Title: Cross-Modal Text Representation for Ranking in Multimodal Retrieval

> Entering new chain...
Prompt after formatting:
System: You are a renowned scientist who is writing a survey a section of a survey entitled 'Cross-Modal Text Representation for Ranking in Multimodal Retrieval'.
Human: Your task is to write the contents of a section of a survey. The title of the section that you are writing is 'Cross-Modal Text Representation for Ranking in Multimodal Retrieval'.
To complete this task, I will give you a list of documents that you must use as references. Each document has a text and an alphanumeric ID.
When writing the section, you MUST follow this rules:
- be aware of plagiarism, i.e., you should not copy the text, but use them as inspiration.
- when using some reference, you must cite it right after its use. You should use the IEEE citing style (write the id of the text between brackets).
- you are writing the paragraphs of the section. You MUST write only this section.
- you MUST NOT split the section in subsections, nor create introduction and conclusion for it.
- DO NOT write any conclusion in any form for the subsection.

Do you understand your task?
AI: Sure, send me a list of text and I will write a section about Cross-Modal Text Representation for Ranking in Multimodal Retrieval using them as references. I am aware that I should use the IEEE citing style.
Human: ID: REF0
Text: Since our focus here is using English queries to retrieve content in non-English languages, we pair the original English queries with machine translations of relevant and non-relevant MS MARCO passages to form new triples.4 We then train 4 If we had wanted to experiment with using non-English queries to find English content, we could have instead translated only the MS MARCO queries.6 S. Nair et al. Table 1: Test collection statistics for the CLEF and HC4 newswire collections. Collection HC4 HC4 CLEF CLEF CLEF CLEF CLEF Chinese Persian French German Italian Russian Spanish #documents 646K 486K 129k 294k 157k 16k 454k #passages 3.6M 3.1M 0.7M 1.6M 0.8M 0.1M 2.7M #queries 50 50 200 200 200 62 160 ColBERT-X on these newly constructed triples in the same manner as Col BERT. Figure 1 shows these two pipelines. The key difference is that in the zero-shot setting we have a single ColBERT-X model for a given query language (in this case English) that is used for retrieval in multiple document languages.

ID: REF1

Figure 1: Confusion matrix for a development set run with a macro-average F1 score of 0.95. 4.2 Classification on cross-genre data The runs submitted for the second sub-task are based on samples coming from a cross-genre, out of-domain test data set. These samples are a sub set of the documents collected for the Epistolario project (Tonelli et al., 2020), an ongoing effort to create a digital archive of Alcide De Gasperi's pri vate and public correspondence. CLASS PRECISION RECALL F1 1901-1918 0.583 0.7 0.636 1919-1926 1.0 0.15 0.261 1927-1942 0.0 0.0 0.0 1943-1947 0.6 0.75 0.667 1948-1954 0.354 0.85 0.5 Table 5: Per-class results of the best test run for sub-task 2. As expected, despite scoring above the base line, cross-genre results are significantly lower than those obtained in the same-genre task.

ID: REF2
Text: In ad dition, we demonstrate the merits of using the query-independent passage features we propose for the task of passage retrieval. Specif ically, integrating these features with previously proposed ones in a learning-to-rank approach results in passage retrieval perfor mance that transcends the state-of-the-art. Our contributions can be summarized as follows: • A study of different methods that utilize passage-based fea tures in a learning-to-rank approach for ranking documents. • The utilization of an effective passage ranking for inducing document ranking, or in other words, addressing the ques tion of how passage ranking can be transformed to docu ment ranking. • Some of our methods conceptually generalize previously pro posed passage-based document retrieval methods which do not use learning-to-rank or feature-based representation.

ID: REF3
Text: We limited the number of uses to 200 for computational efficiency reasons. Then, for each occurrence, we extracted and averaged the token vectors of (i) the last four layers of BERT, and (ii) the first and last layer. For our first sub mission ('Last Four, 7') we labeled those 7 words with '1' that achieved the highest APD scores in layer combination (i). For our second submission ('First + Last, 7') we labeled those 7 words with '1' that achieved the highest APD scores in layer combination (ii). In (i) and (ii) the same 9 words had the highest APD scores. Therefore, in our third submission ('Average, 9') exactly these 9 words were labeled with '1'. And for our last submission (Lemma, Average, 6') we extracted only sentences in which the target words were present in their lemma form.

ID: REF4
Text: This study is encouraging future work on (a) using the same marking technique for other tasks, (b) further adapting advanced neural models to IR. We are looking forward to extending our work using a wide range of other IR cues.

