Title: Benefits of applying DCAL on bio-medicine SR versus the utilisation of the state-of-the-art CAL approach

Deep -> Based on ANN inspired by Biological NN with representation learning
Continuous -> ongoing process of reranking docs for manual review based on most recent qrels assessments
Active Learning -> selection of training documents is done algorithmically with no direction from the user apart from the supplied qrels when he reviewed docs

examples & benefits of DCAL over CAL: RNN

- DCAL instead of using standard ML models (in CAL) i.e. SVM/LightGBM which take input data in the simple form of BoW text representation use a Distributed representation to fight BoW & word2vec disadvantages

- Disadvantages: BoW
  - has high dimensional feature vector due to large size of vocabulary
  - doesn't leverage accompanied statistics between words (No Semantic Meaning)
- Example of DCAL based on ANN for language modelling: Basic **Recursive NN** with automatic differentiation -> data flows in any direction & training mainly with GSD.
- Advantages: DCAL
  - Deep RNN consider the morphological relation of words being modeled
  - The real utility of NN is realised when we have many data. In this case D/ANN outperform other ML models
- Deep learning systems (RNN) give each word a **distributed representation** (dense low dimensional real-valued vector)

Distributed Representation captures dimensions (*each dimension is a hidden feature of the word*) of semantic info in a vector, solid & less prone to data sparsity.

choosing a deep net

- for text processing, sentiment analysis, parsing & named entity recognition use a Recurrent net or a Recursive neural tensor network (RNTN)
- Recursive NN is generalisation of Recurrent NNs.

- Predictions made based on the move of the network through the layers calculating the probability of each output.
- For any language model that operates at character level use a recurrent/recursive net.

! deep belief networks is also a good choice for classification.

Examples & Benefits of DCAL over CAL: BERT

- DL architecture advances performance in NLP tasks with MS MARCO, MultiNLI datasets. Originates from pre-training contextual representations (Semi-supervised Sequence Learning).
- Output: an analysis of Internal Vector representations through insightful classifiers & relationships represented by attention weights.
- Deeply bidirectional, unsupervised language representation, pre-trained using only a plain text corpus for Next sentence prediction & masked word prediction using large datasets. Context-free models such as word2vec generate a single word embedding representation for each word in the vocabulary.
- **Transformer** an attention mechanism that learns dependent relations between text words. It has a necessary encoder for reading text input to generate a LM & a decoder producing task prediction
- Directional models read the text input sequentially left-to-right/right-to-left, the **Transformer** encoder reads the entire sequence of words at once. Model learns word meaning based on all surroundings (left & right of the word) Non-directional

BERT for Evidence Retrieval & Claim Verification - Investigation of BERT in an evidence retrieval & claim verification pipeline for the FEVER fact extraction & verification challenge

- The benchmark task is to classify the accuracy of claims & extract the correct evidence to support/refute claims
- 2 variations:
  - train the BERT retrieval system using pointwise & pairwise loss functions & examine the effect of hard negative mining (HNM)
  - train to classify the samples as supported/refuted/not enough information
- proposed system achieves new R record for retrieving top 5 sentences out of the FEVER documents consisting of 50K Wiki pages
- Input representation begins with a classification embedding ([CLS]) followed by the tokens representations of the 1st & 2nd sentences separated by specific token ([SEP]).
- To use BERT for different tasks, only 1 additional task-specific output layer is needed that can be trained together with fine-tuning the base layers.
- For the classification task a softmax layer is added on the last hidden state of the 1st token which is corresponding to [CLS]. BERT base model with 12 layers in all exper.

- **pointwise approach** inputs classified (evidence/non-evidence). Use of cross entropy classification Ranknet loss function. At testing time, sentences are sorted by softmax output & top 5 sentences are considered as evidence. Threshold on the output scores to filter out uncertain results with trading off R against P.
- **Pairwise approach** pair of positive & negative samples are compared against each other.
- Pointwise & Pairwise BERT sentence retrieval improve the R. Pairwise Ranknet with HNM has best R score. HNM for training retrieval systems improves performance slightly. Pointwise > Pairwise in R-P performance
- **Retrieval** Pairwise Ranknet achieves highest R. Pairwise methods are not superior to Pointwise approach considering P
- The large BERTs are only trained on the best retrieval systems significantly improving performance
- **Online HNM Investigation** The ratio of negative (non-evidence) to positive (evidence) sentences is high. Random sampling limits the number of negative samples but this might lead to training on trivial samples.
- Training epochs (E) increases from 1 to 3. Increasing E for the HNM improves the performance for non normal training
- Loss values are computed in the no-gradient mode like inference time (*for training I presume*)

Rethinking QE for BERT Reranking - Investigation of different sources of QE withf structure & concepts to improve queries for BERT-based rankers

- The added structure between keywords is ignored in most IR systems but it helps BERT achieve higher IR accuracy.

