*MEDINFO 2017: Precision Healthcare through Informatics*
*A.V. Gundlapalli et al. (Eds.)*

*doi:10.3233/978-1-61499-830-3-629*

# Aligned-Layer Text Search in Clinical Notes

**Stephen Wu[a], Andrew Wen[a], Yanshan Wang[b], Sijia Liu[b], Hongfang Liu[b]**

*[a] Department of Medical Informatics and Clinical Epidemiology,*
*Oregon Health & Science University, Portland, OR, USA,*
*[b] Division of Biomedical Informatics, Mayo Clinic, Rochester, MN, USA*

### Abstract

*Search techniques in clinical text need to make fine-grained semantic distinctions, since medical terms may be negated, about someone other than the patient, or at some time other than the present. While natural language processing (NLP) approaches address these fine-grained distinctions, a task like patient cohort identification from electronic health records (EHRs) simultaneously requires a much more coarse-grained combination of evidence from the text and structured data of each patient's health records. We thus introduce aligned-layer language models, a novel approach to information retrieval (IR) that incorporates the output of other NLP systems. We show that this framework is able to represent standard IR queries, formulate previously impossible multi-layered queries, and customize the desired degree of linguistic granularity.*

***Keywords:***

Natural Language Processing; Information Storage and Retrieval; Electronic Health Records

## Introduction

Search (i.e., information retrieval) techniques in clinical text face the challenge of coarse vs. fine granularity. In a setting such as electronic health records (EHRs), the goal is usually to find a broad characterization of a whole patient from that patient's diverse assemblage of text and structured data. How to combine these pieces of evidence into a patient-level judgment is an area of active research. Moreover, in clinical text, dictionary matching for a symptom like "hypertension" must be augmented by fine-grained algorithms to ensure that the symptom is not negated (e.g., "no complaints of …"), about someone other than the patient (e.g., "family history of"), or a host of other possibilities. Textual variants of these fine-grained patterns abound as well, spawning natural language processing (NLP) methods to find concepts of interest from acceptable textual contexts.

Therefore, we aim to incorporate arbitrary NLP-derived features into information retrieval (IR) methods on clinical text, allowing for greater control of the granularity in search. Our main contribution is the *aligned-layer information retrieval* model, which we specify by defining the "layered" nature of text and the nature of phrases in this setting. This model is a straightforward extension of the language modeling approach to IR and its feature-centric successors.

We adopt the "layer" terminology from the language resources community in which, for example, treebanking and propbanking are different layers of annotations on the same text. The crucial contribution in our work is a model in which all text-derived "layers" are aligned by token, and can be utilized and scored simultaneously during search.

After describing our aligned layer approach, we report on a preliminary system implementation and evaluation of the model. For our evaluation, we focus on the problem of cohort identification within clinical notes from the EHR.

### Related Work

Preliminary explorations with concepts (from the Unified Medical Language System (UMLS)) in language modeling have met with moderate success [1-3] in the past. Language modeling in IR [4-6] builds on a rich tradition of probabilistic IR [7], and has a successful history of ranking documents based on well-motivated textual features. The most commonly-used textual features are from Metzler and Croft's dependence model, modeling the probabilistic relationships between query terms (and a candidate document) as Markov Random Fields (MRFs). These term–term dependencies significantly outperformed the original bag-of-words language modeling approaches to IR from which they arose[4; 5]. Subsequently, query hypergraphs made it possible to model higher-order dependencies (e.g., multiword–multiword or multiword–term dependencies) [8]. While the representational power of these feature-based models allows for arbitrary *non-textual* features as well (e.g., named entities or dependencies, such as we introduce here), non-textual features have received minimal attention in the literature.

Recent work on joint text and concept search [2] and split-layer language models [1] extended the notion of mixing different document representations together [9; 10]. Arising out of the medical domain, these techniques all showed some benefit to considering multiple "layers" simultaneously. Our work here extends the discussion of language modeling layers, pushing the question of what semantic representations (or granularity thereof) are effective for IR.