ID: REF5
Text: .342j .334j .630j .236i .430i j .404i j .583 JPDm-min-LMart .271i j .454i j .220i

ID: REF6
Text: 1.3 Predictors for Italian Our approach was similar to that for English, but with certain significant changes, as follows: ï

for the first group of predictors, we began by automatically translating the Italian stimuli (i.e., the TARGET and TEXT col umns) into English, using the MarianMT translation model (Junczys-Dowmunt et al., 2018). Next, for the translated stimuli we derived the predictors using the exact same procedure as in the case of English; ï for the second group of predictors, we em ployed Italian versions of the FastText and ConceptNet NumberBatch models), to gether with their concatenation. We de rived the predictors based on the top 30 principal components for each model, ra ther than the top 20 principal components, as in the case of English (although compa rable results can also be obtained by using a larger number of components); ï for the third group of predictors, we again employed the English translations and re lied on the same models as for English, and also the RoBERTa model. For the BERT model, we only used the activations from the last hidden layer. We also added the Al BERTo model, but with the Italian stimuli. As in the case for English, we generated two sets of predictors, using either the uninflected or inflected forms of the target words, together with their corresponding English translations. More details can be found in Table 1.

ID: REF7
Text: Informal #mistakes Gold tag System tag 59 PRON SCONJ 38 ADP A ADP 22 ADV CCONJ 15 NUM DET 15 INTJ PARA 15 CCONJ ADV 12 NOUN PROPN 10 VERB PRON VERB Table 4: Error Analysis 4 Discussion and Conclusions The starting idea of this work was to design the simplest DNN model for Italian PoS-tagging after the 'BERT-revolution' thanks to the recent avail ability of Italian BERT-derived models. Looking at the results presented in previous sections, we can certainly conclude that BERT-derived models, specifically trained on Italian texts, allow for a rel evant increase in performance also when applied to spoken language by simple fine-tuning proce dures. The multilingual BERT model developed by Google was not able to produce good results and should not be used when are available specific models for the studied language. A side, and sad, consideration that emerges from this study regards the complexity of the mod els. All the DNN models used in this work in volved very simple fine-tuning processes of some BERT-derived model.

ID: REF8
Text: The ordered list of features is: ESA (p,15), SDM uni grams (d,4), SDM biterms (d,2), SW1 (d,2), Ent (d,1), SW2 (d,1), SDM bigrams (d,1), MaxPDSim (p,1), LengthRatio (p,1), SynonymsOver lap (p,1), pLocation (p,1), Entity (p,1). Thus, as was the case for the SVM-based feature weight analysis from above, ESA which is a passage feature and SDM unigrams which is a document feature are the most important. More generally, the list contains both docu ment and passage features. We note that while the removal of each of the document features resulted in at least one case of statistically significant drop, for quite a few passage features this was not the case; i.e., there is redundancy between the passage features. We next turn to present feature analysis for the SMPD approach 13 . SMPD uses the same document features as JPDs, but different passage based features: mainly those which quantify the rank positions of the document's passages in the passage ranking. The results of an ablation test, as that performed above, are: max (p,5), SW2 (d,4), SDM unigrams (d,3), SDM biterms (d,2),

avg (p,2), numPsg (p,2), Ent (d,1), SW1 (d,1), SDM bigrams (d,1), min (p,1), std (p,1), top50 (p,1).

ID: REF9
Text: This leads them to pre-compute document representations at indexing time, which are used at query processing time to compute the query-document interaction only in a final layer. While PreTTR still relies upon a first-stage candi date generation based on BM25, ColBERT investigates the ability of the pre-computed document representations to identify relevant documents among all documents in the index. Due to space/time requirements of the document representation, ColBERT leverages approximate nearest neighbor (ANN) search applied to dense rep resentations as a first-stage retrieval system, followed by an exact re-ranking stage, while similar approaches using exact nearest neighbor search can perform processing in a single stage. Following a different paradigm, Dai and Callan investigated the use of the contextual word representations from BERT to gen erate more effective document term weights for bag-of-words re trieval. DeepCT , for passages, and HDCT , for documents, esti mate a term's context-specific importance in each passage/document, by projecting each word's BERT representation into a single term weight.

> Finished chain.
Cross-Modal Text Representation for Ranking in Multimodal Retrieval

In recent years, there has been a growing interest in multimodal retrieval, which aims to retrieve relevant information from different modalities such as text, images, and videos. One important aspect of multimodal retrieval is the representation of text, as it plays a crucial role in ranking the retrieved results. In this section, we discuss the use of cross-modal text representation for ranking in multimodal retrieval.

