, *What technological challenges does Bitcoin face to becoming a widely used currency?*, all retrieval systems return anticipated named entities, i.e. [Bitcoin], [Blockchain], and [Satoshi Nakamoto]. However, systems miss the key concepts that are needed to truly understand this information need, i.e. [Transaction time], [Transaction cost], [Quantum technology], [Carbon footprint], and [Cyberattack]. Looking at the judgment mappings, these missed entities come from manual queries issued by the researcher looking at these specific dimensions. This motivates improved representations of entities that incorporate entity language models based upon document mentions that are more query-specific.

Topics with core entities explicitly named in the query have better entity ranking performance. For example, history-17, *How significant was Smallpox in the Spanish defeat of the Aztecs?*, all systems placed [History of smallpox in Mexico], [Fall of Tenochtitlan], and [Spanish conquest of the Aztec Empire] in top ranks.

4.3 Entities in Document Ranking

CODEC allows researchers to explore the role of entities in document ranking using the provided document judgment, entity judgments, and entity links that connect documents and entities.

To understand the relationship between document and entity relevance, we take the 6,186 document judgments and map the relevance of entities mentioned in each document using the entity links and entity judgments. We assume entities without judgments are *Not Relevant*. We analyse both top-1 predicted entity and top-k candidate entities for document ranking. Table 3 shows the Pearson Correlation Coefficient between document relevance and the percentage of entities in the document grouped by relevance, i.e. *Not Relevant*, *Not Valuable*, *Somewhat Valuable*,*Very Valuable*. Both predicted entity (+0.19) and candidate entities (+0.22) support that documents with higher proportions of *Very Valuable* entities are positively correlated with document relevance.

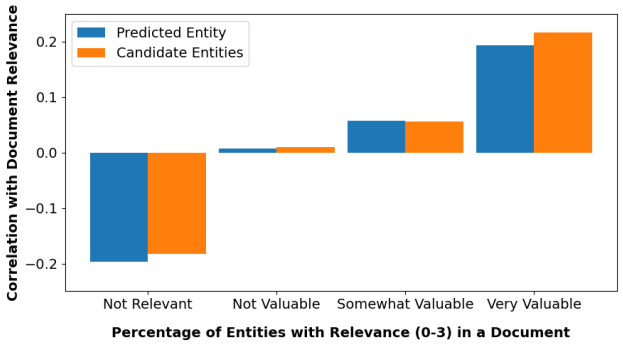

Figure 3: Correlation of document and entity relevance.

We develop an entity query feedback method to build on these findings. Prior work has shown enriching queries with entity-based information improves ad hoc ranking performance [10].

Entity-QE is an oracle entity feedback method that enriches the query with names of relevant entities taken from entity judgments. We use Pyserini BM25 for initial retrieval, removing stopwords, using Porter stemming, and use CODEC tuned BM25 fold parameters. With CODEC standard four-folds, we cross-validate the (1) weighting of original query terms, (2) weighting of *Very Valuable* entity terms, and (3) weighting of *Somewhat Valuable* terms. Across the four-folds, Entity-QE term weighting is: (1) original queries average 9.2 terms with 80% weighting, with *Very Valuable* entities adding 42.6 terms on average with 16% weighting, and *Somewhat Valuable* entities adding 77.8 terms on average with 4% weighting.

Table 7: Entity-QE Document ranking. *Bold* **indicates best system and ($^{\triangle}$) indicates 5% paired-t-test significance against BM25+RM3+T5.**

	MAP	NDCG@10	Recall@1000
BM25+RM3	0.223	0.327	0.800
Entity-QE	0.287	0.405	**0.857**$^{\triangle}$
BM25+RM3+T5	0.346	0.472	0.800
Entity-QE+T5	**0.356**$^{\triangle}$	**0.476**	**0.857**$^{\triangle}$

Table 7 shows Entity-QE improves Recall@1000 to 0.857, which is statistically significant when compared to the best initial retrieval systems BM25+RM3. **Entity-QE+T5** uses T5 as a re-ranker with the same setup as used in baseline runs and improves NDCG@10 to 0.476 and MAP to 0.356, a statistically significant improvement. Overall, these findings support that entity-centric ranking methods benefit for complex topics. CODEC having aligned document and entity judgments will enable new classes of neural ranking models to be developed and evaluated.