Positional language models [11] attempt to model the intuition that closer textual proximity can correspond to greater association. This shares an important intuition with aligned-layer models: position is important in IR. However, our accounting of position is not to model the textual distance, but the content and relationships that are latent within text at the same locations. A number of existing studies [1; 2] require an NLP preprocessing step prior to indexing the collection, and utilize these text-derived features in search. However, these attempts fail to allow navigation among, and correspondences between, those structures.

The task of query formulation [12] is somewhat upstream task to our layered models; query formulation can include the introduction of term dependencies [13], weighting of terms [14], query expansion [15], and the parameterization of associated weights according to additional corpora [16; 17]. Our hope is that future query formulation techniques will make use of aligned-layer structures.

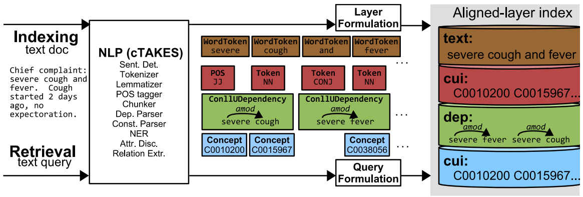
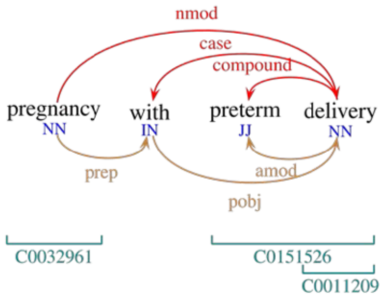


*Figure 1 - cTAKES processing is followed by the indexing of results into various layers (indexes)*

The TREC Medical Records track [18; 19] provided resources for significant innovation in medical IR, which used in this paper for evaluation: collections of clinical text, information needs, and judgments. Language modeling techniques have been successful in this clinical text setting [2], as well as in health-related web search exemplified by the CLEF eHealth challenges [20].



| text tok | $l_{0,0}$<br>pregnancy | $l_{0,1}$<br>with | $l_{0,2}$<br>preterm | $l_{0,3}$<br>delivery |
|---|---|---|---|---|
| pos tag | $l_{1,0}$<br>span=(0,1)<br>*NN* | $l_{1,1}$<br>span=(1,1)<br>*IN* | $l_{1,2}$<br>span=(2,1)<br>*JJ* | $l_{1,3}$<br>span=(3,1)<br>*NN* |
| named ent | $l_{2,0}$<br>span=(0,1)<br>C0032961 | $l_{2,1}$<br>span=(2,2)<br>C0151526 | $l_{2,2}$<br>span=(3,1)<br>C0011209 | |
| stanf dep | $r_{3,0}$<br>source=(0,1)<br>target=(3,1)<br>*nmod* | $r_{3,1}$<br>source=(3,1)<br>target=(1,1)<br>*case* | $r_{3,2}$<br>source=(3,1)<br>target=(2,1)<br>*compound* | |
| conll dep | $r_{4,0}$<br>source=(0,1)<br>target=(1,1)<br>*prep* | $r_{4,1}$<br>source=(1,1)<br>target=(3,1)<br>*pobj* | $r_{4,2}$<br>source=(3,1)<br>target=(2,1)<br>*amod* | |

*Figure 2 - Aligned layers (tokens, part-of-speech tags, named entities, Stanford dependencies, and CoNLL-X dependencies) for a sample query "pregnancy with preterm delivery." Span, source, and target indices are with respect to the L₀ layer's indices.*

## Methods

We now turn our attention to the aligned-layer retrieval model and its implementation for our experiments. Let us assume that we run an NLP pipeline as a preprocess; e.g., the clinical Text Analysis and Knowledge Extraction System (cTAKES)[1] [21]. Developed specifically for clinical text, this pipeline produces NLP artifacts as illustrated in the left and middle Figure 1.