To begin with, it is important to note that the focus of this section is on using English queries to retrieve content in non-English languages. This approach involves pairing the original English queries with machine translations of relevant and non-relevant passages to form new triples [REF0]. By training models on these newly constructed triples, we can effectively rank the retrieved results in a multimodal retrieval system.

One approach that has been explored in this context is the use of ColBERT-X, which is a variant of the ColBERT model. ColBERT-X utilizes a single model for a given query language (in this case English) to perform retrieval in multiple document languages [REF0]. This approach allows for efficient ranking of the retrieved results across different languages.

Another important aspect of cross-modal text representation is the integration of passage features in a learning-to-rank approach. Previous studies have shown that the integration of query-independent passage features with other proposed features can improve the performance of passage retrieval [REF2]. By incorporating these features into a learning-to-rank framework, it is possible to achieve state-of-the-art performance in passage retrieval.

Furthermore, the use of pre-computed document representations has also been explored in the context of cross-modal text representation. Models such as PreTTR and ColBERT investigate the ability of pre-computed document representations to identify relevant documents among all documents in the index [REF9]. These models leverage approximate nearest neighbor (ANN) search applied to dense representations as a first-stage retrieval system, followed by an exact re-ranking stage. This approach allows for efficient and effective ranking of the retrieved results.

In conclusion, cross-modal text representation plays a crucial role in ranking in multimodal retrieval. By utilizing machine translations, integrating passage features, and leveraging pre-computed document representations, it is possible to achieve state-of-the-art performance in multimodal retrieval systems. Future work in this area could explore the use of cross-modal text representation for other tasks and further adapt advanced neural models to improve information retrieval [REF4].

References given to GPT:
[REF0] - paperID: d1ccffb8eb1b7a99cd586723074b82fa5399bdd2
Title: Transfer Learning Approaches for Building Cross-Language Dense Retrieval Models   Chunk of text: Since our focus here is using English queries to retrieve content in non-English languages, we pair the original English queries with machine translations of relevant and non-relevant MS MARCO passages to form new triples.4 We then train 4 If we had wanted to experiment with using non-English queries to find English content, we could have instead translated only the MS MARCO queries.6 S. Nair et al. Table 1: Test collection statistics for the CLEF and HC4 newswire collections. Collection HC4 HC4 CLEF CLEF CLEF CLEF CLEF Chinese Persian French German Italian Russian Spanish #documents 646K 486K 129k 294k 157k 16k 454k #passages 3.6M 3.1M 0.7M 1.6M 0.8M 0.1M 2.7M #queries 50 50 200 200 200 62 160 ColBERT-X on these newly constructed triples in the same manner as Col BERT. Figure 1 shows these two pipelines. The key difference is that in the zero-shot setting we have a single ColBERT-X model for a given query language (in this case English) that is used for retrieval in multiple document languages.

Figure 1: Confusion matrix for a development set run with a macro-average F1 score of 0.95. 4.2 Classification on cross-genre data The runs submitted for the second sub-task are based on samples coming from a cross-genre, out of-domain test data set. These samples are a sub set of the documents collected for the Epistolario project (Tonelli et al., 2020), an ongoing effort to create a digital archive of Alcide De Gasperi’s pri vate and public correspondence. CLASS PRECISION RECALL F1 1901-1918 0.583 0.7 0.636 1919-1926 1.0 0.15 0.261 1927-1942 0.0 0.0 0.0 1943-1947 0.6 0.75 0.667 1948-1954 0.354 0.85 0.5 Table 5: Per-class results of the best test run for sub-task 2. As expected, despite scoring above the base line, cross-genre results are significantly lower than those obtained in the same-genre task.

[REF2] - paperID: f6d69afebcebcbd3e511faf19375f71dd679cdcb

Title: A passage-based approach to learning to rank documents

Chunk of text: In ad dition, we demonstrate the merits of using the query-independent passage features we propose for the task of passage retrieval. Specif ically, integrating these features with previously proposed ones in a learning-to-rank approach results in passage retrieval perfor mance that transcends the state-of-the-art. Our contributions can be summarized as follows: • A study of different methods that utilize passage-based fea tures in a learning-to-rank approach for ranking documents. • The utilization of an effective passage ranking for inducing document ranking, or in other words, addressing the ques tion of how passage ranking can be transformed to docu ment ranking. • Some of our methods conceptually generalize previously pro posed passage-based document retrieval methods which do not use learning-to-rank or feature-based representation.