4.4 Query Reformulation

In this section, we study the utility of the manually reformulated queries used by the experts. We show that the best manual reformulation outperforms the original query on document and entity ranking. We also develop a query expansion method that uses all query reformulations that improve over the strong baselines.

4.4.1 Best Reformulation vs Original Query. We use a tuned BM25 and RM3 expansion model to analyse the performance of query reformulations against the original query, as this is a strong system across document and entity ranking. Figure 4 shows the distribution of the best query reformulation against the original query across document and entity ranking for MAP, NDCG@10, and Recall@1000.

The best query reformulation improves Recall@1000 of document ranking to 0.845, a statistically significant difference. As is depicted in the boxplot, the best reformulation leads to almost 75% of topics having Recall@1000 over 0.80. However, the best query reformulation has a smaller relative improvement on Recall@1000 for entity ranking (0.712) and is not statistically significant. This suggests that several query reformulations are required for a robust initial entity ranking (as shown in Section 4.4.2).

Analysing history-6 topic, *What were the lasting social changes brought about by the Black Death?*, the original query performs poorly with a Recall@1000 of 0.428 on document ranking. However, seven of thirteen query reformulations have Recall@1000 between 0.810 and 0.905, i.e *The black death (bubonic plague) and end of feudalism*, *The black death (bubonic plague) and the Renaissance*, and *bubonic plague / black death and Roman Catholic Church*. The researcher is iterating on entity names and synonym expansion to identify missing documents and entities.

The best reformulation significantly improves document and entity ranking compared to the original query on MAP and NDCG@10 measures. Document ranking MAP improves from 0.233 to 0.270, and NDCG@10 from 0.327 to 0.407. NDCG@10 saw the largest relative improvement, with around 75% topics having an NDCG@10

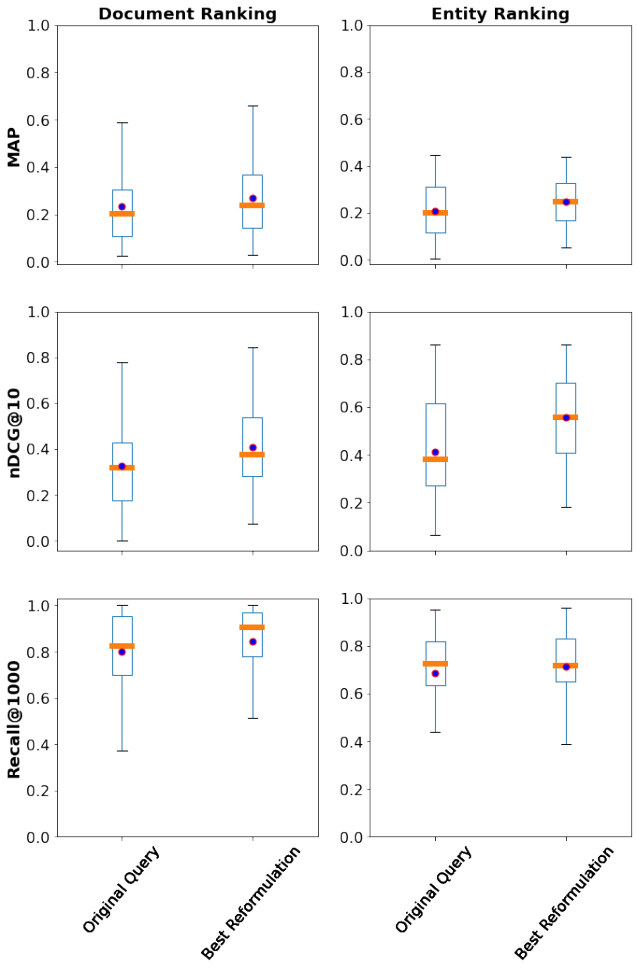

Figure 4: Boxplot BM25+RM3 topic performance of (1) original query, and (2) best manual query reformulation. Blue dot indicates means and orange line median across topics.