### Indexing: Aligning text and NLP artifacts

We represent this NLP-preprocessed text as a finite sequence of multiple aligned layers. Specifically, a document or query is composed of multiple layers $L_0, L_1, L_2, \ldots$ We reserve $L_0$ to be a *base layer* of the original text; namely, a sequence of tokens (optionally stopped and stemmed) by which all other layers are aligned.

Figure 2 illustrates a query text (tokens) as a base layer, with several underlying layers—part-of-speech (POS) tags, named entities (NEs) with mappings to concept unique identifiers (CUIs), and two types of dependency parses (left). It then shows their translation into aligned layers (right).

Other than the base layer, each layer is an *artifact layer*, and is composed of a sequence of *artifacts*. For example, in Figure 2, $L_2$ is named entities, and $L_2 = l_{2,0}, l_{2,1}, l_{2,2}, \ldots = \mathrm{C0032961}, \mathrm{C0151526}, \mathrm{C0011209}, \ldots$. The artifacts are aligned with the base layer by storing 2 additional numbers: start index and length. Thus, a function $span(\cdot)$ on $l_{2,1}$, a named entity spanning the words "preterm delivery," would have $span(l_{2,1}) = (2,2)$ since the artifact concerns "preterm delivery" and starts at index 2 with a length of 2 tokens.

Additionally, we define a special *relation* artifact with a slight modification of other artifacts. For relation layers like $L_3$ (stanford dependency parses, in Figure 2), artifacts are relations between two *other* artifacts' locations. We will write these as $L_3 = r_{3,0}, r_{3,1}, r_{3,2}, \ldots$ to emphasize that these artifacts are relations. Then, $source(\cdot)$ and $target(\cdot)$ give the position and length of each of the relation's arguments; $source(r_{3,0}) = (0,1)$ points from "pregnancy" to $target(r_{3,0}) = base(r_{3,0}) = (3,1)$ "delivery."

Note that the transformation to aligned language layers is slightly lossy. In Figure 2 the $L_3$ artifacts corresponding to the head and dependent of each dependency relation are not preserved—only the $L_0$-aligned spans. The loss is minimal in most NLP structures of interest.

### Scoring: Terms and phrases

In IR language models, it is common to rank according to $\mathrm{score}(D, Q) = P(D) \cdot P(Q|D)$. The conditional probability encapsulates the intuition that an ad hoc user trying to find document $D$ will try to write an effective query $Q$. We focus our attention on the conditional probability $P(Q|D)$.

This is most simply expressed in the standard *query likelihood* model with Dirichlet smoothing (with the parameter $\mu_D$), a baseline for language modeling approaches. We write single-**term queries** with the notation of the aligned layers above, where the $a$ variable in $l_a$ simply indicates that such term queries can be written for any single layer:

$$P_{\ t}(l_a|D) = \frac{\#t_D(l_a) + \mu \cdot \frac{\#t_{\mathbf{D}}(l_a)}{|\mathbf{D}|_a}}{|D|_a + \mu} \tag{1}$$

where $\#t_D$ is the number of occurrences of the argument (a query term $l_a$) in the document; $\#t_{\mathbf{D}}$ is the number of occurrences of that argument in the whole collection. Similarly, $|D|_a$ is the number of artifacts in the document from layer $l_a$; $|\mathbf{D}|_a$ counts the same layer's artifacts in the whole collection.

With a layered representation of both queries and documents, these term operators allow for querying of artifacts in any layer $L_a$. The standard query likelihood model in other texts [5] is then just a special case of our term query utilizing only text layer artifacts $L_0$ (i.e., from $L_{text}$). The smoothed

[1] http://ctakes.apache.org

probability is also calculated with respect to frequencies in layer $L_a$. Term queries do not require *aligned* layers — they consider each layer separately.