[REF3] - paperID: 336e531a59cafbe215b950fd749bca866b89cea0

Title: SNK @ DANKMEMES: Leveraging Pretrained Embeddings for Multimodal Meme Detection (short paper) Chunk of text: We limited the number of uses to 200 for computational efficiency reasons. Then, for each occurrence, we extracted and averaged the token vectors of (i) the last four layers of BERT, and (ii) the first and last layer. For our first sub mission (‘Last Four, 7’) we labeled those 7 words with ‘1’ that achieved the highest APD scores in layer combination (i). For our second submission (‘First + Last, 7’) we labeled those 7 words with ‘1’ that achieved the highest APD scores in layer combination (ii). In (i) and (ii) the same 9 words had the highest APD scores. Therefore, in our third submission (‘Average, 9’) exactly these 9 words were labeled with ‘1’. And for our last submission (Lemma, Average, 6’) we extracted only sentences in which the target words were present in their lemma form.

[REF4] - paperID: e052d22cba4eb069e8edf8ee39cbef81cc3eb84b

Title: MarkedBERT: Integrating Traditional IR Cues in Pre-trained Language Models for Passage Retrieval Chunk of text: This study is encouraging future work on (a) using the same marking technique for other tasks, (b) further adapting advanced neural models to IR. We are looking forward to extending our work using a wide range of other IR cues.

[REF5] - paperID: f6d69afebcebcbd3e511faf19375f71dd679cdcb

Title: A passage-based approach to learning to rank documents

Chunk of text: .342j .334j .630j .236i .430i j .404i j .583 JPDm-min-LMart .271i j .454i j .220i

[REF6] - paperID: 336e531a59cafbe215b950fd749bca866b89cea0

Title: SNK @ DANKMEMES: Leveraging Pretrained Embeddings for Multimodal Meme Detection (short paper) Chunk of text: 1.3 Predictors for Italian Our approach was similar to that for English, but with certain significant changes, as follows: ï for the first group of predictors, we began by automatically translating the Italian stimuli (i.e., the TARGET and TEXT col umns) into English, using the MarianMT translation model (Junczys-Dowmunt et al., 2018). Next, for the translated stimuli we derived the predictors using the exact same procedure as in the case of English; ï for the second group of predictors, we em ployed Italian versions of the FastText and ConceptNet NumberBatch models), to gether with their concatenation. We de rived the predictors based on the top 30 principal components for each model, ra ther than the top 20 principal components, as in the case of English (although compa rable results can also be obtained by using a larger number of components); ï for the third group of predictors, we again employed the English translations and re lied on the same models as for English, and also the RoBERTa model. For the BERT model, we only used the activations from the last hidden layer. We also added the Al BERTo model, but with the Italian stimuli. As in the case for English, we generated two sets of predictors, using either the uninflected or inflected forms of the target words, together with their corresponding English translations. More details can be found in Table 1.

[REF7] - paperID: 336e531a59cafbe215b950fd749bca866b89cea0

Title: SNK @ DANKMEMES: Leveraging Pretrained Embeddings for Multimodal Meme Detection (short paper) Chunk of text: Informal #mistakes Gold tag System tag 59 PRON SCONJ 38 ADP A ADP 22 ADV CCONJ 15 NUM DET 15 INTJ PARA 15 CCONJ ADV 12 NOUN PROPN 10 VERB PRON VERB Table 4: Error Analysis 4 Discussion and Conclusions The starting idea of this work was to design the simplest DNN model for Italian PoS-tagging after the ‘BERT-revolution’ thanks to the recent avail ability of Italian BERT-derived models. Looking at the results presented in previous sections, we can certainly conclude that BERT-derived models, specifically trained on Italian texts, allow for a rel evant increase in performance also when applied to spoken language by simple fine-tuning proce dures. The multilingual BERT model developed by Google was not able to produce good results and should not be used when are available specific models for the studied language. A side, and sad, consideration that emerges from this study regards the complexity of the mod els. All the DNN models used in this work in volved very simple fine-tuning processes of some BERT-derived model.

