Omar

Ken

# WSDM-2024 Tutorial:
## Some Useful Things to Know When Combining IR and NLP: The Easy, the Hard and the Ugly

Omar Alonso, Amazon, Palo Alto, CA, USA

Kenneth Church, Northeastern University, San Jose, CA, USA

# Disclaimer

The views, opinions, positions, or strategies expressed in this talk are mine (Omar) and do not necessarily reflect the official policy or position of Amazon.

# Tutorial agenda

Introduction

Hard

Easy

Medium

Ugly

Conclusions

# Introduction

- High level overview of information access (search, QA, etc.)
- What are the components that we need?
  - IR stack (indexing, crawling, ranking, query understanding, etc.)
  - NLP stack (POS, NER, specific tasks, embeddings, transformers, etc.)
- HEMU dimensions
  - Hard
  - Easy
  - Medium
  - Ugly

# Information seeking

- The user has an information need
- Expressed as a query (or question)
- Examine search results or answer(s)
- Re-formulate if needed
- Systems
  - Information retrieval/Search engine,
  - Q&A systems
  - Chatbots
  - ChatGPT-like
  - Forums
  - Social networks

# The back-end side

- IR finds content in a collection (usually text) that satisfies an information need
- Many NLP components and techniques in the tech stack
- Classical IR systems don’t answer a question directly
- 10-blue links
- Question Answering systems need experts curated data set
- Pre-trained language models can produce prose that looks like an answer
- Combination of IR and LLMs

# Typical search session

- Two queries of two words …
  - … looking each time to at most two pages
  - … and doing two clicks per page
- IR engine needs to guess what the user wants
- Query intent
  - Navigational, informational, transactional
- Query understanding
  - Normalization, spell correction, annotation, term expansion, query re-writing
- Current search systems
  - Index, retrieve, and rank
  - Not many changes

# Quick terminology recap – document retrieval

- Classic search model
  - Documents and queries as vectors; cosine similarity as proxy for relevance
  - TF-IDF, BM25
- Web search
  - Link structure as a large-scale voting system; PageRank
- Learning to rank (LTR)
  - Use behavioral data to learn a ranking function
- Neural-based ranking models
  - Use NNs to score or rank documents
- Representation learning
  - Encode queries and documents into vector representations
  - Retrieval using nearest neighbor search

# Quick terminology recap – Question Answering

- Classic systems
  - Limited large-scale success
  - Answers are list of snippets from documents or provided by a human
- NN approaches
  - Instead of ranking QA pairs, extract answer spans within passages
- Open-domain
  - Retrieve relevant passages -> machine reading comprehension -> answer
- Generative systems

# Information needs and queries

- Relevance to what?
- In Web
  - Two queries of two terms …
  - … looking each time to at most two pages and doing two clicks per page
  - Not a lot of data to guess correctly
- Relevance to the query
  - Problematic
  - Short queries
- Information need
- User intent

# Users and information needs

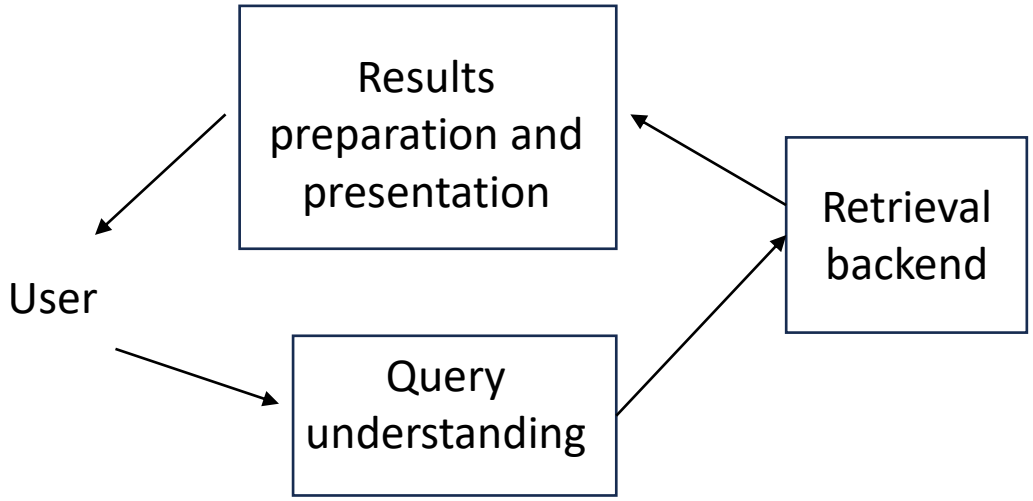
- Keyword queries
  - Free text queries
- Keyword++
  - Queries with filters or facets
- Natural language
  - Natural language queries, questions
- Zero queries
  - You are the query

# How do we know if users are happy?

- Search returns relevant results to users
  - How do you assess this at scale?
- Search results get clicked a lot
  - Misleading titles/summaries can cause users to click
- Users buy after using the search engine
  - Users spend a lot of $ after using the search engine
- Repeat visitors/buyers
  - Do users leave soon after searching?
  - Do they come back within a week/month/… ?

# Information seeking architectures

- At a high level, more or less the same
- Indexing and retrieval
- Query understanding
  - Normalization
  - Spelling correction
  - Segmentation
  - Annotation (NER, POS, etc.)
  - Term expansion
  - Query-rewriting
- Ranking
  - Many models to chose from
- Answer generation/snippets
- SERP construction
  - Web: 10-blue links
- ChatGPT-like
  - Answer



# This tutorial

- Lots of technology changes recently
- Are things really that new or just another iteration?
- It is very easy to build prototypes and test new solutions
- Have we solved all problems?
- IR, NLP, ML are converging
  - How can we combine all these new tech to solve new problems?
  - Where should we focus on?

# HEMU dimensions

- Computing resources
  - laptop < cloud access < cluster (1K machines is a typical industry cluster)
- Data
  - Small, medium, big
  - Public, sensitive
- Algorithmic complexity
  - Hashing/indexing, PageRank, Deep Learning
- Skills
  - Some things require the worlds' expert, and other things can be done by a software engineer
  - Things can be done by a non-programmer

# HEMU dimensions - II

- Size of team
  - You
  - Pizza team
  - The number of authors per paper has been growing suggesting that you need more and more people to do certain things
- Cost
  - Includes all of the above
  - As well as externalities such as power
- Value
  - Some things are super expensive and not worth doing
  - Think about time scales... some things are good for years/decades, and other things age quickly
  - How easy is it for the competition to do a fast follow?
- Organization
  - Is your team or company ready for your idea?
  - Immediate, medium, and long-term success