Probabilities for individual query terms can be calculated separately and combined to produce an overall probability. This is done implicitly, but can be made explicit with a **list query**; items in the list need not be in the same layer because they are each considered individually. A list query also groups the items in the list for use in other queries.

However, list queries ignore the collocation of layered artifacts; thus, we define the **phrase query.** A phrase is two or more artifacts from an arbitrary combination of layers, e.g., $l_{ai}l_{bj}$, the $i^{\text{th}}$ artifact in $L_a$ and the $j^{\text{th}}$ artifact in $L_b$. Phrases must be specified with ordering *o* as ordered or unordered (True or False), with a window size *w* to search within (measured according to positions in the base layer $L_0$).

$$P_{\text{ph}}\left(l_{ai}l_{bj} \ldots \mid D; o, w\right) = \frac{\#\text{ph}_D(l_{ai}l_{bj}\cdots) + \mu \frac{\#\text{ph}_{\mathbf{D}}(l_{ai}l_{bj}\cdots)}{|\mathbf{D}|_a}}{\max(|D|_a, |D|_b, \ldots) + \mu} \tag{2}$$

The function $\#\text{ph}_D$ counts the number of cross-layer phrase matches in a document. Having aligned $l_{ai}l_{bj}\ldots$ to the base layer $L_0$, we can check across layers for matches within a window. Window length is between the end of the first matched artifact, and the beginning of the last matched artifact (i.e., the last artifact only needs to have start index within the window length; it can end outside the window). Writing $l_{ai}l_{bj}$ illustrates that artifacts need not arise from the same layer, and will be in different positions within their respective layers (though aligned to the base layer $L_0$). However, ordered phrases with *o*=True must additionally consist of only non-overlapping, sequential artifacts.

Whereas in single-layer queries, a probability estimate would consist of all items in that layer, there are now two or more layers to consider. It may at first seem that the denominator should be the product $|D|_a \cdot |D|_b \cdot \ldots$. This would mean: "out of all possible multi-layered phrases that could be constructed in document *D*, how likely is the specific construction we are looking for?" This choice of denominator would yield sparse probability estimates. Instead, we have chosen the size of the largest layer stipulated in the phrase. This answers the question: "of the artifacts in the largest layer, what proportion participate in a phrase consistent with the query?" This may be thought of as a backed-off estimate, and is more tractable. The choice of a denominator is an interesting area of further investigation beyond the scope of this work.

We should also note that this aligned-layer phrase queries subsume ordered and unordered phrases of feature-centric IR models [6], since those features are simply functions of $L_0$.

### Implementation

The fundamental aims of the aligned-layer IR model require significant extension in any search engine. We implemented an aligned-layer language model via a plugin to Elasticsearch with multiple components, tested with Elasticsearch 1.7.2. We used cTAKES v3.2 as a preprocess [22], producing *character-aligned* artifacts (based on Apache UIMA[2]). Aligned-layer models are similar to searching over these UIMA-based data structures, but with scored ranking, across mutiple documents, and aligned on tokens instead of characters.

For indexing, each layer was represented as a field within a document. For our evaluation and analysis, we generated the an index with the following fields:

- $L_{text}$: Word tokens, as identified by the cTAKES tokenizer; this is considered the base layer $L_0$
- $L_{cui}$: Concept Unique Identifiers (CUIs), as mapped from the UMLS Metathesaurus by cTAKES
- $L_{tui}$: Type Unique Identifiers, a many-to-one mapping that groups concepts (CUIs) into semantic types (TUIs)
- $L_{lemma}$: Normalized by a version of the National Library of Medicine's (NLM) Lexical Variant Generator (LVG)
- $L_{dep}$: Conll-U dependency parses from Clear Parser [23] trained on treebanked clinical text from Mayo Clinic
- $L_{pos}$: Part-of-speech tags produced by the OpenNLP[3] MaxEnt POS tagger, trained on treebanked clinical text from Mayo Clinic

Each artifact was marked with its span (position and length, in tokens) via Lucene Payloads; this is a trivial marking for the base layer of text $L_0$, but can be significant in other layers, such as for multi-word expressions in the CUI layer of Figures 1 and 2. In addition, the size of each field within a document was stored as metadata with that document at indexing time.