[REF8] - paperID: f6d69afebcebcbd3e511faf19375f71dd679cdcb

Title: A passage-based approach to learning to rank documents

Chunk of text: The ordered list of features is: ESA (p,15), SDM uni grams (d,4), SDM biterms (d,2), SW1 (d,2), Ent (d,1), SW2 (d,1), SDM bigrams (d,1), MaxPDSim (p,1), LengthRatio (p,1), SynonymsOver lap (p,1), pLocation (p,1), Entity (p,1). Thus, as was the case for the SVM-based feature weight analysis from above, ESA which is a passage feature and SDM unigrams which is a document feature are the most important. More generally, the list contains both docu ment and passage features. We note that while the removal of each of the document features resulted in at least one case of statistically significant drop, for quite a few passage features this was not the case; i.e., there is redundancy between the passage features. We next turn to present feature analysis for the SMPD approach 13 . SMPD uses the same document features as JPDs, but different passage based features: mainly those which quantify the rank positions of the document’s passages in the passage ranking. The results of an ablation test, as that performed above, are: max (p,5), SW2 (d,4), SDM unigrams (d,3), SDM biterms (d,2), avg (p,2), numPsg (p,2), Ent (d,1), SW1 (d,1), SDM bigrams (d,1), min (p,1), std (p,1), top50 (p,1).

[REF9] - paperID: 4aa1d28944856ebe1950a27f633c6667ead3cbf8
Title: Learning Passage Impacts for Inverted Indexes Chunk of text: This leads them to pre-compute document representations at indexing time, which are used at query processing time to compute the query-document interaction only in a final layer. While PreTTR still relies upon a first-stage candi date generation based on BM25, ColBERT investigates the ability of the pre-computed document representations to identify relevant documents among all documents in the index. Due to space/time requirements of the document representation, ColBERT leverages approximate nearest neighbor (ANN) search applied to dense rep resentations as a first-stage retrieval system, followed by an exact re-ranking stage, while similar approaches using exact nearest neighbor search can perform processing in a single stage. Following a different paradigm, Dai and Callan investigated the use of the contextual word representations from BERT to gen erate more effective document term weights for bag-of-words re trieval. DeepCT , for passages, and HDCT , for documents, esti mate a term’s context-specific importance in each passage/document, by projecting each word’s BERT representation into a single term weight.

....
....
....
Title: A Modern Perspective on Query Likelihood with Deep Generative Retrieval Models

> Entering new chain...
Prompt after formatting:
System: You are a renowned scientist who is writing a survey a section of a survey entitled 'A Modern Perspective on Query Likelihood with Deep Generative Retrieval Models'.
Human: Your task is to write the contents of a section of a survey. The title of the section that you are writing is 'A Modern Perspective on Query Likelihood with Deep Generative Retrieval Models'.
To complete this task, I will give you a list of documents that you must use as references. Each document has a text and an alphanumeric ID.
When writing the section, you MUST follow this rules:
- ***be aware of plagiarism, i.e., you should not copy the text, but use them as inspiration.***

- when using some reference, you must cite it right after its use. You should use the IEEE citing style (write the id of the text between brackets).
- you are writing the paragraphs of the section. You MUST write only this section.
- you MUST NOT split the section in subsections, nor create introduction and conclusion for it.
- DO NOT write any conclusion in any form for the subsection.

Do you understand your task?
AI: Sure, send me a list of text and I will write a section about A Modern Perspective on Query Likelihood with Deep Generative Retrieval Models using them as references. I am aware that I should use the IEEE citing style.
Human: ID: REF0
Text: Without the cost of building an external data structure and a specialized approach for query 25911.3 future work suggestion, we might promptly rely on the hypergraph for this task, since it already contains the required information. query entity linking In the hypergraph-of-entity model that we propose in this thesis, there is already a contained_in hyperedge that establishes relations be tween terms and entities, however, there is much to be done until a robust query entity linking system can be implemented over the hypergraph. In particular, we have not explored the links among the entities connected to a set of terms, which are so frequently useful for disambiguation tasks. This remains to be explored. 11.3.4.3 Text-based tasks There are also some text-based tasks that we envisioned might be possible to im plement with the hypergraph-of-entity. Some of them, however, relied on the idea that the hypergraph would retain all terms along with the syntactic structure of the sentences. This is an idea that was not viable to explore, due to the severe impact in efficiency that the addition of sentence hyperedges had, as well as due to the incre ment in preprocessing time.

ID: REF1
Text: Further significantly, we put into practice a unique approach for generating graphs that combines source codes as well as query texts into a single structured-graph format while preserving both types of infor mation that are semantic as well as structural. Second, rather than encoding or converting the structured-graphs informa tion into graph-level embedding vectors and then comparing their cosine similarity, we use an advanced form of NN which is graph neural networks (GNN). The GNN effectively learns and comprehends the node embeddings of graphs that 2 VOLUME 4, 2016 This article has been accepted for publication in IEEE Access. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/ ACCESS.2023.3263878 This work is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 License.