over 0.3 (i.e. proportionally fewer failing topics than the original query). Similarly, the best query reformulation significantly improves entity ranking, with MAP improving from 0.209 to 0.248 and NDCG@10 from 0.412 to 0.557.

Entity ranking NDCG@10 saw a 35% improvement due to the best query reformulation, the largest relative improvement of any measure across either task. Analysing the runs, this was driven by query reformulations accessing specific clusters of highly relevant entities within the top ranks. For example, the original query for topic history-15, *Why did Winston Churchill lose the 1945 General Election after winning World War II?*, had an NDCG@10 of 0.323. The best query reformulation, *Appeasement and Great Depression cost Conservatives in 1945 General Election*, improves NDCG@10 to 0.609. The improvement of top-ranked entities is due to the introduction of key events (i.e. [Appeasement] and [Great Depression]) and entities (i.e. [Conservatives]) being part of the query reformulation.

4.4.2 Reformulation Query Expansion. We develop a query feedback method **Reform-QE**, which uses both the original and query reformulation terms. We use BM25 in a similar setup to Entity-QE, cross-validating the weighting of the original query terms against the weighting of the aggregate query reformulation terms. The original queries average 9.2 terms, and the aggregate query reformulation averages 42.8 terms. For document ranking, the original queries average 66.7% weighting and aggregate query reformulation averages 33.3% weighting across the folds. For entity ranking, the original queries average 60% weighting and aggregate query reformulation averages 40% weighting across the folds. **Reform-QE+T5** uses T5 as a re-ranker with the same setup as used in baseline runs.

Table 8 depicts document ranking results for the query feedback methods that use the query reformulations. Reform-QE significantly improves over the best initial retrieval system, BM25+RM3+T5, achieving Recall@1000 of 0.864. Reform-QE+T5 also has the highest NDCG@10 and a statistically significant improvement in MAP when compared with BM25+RM3+T5.

Table 8: Reform-QE Document ranking. *Bold* **indicates best system and ($^{\triangle}$) indicates 5% paired-t-test significance against BM25+RM3+T5.**

	MAP	NDCG@10	Recall@1000
BM25+RM3	0.223	0.327	0.800
Reform-QE	0.275	0.384	**0.864**$^{\triangle}$
BM25+RM3+T5	0.346	0.472	0.800
Reform-QE+T5	**0.357**$^{\triangle}$	**0.474**	**0.864**$^{\triangle}$

Table 9 shows entity ranking results of the query feedback methods that use the query reformulations. Reform-QE significantly outperforms the best entity system, BM25+RM3, across MAP, NDCG@10, and Recall@1000. There is a larger relative improvement when using query reformulations compared to document ranking, highlighting how several queries are needed to expose the full range of relevant entities.

Table 9: Reform-QE Entity ranking. *Bold* **indicates best system and ($^{\triangle}$) indicates 5% paired-t-test significance against BM25+RM3.**

	MAP	NDCG@10	Recall@1000
BM25+RM3	0.209	0.412	0.685
Reform-QE	**0.253**$^{\triangle}$	**0.525**$^{\triangle}$	**0.738**$^{\triangle}$

Overall, query reformulations offer systems a chance to explore complex topics and access information about key dimensions of the topic not explicitly expressed in the query. CODEC query reformulations allow research into query reformulation or query performance prediction on complex topics.