Queries are scored in Elasticsearch and Lucene via a highly optimized scoring interface. However, our scoring functions cannot be represented within that original structure; for example, the Lucene implementation of Dirichlet smoothing on a language model fails to divide by the collection frequency in some cases. Therefore, we implemented scoring via Elasticsearch Script Queries, which are typically used in a filtering context but here provide us the flexibility to score according to the models defined above.

Because they make use of aligned-layer terms and phrases, queries must be constructed and parsed differently than in other search systems. We implemented our own query parser in JavaCC to term and phrase operators in the aligned-layer query language.

### Task and Experiments

We provide a preliminary evaluation of aligned-layer language models, and of our Elasticsearch-based implementation in particular, on the task of patient cohort identification as exemplified in the 2011-2012 Text Retrieval Conference (TREC) Medical Records Track, or TREC-Med [18; 19]. In brief, there were 81 topics (34 from 2011, 47 from 2012) such as "patients with hearing loss," and the task was to return lists of relevant patient visits (a de-identified surrogate for whole patient records) from among 17,198 possible visits. System-produced lists of visits were compared with human relevance judgments (which were gathered for the TREC-Med 2011-2012 competitions).

Table 1 shows how the same query is represented across our evaluated approaches. As a baseline test TXT, we used unstemmed, stopped TREC-Med queries with the standard query likelihood model -- namely, term queries on $L_{text}$. A second baseline was CUI, where we used term queries on $L_{cui}$.

In aligned-layer IR, term operators on multiple layers can be weighed together simultaneously for results. We replaced all named entity mentions in the query with the first-matched CUI, mixing the two layers in MIX. Since the cTAKES Named Entity Recognition (NER) module actually returned a set of CUIs per named entity, we included a model CUI-LS in which named entities were replaced with all of the CUIs associated with the same span, combined in a list. Next, we augmented

[2] http://uima.apache.org

[3] https://opennlp.apache.org

the CUI list with a #ph containing $L_{text}$ terms corresponding to any named entities, terming this PH-LS.

*Table 1 - Eight versions of aligned-layer queries for Topic 125. #p are phrase queries, #l are list queries, others are term queries.*

<table>
<tr><th>Layers</th><th>Query representation</th></tr>
<tr><td>TXT</td><td>coinfected hepatitis c hiv</td></tr>
<tr><td>CUI</td><td>cui:C0019196 cui:C0019158</td></tr>
<tr><td>MIX</td><td>coinfected cui:C0019158 hiv</td></tr>
<tr><td>PH-LS</td><td>coinfected #l(cui:C0019158|hepatitis) hiv</td></tr>
<tr><td>TXT-MRF</td><td>coinfected^0.85 hepatitis^0.85 c^0.85 hiv^0.85 #p(8|true|coinfected|hepatitis)^0.1 #p(8|false|coinfected|hepatitis)^0.05 #p(8|true|hepatitis|c)^0.1 #p(8|false|hepatitis|c)^0.05 #p(8|true|c|hiv)^0.1 #p(8|false|c|hiv)^0.05 #p(12|true|coinfected|hepatitis|c)^0.1 #p(12|false|coinfected|hepatitis|c)^0.05 #p(12|true|hepatitis|c|hiv)^0.1 #p(12|false|hepatitis|c|hiv)^0.05 #p(16|true|coinfected|hepatitis|c|hiv)^0.1 #p(16|false|coinfected|hepatitis|c|hiv)^0.05</td></tr>
<tr><td>CUI-MRF</td><td>cui:C0019196^0.85 cui:C0019158^0.85 #p(8|true|cui:C0019196|cui:C0019158)^0.1#p(8|false|cui:C0019196|cui:C0019158)^0.05</td></tr>
<tr><td>MIX-MRF</td><td>coinfected^0.85 cui:C0019158^0.85 hiv^0.85 #p(8|true|coinfected|cui:C0019158)^0.1 #p(8|false|coinfected|cui:C0019158)^0.05 #p(8|true|cui:C0019158|hiv)^0.1 #p(8|false|cui:C0019158|hiv)^0.05 #p(12|true|coinfected|cui:C0019158|hiv)^0.1 #p(12|false|coinfected|cui:C0019158|hiv)^0.05</td></tr>
<tr><td>PH-LS-MRF</td><td>coinfected^0.85 #l(cui:C0019158|hepatitis)^0.85 hiv^0.85 #p(8|true|coinfected|#l(cui:C0019158|hepatitis))^0.1 #p(8|false|coinfected|#l(cui:C0019158|hepatitis))^0.05 #p(8|true|#l(cui:C0019158|hepatitis)|hiv)^0.1 #p(8|false|#l(cui:C0019158|hepatitis)|hiv)^0.05 #p(12|true|coinfected|#l(cui:C0019158|hepatitis)|hiv)^0.1</td></tr>
</table>