- 2 QE variations (combined benefits BERT reranking the most) by adding:
  - structural words that create a coherent NL sentence
  - additional terms to add new concepts to the query
- **QE with Structure** shows that non-concept stop-words improve BERT's effectiveness by building sentence structure to understand docs with similar content better
- Methods to use BERT's sensitivity to structure:
  - **Generated Structure** uses a neural machine translation method to generate synthetic questions from the original question. Copy keywords & adds few question words adding structure without new concepts

  - **Template Structure** tests the maximum possible range of benefit from adding structure to queries. Manually converting the keyword queries to a question with template. Queries can be reformulated into questions/description requests. All original keyword terms are included in the reformulation

- **Expansion with** related **Concepts** while grammar structures not considered. ClassicQEConcepts uses QE pseudo-RF RM3 to find related concepts.
- The expansion terms are concatenated to the original query & ordered by their RM3 scores
- Expands the query with both sentence structure & new concepts.
- Rely on scraping Google’s proposals for reformulated queries to acquire additional related questions.
- Verify/eliminate the power of additional structure & concepts with an oracle to filter manually suggested questions for query description match
- Robust04 dataset (249 queries & 0.5M docs)
- query types
  - Descriptions, containing long NLT describing the information need
  - Titles, the short keyword query text commonly used by search engine users
- **Measures** MAP, NDCG@, P@

- Baselines & Experimental Methods. Baselines include 3 BoW IR models using the Indri SE:
    - **Indri-LM** uses the query LM
    - **Indri-SDM** (uses Sequential Dependency Model)
    - **Indri-QE** uses the QE algorithm RM3, params selected after parameter sweep including number of RF documents, terms & weight of the original query

BERT-Title-Title & BERT-Desc-Desc were trained & evaluated on query titles & descriptions correspondingly.

Query titles replaced in the BERT reranker with QE by concatenating the QE to the original query title. Model & data setup of the passage-based BERT reranker same as BERT-MaxP. Splits docs into passages, estimates relevance between query & passage using BERT 2-sentence classification model & ranks docs using max passage scores.

Model settings B=16 & LR=1e−5 for 1000 iterations, max train depth = 1000 in the initial retrieved docs & only rerank top 100 docs from the initial ranking during test. Sample 10% of passages with overlap in addition to the 1st passage of each doc to prevent overfitting.

Other BERT rerankers use same BERT-MaxP strategy & apply domain adaptation & fusion with BM25 to improve accuracy. Generated Structure follows CopyNet to translate keywords to questions. Use of AllenNLP implementation of CopyNet using an embedding dimension of 100 (initialised to GloVe vectors) & an encoder/decoder of size 400 units. **Training dataset:** Wiki-Answers 3M pairs of questions paired with synthetically generated keyword question (ClassicQEConcepts) used the Indri implementation of RM3.

Future work: automatic identification of extensions (questions) that are in-domain to the source corpus & match with the original intent

BibTex REFERENCES from 2020 ECIR papers on BERT

[1] @inproceedings{padaki2020rethinking, title={Rethinking Query Expansion for BERT Reranking}, author={Padaki, Ramith & Dai, Zhuyun & Callan, Jamie}, booktitle={Advances in Information Retrieval: 42nd European Conference on IR Research, ECIR 2020, Lisbon, Portugal, April 14--17, 2020, Proceedings, Part II 42}, pages={297--304}, year={2020}, organization={Springer}}

[2] @article{soleimani2019bert, title={BERT for Evidence Retrieval & Claim Verification}, author={Soleimani, Amir & Monz, Christof & Worring, Marcel}, journal={arXiv preprint arXiv:1910.02655}, year={2019}}

the other 2 ECIR papers about BERT were focusing on examining Graphs and retrieval heuristics.