5 CONCLUSION

We introduce CODEC, a document and entity ranking resource that focuses on complex research topics. Social science researchers produce 42 topics spanning history, economics, and politics. To support open research, we create a new semantically annotated focused collection derived from subsets of the Common Crawl. CODEC is grounded to the KILT's Wikipedia knowledge base for entity linking and retrieval. We provide 17,509 document and entity judgments (416.9 per topic) by assessing the pooled initial runs and manual exploration of the topics using interactive search systems, adding 387 manual query reformulations (9.2 per topic).

CODEC system analysis demonstrates topics are challenging for state-of-the-art traditional models and neural rankers. Failures demonstrate encoding entities and relationships is challenging for both document and entity ranking. Specifically, queries with large amounts of latent knowledge, where new expansion techniques are a promising research direction.

We find that document relevance is positively correlated with the occurrence of relevant entities. We leverage this relationship with an entity query expansion method that outperforms strong baseline systems on document ranking. We also demonstrate that query reformulation can play an important role in accessing latent dimensions within complex topics. Both individual query reformulations and aggregated reformulations improve document and entity ranking. Overall, this resource represents an important step toward developing and evaluating entity-centric search models on complex topics.

6 FUTURE WORK

We envision CODEC to be an evolving collection, with additional judgments and tasks added in the future, i.e. knowledge grounded generation, passage ranking, and entity linking. The topics could also be further enhanced with facet annotations and semantic annotations to support tail and non-KG entities research.

7 ACKNOWLEDGEMENTS

The authors would like to thank all the domain experts who gave up their valuable time to help develop CODEC, especially Jamie Macfarlane and Louise Lu. Additionally, we'd like to acknowledge Federico Rossetto for his support and help with visualisation. This work is supported by the 2019 Bloomberg Data Science Research Grant, the Engineering and Physical Sciences Research Council grant EP/V025708/1, and the 2019 Google Research Grant.

REFERENCES

[1] Nasreen Abdul-Jaleel, James Allan, W Bruce Croft, Fernando Diaz, Leah Larkey, Xiaoyan Li, Mark D Smucker, and Courtney Wade. 2004. UMass at TREC 2004: Novelty and HARD. *Computer Science Department Faculty Publication Series* (2004), 189.

[2] Alan Akbik, Duncan Blythe, and Roland Vollgraf. 2018. Contextual string embeddings for sequence labeling. In *Proceedings of the 27th international conference on computational linguistics*. 1638–1649.

[3] James Allan, Donna Harman, Evangelos Kanoulas, Dan Li, Christophe Van Gysel, and Ellen Voorhees. 2017. TREC 2017 Common Core Track Overview. In *Proceedings of the Twenty-Sixth Text REtrieval Conference (TREC 2017)*. Gaithersburg, Maryland.

[4] Krisztian Balog and Robert Neumayer. 2013. A test collection for entity search in DBpedia. In *Proceedings of the 36th international ACM SIGIR conference on Research and development in information retrieval*. 737–740.

[5] Nicola De Cao, Gautier Izacard, Sebastian Riedel, and Fabio Petroni. 2021. Autoregressive Entity Retrieval. In *International Conference on Learning Representations*. https://openreview.net/forum?id=5k8F6UU39V

[6] Shubham Chatterjee and Laura Dietz. 2021. Entity Retrieval Using Fine-Grained Entity Aspects. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*. 1662–1666.

[7] Nick Craswell, Daniel Campos, Bhaskar Mitra, Emine Yilmaz, and Bodo Billerbeck. 2020. ORCAS: 20 million clicked query-document pairs for analyzing search. In *Proceedings of the 29th ACM International Conference on Information & Knowledge Management*. 2983–2989.

[8] Nick Craswell, Bhaskar Mitra, Emine Yilmaz, and Daniel Campos. 2021. Overview of the TREC 2020 deep learning track. In *Text REtrieval Conference (TREC)*. TREC.