Searching for an optimal formulation and weighting of queries is outside the scope of this paper. However, for each of the above approaches, we also implemented a version with the Markov Random Field (MRF) "term" dependence model [24], which attempts to move beyond bag-of-words (BOW) models. Comparing the TXT-MRF row to the TXT row in Table 1 illustrates the behavior of this model: it tries out, with appropriate weights, the possibility that a multi-layered query may have phrases within it. We implemented a sequential dependence model with a limit of 5 aligned-layer components. These "term" dependence versions are marked as -MRF.

## Results

We used mean average precision (MAP) as the primary evaluation measure for evaluation. MAP provides a single-figure measure of quality across recall levels [25]. While TREC-Med 2011 reported bpref [26] as its primary evaluation metric, and TREC-Med 2012 reported infAP [27], we here report the mean average precision (MAP) due to its stability for both training and testing in previous work [28; 29]. Table 2 shows the MAP scores, where we have separated the topics from TREC-Med 2011 (left) and 2012 (right).

*Table 2 - Retrieval performance for a range of possible aligned-layer models*

| | **2011** | | **2012** | |
|---|---|---|---|---|
| **Model** | **BOW** | **MRF** | **BOW** | **MRF** |
| TXT | 0.2960 | 0.2936 | 0.2152 | 0.2224 |
| CUI | 0.3042 | 0.3119 | **0.3126** | 0.3101 |
| MIX | 0.2807 | 0.2847 | 0.2215 | 0.2167 |
| CUI-LS | 0.3076 | 0.3095 | 0.2161 | 0.2120 |
| PH-LS | **0.3185** | 0.3167 | 0.2172 | 0.2126 |

While this performance is below the state-of-the-art, it is interesting to note that it is layers with CUIs that obtain the highest performance in these tests (CUI-BOW on 2011 queries, and PH-LS-BOW on 2012 queries). Without text-layer query expansion, query logs, clickthrough data, or the like, another well-motivated, semantically-rich layer seems to be beneficial.

## Discussion

For aligned-layer language models, it is interesting to ask whether there is the possibility of fine-tuning results on individual queries. Here, then, we consider one of the worst-performing queries, topic 121, and seek to write aligned-layer queries that would improve those particular topics.

Topic 121 is "Patients with CAD who presented to the Emergency Department with Acute Coronary Syndrome and were given Plavix." The CUIs found were C0948089 (acute coronoary syndrome), C0010068 (coronary heart disease), and C0039082 (syndrome) – note the NER did not find the medication Plavix, or its generic form, clopidogrel. The highest scoring model on this topic was MIX-BOW=0.0639. In the event of a failed detection, we may consider using some kind of backoff model to less-specific expressions.