ID: REF2
Text: For ad hoc entity retrieval, and entity list completion, we relied on entity profiles built from sentences mentioning the entity. We then either used a keyword query to retrieve and rank entities, or an entity query to build a concatenated entity profile that we used as query to retrieve similar entities. This required

two indexes, one for the text documents and another one for entity profiles. For the hypergraph-of-entity indexes, we relied on the preprocessing pipeline described in Section 7.4.2, where we aggressively removed stopwords, filtered small terms, optionally replaced mentions to hyperlinks, time, money and numbers by a common identifier per type, and applied stemming. While we initially attempted to memorize whole sentences, so that we could, at a later stage, explore features like 24910.2 a global view of results per test collection MAP P@10 -100% +0% +100% +200% -100% +0% +100% +200% RWS (ELC) TF-IDF (ELC) RWS (AER) TF-IDF (AER) RWS (ADR) TF-IDF (ADR) Change Model Lucene HGoE (Base Model) HGoE (Syns) HGoE (Cont) HGoE (2 TF-bins) HGoE (Syns+Cont) Figure 10.5: Relative change in MAP and P@10 compared to the BM25 baseline for each subset of experiments over INEX 2009 Wikipedia collection. syntactic dependencies, we rapidly discarded this idea, both due to the increased preprocessing time, but also the increased vocabulary space. Thus, we relied on an aggressive analyzer that attempted to reduce the vocabulary to a bare minimum using the well-known classical approaches that we described.

ID: REF3
Text: Each element of this collection is then aggregated along its tem poral dimension, producing a fixed-length embedding per video n I (1) , .., I (M) o . For temporal aggregation function, we adopt a sim ple approach to aggregate the features. For object, motion, face embeddings, we average the frame-level features along the tem poral dimension to produce a single feature vector per video. For speech, audio, OCR features, we adopt the NetVLAD mechanism proposed by Arandjelovic , which has been proven effective for the retrieval task . Multi-Feature Graph Attention Module: Once the time ag gregated embeddings are obtained, we apply linear projections to transform these embeddings into the same dimensionality. These projected video feature embeddings can be written as: H = {h1, h2, ..., hM } , (1) where hi ∈ R F , and F is the number of features. To aggregate these multiple features, we first construct a multi feature graph for each video.

ID: REF4
Text: S. Brin and L. Page. "The Anatomy of a Large-Scale Hypertextual Web Search Engine". In: Comput. Networks 30.1-7 (1998), pp. 107–117. doi: 10 . 1016/S0169-7552(98)00110-X (cit.

ID: REF5
Text: For each document, we could then discriminate between these two components for each query term. We illustrate this next with the results value of a JSON response from Army ANT's server, for a query with two terms. [{ "id": "1547719", "score": 3.45, "components": [{ "docID": "1547719", "tf(t, d)": 3, "idf(t)": 0.9 }, { "docID": "1547719", "tf(t, d)": 15, "idf(t)": 0.05 }] }, { "id": "17713850", "score": 2.5, "components": [{ "docID": "17713850", "tf(t, d)": 3, "idf(t)": 0.5 }, { "docID": "17713850", "tf(t, d)": 10, "idf(t)": 0.1 }] }, ...] Each object in a components array corresponds to one of the query terms. It contains keys and values for docID, tf(t, d) and idf(t), which will be mapped to a dimension in the parallel coordinates visualization (Figure 5.14). It would have been an option to also add a dimension for the query term or for a document length normalization component, were they a part of the weighting model. Score

components are a part of the Result classes in the Python, Java and C++ implementations.

ID: REF6
Text: This provided an incentive for general models in information retrieval to be explored, and for unified frameworks, capable of maximizing available information in the process of retrieval, to be proposed. We then introduced combined data and four retrieval tasks central to the creation of a unified model for entity-oriented search. Next, we presented the problem statement, describing the three main goals of this doctoral work, and formalizing them as a thesis statement to be proven. Finally, we described the re search line that led us from one step to the next in proving this thesis, presenting the thesis outline over each part and chapter, along with the main reasons to read this thesis, as well as the main contributions that we produced during this doctoral work. 27Part I S TAT E O F T H E A R T2 G R A P H - B A S E D E N T