[9] J Shane Culpepper, Guglielmo Faggioli, Nicola Ferro, and Oren Kurland. 2021. Do hard topics exist? A statistical analysis. In *IIR*.

[10] Jeffrey Dalton, Laura Dietz, and James Allan. 2014. Entity query feature expansion using knowledge base links. *Proceedings of the 37th international ACM SIGIR conference on Research & development in information retrieval* (2014).

[11] Gianluca Demartini, Tereza Iofciu, and Arjen P de Vries. 2009. Overview of the INEX 2009 entity ranking track. In *International Workshop of the Initiative for the Evaluation of XML Retrieval*. Springer, 254–264.

[12] Laura Dietz. 2019. ENT Rank: Retrieving entities for topical information needs through entity-neighbor-text relations. In *Proceedings of the 42nd international ACM SIGIR conference on research and development in information retrieval*. 215–224.

[13] Laura Dietz, Manisha Verma, Filip Radlinski, and Nick Craswell. 2017. TREC Complex Answer Retrieval Overview.. In *TREC*.

[14] Jiafeng Guo, Gu Xu, Xueqi Cheng, and Hang Li. 2009. Named entity recognition in query. In *Proceedings of the 32nd international ACM SIGIR conference on Research and development in information retrieval*. 267–274.

[15] Faegheh Hasibi, Fedor Nikolaev, Chenyan Xiong, Krisztian Balog, Svein Erik Bratsberg, Alexander Kotov, and Jamie Callan. 2017. DBpedia-entity v2: a test collection for entity search. In *Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval*. 1265–1268.

[16] Samuel Huston and W Bruce Croft. 2010. Evaluating verbose query processing techniques. In *Proceedings of the 33rd international ACM SIGIR conference on Research and development in information retrieval*. 291–298.

[17] Omar Khattab and Matei Zaharia. 2020. Colbert: Efficient and effective passage search via contextualized late interaction over bert. In *Proc. of SIGIR*. 39–48.

[18] Ravi Kumar and Andrew Tomkins. 2010. A characterization of online browsing behavior. In *Proceedings of the 19th international conference on World wide web*. 561–570.

[19] Dawn Lawrie, James Mayfield, Douglas W. Oard, and Eugene Yang. 2022. HC4: A New Suite of Test Collections for Ad Hoc CLIR. https://arxiv.org/abs/2201.09992

[20] Canjia Li, Andrew Yates, Sean MacAvaney, Ben He, and Yingfei Sun. 2020. PARADE: Passage Representation Aggregation for Document Reranking. *arXiv:2008.09093* (2020).

[21] Jimmy Lin, Xueguang Ma, Sheng-Chieh Lin, Jheng-Hong Yang, Ronak Pradeep, and Rodrigo Nogueira. 2021. Pyserini: A Python toolkit for reproducible information retrieval research with sparse and dense representations. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*. 2356–2362.

[22] Binsheng Liu, Nick Craswell, Xiaolu Lu, Oren Kurland, and J Shane Culpepper. 2019. A comparative analysis of human and automatic query variants. In *Proceedings of the 2019 ACM SIGIR International Conference on Theory of Information Retrieval*. 47–50.

[23] Sean MacAvaney, Craig Macdonald, and Iadh Ounis. 2022. Streamlining Evaluation with ir-measures. In *ECIR*. https://arxiv.org/abs/2111.13466

[24] Sean MacAvaney, Andrew Yates, Arman Cohan, and Nazli Goharian. 2019. CEDR: Contextualized Embeddings for Document Ranking. In *Proceedings of the 42nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR 2019)*. Paris, France, 1101–1104.

[25] Sean MacAvaney, Andrew Yates, Sergey Feldman, Doug Downey, Arman Cohan, and Nazli Goharian. 2021. Simplified Data Wrangling with ir_datasets. In *SIGIR*.