- In place of `plavix`, we searched for a dependency relationship where the head word has a lemma "give," and the child is any word with a proper noun. It then weights that relation alongside the term plavix. This brought the MAP to 0.0803.
- In place of `presented emergency department`, we considered any verbs that might describe a patient's arrival. We stipulated that there should be some verb in a dependency relationship with the phrase "emergency department." This brought the MAP to 0.0867.

The multi-layered queries clearly allow for greater coarseness or fineness according to the needs of a query.

## Conclusion

We have introduced aligned-layer language models, a novel approach to IR that incorporates the output of other NLP systems. Core to this contribution are the definition of layers, alignment, and multi-layer scoring models. We have shown that this system can represent standard IR queries, as well as formulate multi-layered queries that were previously impossible. A case study demonstrates how the aligned-layer approach may feasibly be further extended to customize linguistic granularity to specific queries.

An open question of the proposed approach is how to design the layers for different corpora. Since different corpora contain variegated domain knowledge, we could design individualized layers for each corpus. In future work, we will further explore how our aligned-layer index can serve as a corpus analysis tool for quantitative representativeness of linguistic features; this would further drive a model to automatically design and select layers. In addition to this manual approach to exploring the linguistic content, we will learn parameters and weights for the model to optimize performance on IR tasks. Finally, we will release the code for aligned-layer language models to the open-source community.

## Acknowledgements

This work is funded in part by the U.S. National Institutes of Health grant R01LM011934. Thanks to Dingcheng Li, who was involved in early work on layered language models.

## References

[1] S. Wu, D. Zhu, W. Hersh, and H. Liu, Clinical Information Retrieval with Split-layer Language Models, in: *Proceedings of the SIGIR Workshop on Health Search and Discovery*, 2013.

[2] D. Zhu and B. Carterette, Joint search in text and concept spaces for EMR-based cohort identification, in: *Bioinformatics and Biomedicine (BIBM), 2013 IEEE International Conference on*, 2013, pp. 597-601.

[3] S. Choi, J. Choi, S. Yoo, H. Kim, and Y. Lee, Semantic concept-enriched dependence model for medical information retrieval, *Journal of Biomedical Informatics* **47** (2014), 18-27.

[4] J.M. Ponte and W.B. Croft, A language modeling approach to information retrieval, in: *Proceedings of the 21st annual international ACM SIGIR conference on Research and development in information retrieval*, 1998, pp. 275-281.

[5] C. Zhai and J. Lafferty, A study of smoothing methods for language models applied to ad hoc information retrieval, in: *Proceedings of the 24th annual international ACM SIGIR conference on Research and development in information retrieval*, ACM, 2001, pp. 334-342.

[6] D. Metzler, *A feature-centric view of information retrieval*, Springer Science & Business Media, 2011.

[7] K.S. Jones, S. Walker, and S.E. Robertson, A probabilistic model of information retrieval: development and comparative experiments: Part 2, *Information Processing \& Management* **36** (2000), 809-840.

[8] M. Bendersky and W.B. Croft, Modeling higher-order term dependencies in information retrieval using query hypergraphs, in: *Proceedings of the 35th international ACM SIGIR conference on Research and development in information retrieval*, 2012, pp. 941-950.

[9] P. Ogilvie and J. Callan, Combining document representations for known-item search, in: *SIGIR'03*, 2003, pp. 143-150.

[10] C. Zhai, Statistical language models for information retrieval, *Synthesis Lectures on Human Language Technologies* **1** (2008), 1-141.

[11] Y. Lv and C. Zhai, Positional language models for information retrieval, in: *Proceedings of the 32nd international ACM SIGIR conference*, 2009, pp. 299-306.

[12] M. Bendersky, D. Metzler, and W.B. Croft, Effective query formulation with multiple information sources, in: *Proceedings of the fifth ACM international conference on Web search and data mining*, 2012, pp. 443-452.