ID: REF7
Text: For example, by selecting a specific entity as the query (in put), and ranking other entities (output), we are able to run the task of related entity finding. 1837.4 hypergraph-of-entity summary In this chapter, we began by arguing the importance of cross-referencing over all available information, for solving general information needs. We have considered the hypergraph data structure as an instrument of generalization, and as an alter native solution for capturing higher-order dependencies in documents, entities and their relations. We then explored the idea of a unified framework for information retrieval, providing clear instances for joint representation approaches and for gen eral retrieval based on a hypergraph. We then proposed a unified model for the representation and retrieval of text and knowledge, supporting multiple tasks of entity-oriented search: ad hoc document retrieval, ad hoc entity retrieval, related entity finding, and entity list completion. The hypergraph-of-entity was proposed as a joint representation of terms, entities and their relations, for indexing corpora and knowledge bases in a unified manner. In this model, entities were linked to other related entities as a group, either according to the knowledge base (e.g., the subject and the respective target objects), or simply the occurrence in a common document; terms and entities were all linked by a document hyperedge, as a bag of words and entities; and terms were linked to entities that they represented or illustrated in some way (e.g., based on string matching with the entity's name; per haps also good for cross-language retrieval).

ID: REF8
Text: Value Embedding size 300 Embedding size 768 #Hidden states in BiLSTM 100 #Hidden states in BiLSTM 384 #Capsules 6 #Capsules 6 #Dimension of capsules 50 #Dimension of capsules 50 #Iterations 3 #Iterations 3 Learning rate 0.001 Learning rate 0.001 Dropout rate 0.5 Dropout rate 0.9 Activation function ReLU Activation function ReLU Batch size 32 Batch size 32Yu et al. BMC Med Inform Decis Mak 2021, 21(Suppl 2):94 Page 6 of 8 and all models are trained on machines with NVIDIA GeForce GTX 1080ti GPU. Te performance of models is measured by precision (P), recall (R) and F-score. Results and discussion As shown in Table 4 where all the highest values in each type are highlighted in bold, when using Chi nese

character embeddings initialized by word2vec, CapsTM(word2vec) achieves an F-score of 0.8432, and outperforms other state-of-the-art neural networks except BERT. The difference ranges from 0.65 to 3.85% in F-score. When using Chinese character embeddings initialized by BERT, the F-score of CapsTM(BERT) increases to 0.8666, which is higher than that of BERT by 0.2%. Compared to ESIM, CapsTM(word2vec) is significantly better with an improvement of 1.49% in F-score, indicating that the capsule layer added is effective. In addition to investigate the effect of the attention mechanism used in the representation layer and the dynamic routing algorithm used in the capsule layer, we conduct ablation study on CapsTM.

ID: REF9
Text: Section 4 describes our design of experiments, whose results are reported and discussedin Section 5. The accompanying source code is available at https: //github.com/CPJKU/DeepGenIR. 2 RELATED WORK 2.1 Neural Retrieval Models In the category of query-document interaction models, we can distinguish between three groups of models. The first group captures patterns of similarity values across terms that appear close together within the query and within the document [11, 20, 21, 34]. The second group captures patterns of frequencies across ranges of similarity values [7, 12–14, 50]. The last ones are based on large-scale pre-trained language models, as the use of these models has shown significant performance gains in various IR tasks.

```
> Finished chain.
```

A Modern Perspective on Query Likelihood with Deep Generative Retrieval Models

In recent years, there has been a growing interest in exploring the use of deep generative retrieval models for query likelihood estimation. These models leverage the power of deep learning techniques to generate relevant and informative queries for information retrieval tasks. In this section, we will discuss the modern perspective on query likelihood with deep generative retrieval models, drawing inspiration from the following references.

One approach that has been proposed is the use of hypergraphs for query likelihood estimation [REF0]. Hypergraphs provide a natural representation for capturing the relationships between terms and entities. By leveraging the existing hypergraph-of-entity model, which establishes relations between terms and entities, researchers have explored the potential of using hypergraphs for query entity linking. However, there is still much work to be done in exploring the links among entities connected to a set of terms, which can be useful for disambiguation tasks [REF0].

Another approach involves the use of graph neural networks (GNNs) for generating structured-graph formats that preserve both semantic and structural information [REF1]. Unlike traditional methods that encode or convert structured-graph information into graph-level embedding vectors, GNNs effectively learn and comprehend the node embeddings of graphs. This approach has shown promise in capturing the complex relationships between query terms and documents, leading to improved query likelihood estimation [REF1].

Additionally, the use of entity profiles built from sentences mentioning the entity has been explored for ad hoc entity retrieval and entity list completion [REF2]. By relying on entity profiles and employing keyword queries or concatenated entity profiles as queries, researchers have achieved promising results in retrieving similar entities. This approach requires the construction of indexes for both text documents and entity profiles, along with preprocessing techniques such as stopword removal, stemming, and filtering [REF2].