[26] Iain Mackie, Jeffrey Dalton, and Andrew Yates. 2021. How Deep is your Learning: the DL-HARD Annotated Deep Learning Dataset. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*.

[27] Federico Nanni, Simone Paolo Ponzetto, and Laura Dietz. 2018. Entity-aspect linking: providing fine-grained semantics of entities in context. In *Proceedings of the 18th ACM/IEEE on Joint Conference on Digital Libraries*. 49–58.

[28] Tri Nguyen, Mir Rosenberg, Xia Song, Jianfeng Gao, Saurabh Tiwary, Rangan Majumder, and Li Deng. 2016. MS MARCO: A Human Generated MAchine Reading COmprehension Dataset. *arXiv:1611.09268v1* (2016).

[29] Rodrigo Nogueira, Zhiying Jiang, Ronak Pradeep, and Jimmy Lin. 2020. Document Ranking with a Pretrained Sequence-to-Sequence Model. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: Findings*. 708–718.

[30] Fabio Petroni, Aleksandra Piktus, Angela Fan, Patrick Lewis, Majid Yazdani, Nicola De Cao, James Thorne, Yacine Jernite, Vladimir Karpukhin, Jean Maillard, et al. 2021. KILT: a Benchmark for Knowledge Intensive Language Tasks. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. 2523–2544.

[31] Jordan Ramsdell and Laura Dietz. 2020. A Large Test Collection for Entity Aspect Linking. In *Proceedings of the 29th ACM International Conference on Information & Knowledge Management*. 3109–3116.

[32] Stephen E Robertson and Steve Walker. 1994. Some simple effective approximations to the 2-poisson model for probabilistic weighted retrieval. In *SIGIR'94*. Springer, 232–241.

[33] Tony Russell-Rose, Jon Chamberlain, and Leif Azzopardi. 2018. Information retrieval in the workplace: A comparison of professional search practices. *Information Processing & Management* 54, 6 (2018), 1042–1057.

[34] Christopher Sciavolino, Zexuan Zhong, Jinhyuk Lee, and Danqi Chen. 2021. Simple Entity-Centric Questions Challenge Dense Retrievers. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*. 6138–6148.

[35] Mark D Smucker, James Allan, and Ben Carterette. 2007. A comparison of statistical significance tests for information retrieval evaluation. In *Proceedings of the sixteenth ACM conference on Conference on information and knowledge management*. 623–632.

[36] Johannes M. van Hulst, Faegheh Hasibi, Koen Dercksen, Krisztian Balog, and Arjen P. de Vries. 2020. REL: An Entity Linker Standing on the Shoulders of Giants. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '20)*. ACM.

[37] Ellen M. Voorhees. 2004. Overview of the TREC 2004 Robust Track. In *Proceedings of the Thirteenth Text REtrieval Conference (TREC 2004)*. Gaithersburg, Maryland, 52–69.

[38] Ledell Wu, Fabio Petroni, Martin Josifoski, Sebastian Riedel, and Luke Zettlemoyer. 2020. Zero-shot Entity Linking with Dense Entity Retrieval. CoRR abs/1911.03814 (2019). (2020).

[39] Ledell Yu Wu, F. Petroni, Martin Josifoski, Sebastian Riedel, and Luke Zettlemoyer. 2020. Zero-shot Entity Linking with Dense Entity Retrieval. In *EMNLP*.

[40] Chenyan Xiong, Jamie Callan, and Tie-Yan Liu. 2017. Word-entity duet representations for document ranking. In *Proceedings of the 40th International ACM SIGIR conference on research and development in information retrieval*. 763–772.

[41] Lee Xiong, Chenyan Xiong, Ye Li, Kwok-Fung Tang, Jialin Liu, Paul N Bennett, Junaid Ahmed, and Arnold Overwijk. 2020. Approximate Nearest Neighbor Negative Contrastive Learning for Dense Text Retrieval. In *International Conference on Learning Representations*.