[13] D. Metzler and W.B. Croft, A Markov random field model for term dependencies, *Proceedings of the 28th annual international ACM SIGIR conference on Research and development in information retrieval* (2005), 472.

[14] M. Bendersky, D. Metzler, and W.B. Croft, Learning concept importance using a weighted dependence model, in: *Proceedings of the third ACM international conference on Web search and data mining*, 2010, pp. 31-40.

[15] D. Metzler and W.B. Croft, Latent concept expansion using markov random fields, in: *Proceedings of the 30th annual international ACM SIGIR conference on Research and development in information retrieval*, 2007, pp. 311-318.

[16] F. Diaz and D. Metzler, Improving the estimation of relevance models using large external corpora, in: *Proceedings of the 29th annual international ACM SIGIR conference on Research and development in information retrieval*, ACM, 2006, pp. 154-161.

[17] M. Bendersky, D. Metzler, and W.B. Croft, Parameterized concept weighting in verbose queries, in: *Proceedings of the 34th international ACM SIGIR conference on Research and development in Information Retrieval*, 2011, pp. 605-614.

[18] E.M. Voorhees and R.M. Tong, Overview of the TREC 2011 Medical Records Track, in: *Proceedings of The 20th Text REtrieval Conference (TREC)*, 2011.

[19] E. Voorhees and W. Hersh, Overview of the TREC 2012 medical records track, in: *The Twenty-first Text REtrieval Conference Proceedings (TREC)*, 2012.

[20] D. Zhu, S.T.-I. Wu, J.J. Masanz, B. Carterette, and H. Liu, Using Discharge Summaries to Improve Information Retrieval in Clinical Domain, in: *CLEF (Working Notes)*, 2013.

[21] G.K. Savova, J.J. Masanz, P.V. Ogren, J. Zheng, S. Sohn, K.C. Kipper-Schuler, and C.G. Chute, Mayo clinical Text Analysis and Knowledge Extraction System (cTAKES): architecture, component evaluation and applications, *Journal of the American Medical Informatics Association* **17** (2010), 507.

[22] G.K. Savova, J.J. Masanz, P.V. Ogren, J. Zheng, S. Sohn, K.C. Kipper-Schuler, and C.G. Chute, Mayo clinical Text Analysis and Knowledge Extraction System (cTAKES): architecture, component evaluation and applications, *Journal of the American Medical Informatics Association : JAMIA* **17** (2010), 507-513.

[23] J.D. Choi and A. McCallum, Transition-based Dependency Parsing with Selectional Branching, in: *ACL (1)*, 2013, pp. 1052-1062.

[24] D. Metzler and W.B. Croft, A Markov random field model for term dependencies, in: *Proceedings of the 28th annual international ACM SIGIR conference on Research and development in information retrieval*, 2005, pp. 472-479.

[25] B. Croft, D. Metzler, and T. Strohman, *Search Engines: Information Retrieval in Practice*, Addison Wesley, 2009.

[26] C. Buckley and E.M. Voorhees, Retrieval evaluation with incomplete information, in: *Proceedings of the 27th Annual SIGIR*, 2004, pp. 25-32.

[27] E. Yilmaz, E. Kanoulas, and J.A. Aslam, A simple and efficient sampling method for estimating AP and NDCG, in: *Proceedings of the 31st annual international ACM SIGIR conference on Research and development in information retrieval*, 2008, pp. 603-610.

[28] D. Zhu, S. Wu, B. Carterette, and H. Liu, Using large clinical corpora for query expansion in text-based cohort identification, *Journal of biomedical informatics* **49** (2014), 275-281.

[29] S. Wu, D. Zhu, W. Hersh, and H. Liu, Clinical information retrieval with split-layer language models, in: *Proceedings of the Association for Computing Machinery SIGIR workshop on health search and discovery (HSD)*, 2013.

### Address for correspondence

Stephen Wu, wu.stephen.t@gmail.com, 3181 SW Sam
Jackson Park Rd, Portland, OR 97239