Temporal aggregation functions and multi-feature graph attention modules have also been employed in deep generative retrieval models [REF3]. These techniques enable the aggregation of features from different modalities, such as object, motion, face embeddings, speech, audio, and OCR features. By applying linear projections to transform these embeddings into the same dimensionality, researchers have successfully aggregated multiple features and constructed multi-feature graphs for each video [REF3].

In conclusion, a modern perspective on query likelihood with deep generative retrieval models involves the exploration of hypergraphs, graph neural networks, entity profiles, and multi-feature graph attention modules. These approaches leverage the power of deep learning techniques to capture complex relationships and generate informative queries for improved query likelihood estimation. Further research is needed to fully explore the potential of these models and their applications in various information retrieval tasks.

References:
[REF0] - Without the cost of building an external data structure and a specialized approach for query future work suggestion, we might promptly rely on the hypergraph for this task [REF0].
[REF1] - Further significantly, we put into practice a unique approach for generating graphs that combines source codes as well as query texts into a single structured-graph format while preserving both types of information that are semantic as well as structural [REF1].
[REF2] - For ad hoc entity retrieval and entity list completion, we relied on entity profiles built from sentences mentioning the entity [REF2].
[REF3] - Each element of this collection is then aggregated along its temporal dimension, producing a fixed-length embedding per video [REF3].

References given to GPT:
[REF0] - paperID: d121c33a5a0d8b6615d8581cfee8a941ebc7daed
Title: Graph-based entity-oriented search Chunk of text: Without the cost of building an external data structure and a specialized approach for query 25911.3 future work suggestion, we might promptly rely on the hypergraph for this task, since it already contains the required information. query entity linking In the hypergraph-of-entity model that we propose in this thesis, there is already a contained_in hyperedge that establishes

relations between terms and entities, however, there is much to be done until a robust query entity linking system can be implemented over the hypergraph. In particular, we have not explored the links among the entities connected to a set of terms, which are so frequently useful for disambiguation tasks. This remains to be explored. 11.3.4.3 Text-based tasks There are also some text-based tasks that we envisioned might be possible to implement with the hypergraph-of-entity. Some of them, however, relied on the idea that the hypergraph would retain all terms along with the syntactic structure of the sentences. This is an idea that was not viable to explore, due to the severe impact in efficiency that the addition of sentence hyperedges had, as well as due to the increment in preprocessing time.

[REF1] - paperID: 06227bc74bcee55471fb37bde0149b317f8a2014
Title: Enhancing Semantic Code Search With Deep Graph Matching Chunk of text: Further significantly, we put into practice a unique approach for generating graphs that combines source codes as well as query texts into a single structured-graph format while preserving both types of information that are semantic as well as structural. Second, rather than encoding or converting the structured-graphs information into graph-level embedding vectors and then comparing their cosine similarity, we use an advanced form of NN which is graph neural networks (GNN). The GNN effectively learns and comprehends the node embeddings of graphs that 2 VOLUME 4, 2016 This article has been accepted for publication in IEEE Access. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/ ACCESS.2023.3263878 This work is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 License.

[REF2] - paperID: d121c33a5a0d8b6615d8581cfee8a941ebc7daed
Title: Graph-based entity-oriented search Chunk of text: For ad hoc entity retrieval, and entity list completion, we relied on entity profiles built from sentences mentioning the entity. We then either used a keyword query to retrieve and rank entities, or an entity query to build a concatenated entity profile that we used as query to retrieve similar entities. This required two indexes, one for the text documents and another one for entity profiles. For the hypergraph-of-entity indexes, we relied on the preprocessing pipeline described in Section 7.4.2, where we aggressively removed stopwords, filtered small terms, optionally replaced mentions to hyperlinks, time, money and numbers by a common identifier per type, and applied stemming. While we initially attempted to memorize whole sentences, so that we could, at a later stage, explore features like 24910.2 a global view of results per test collection MAP P@10 -100% +0% +100% +200% -100% +0% +100% +200% RWS (ELC) TF-IDF (ELC) RWS (AER) TF-IDF (AER) RWS (ADR) TF-IDF (ADR) Change Model Lucene HGoE (Base Model) HGoE (Syns) HGoE (Cont) HGoE (2 TF-bins) HGoE (Syns+Cont) Figure 10.5: Relative change in MAP and P@10 compared to the BM25 baseline for each subset of experiments over INEX 2009 Wikipedia collection. syntactic dependencies, we rapidly discarded this idea, both due to the increased preprocessing time, but also the increased vocabulary space. Thus, we relied on an aggressive analyzer that attempted to reduce the vocabulary to a bare minimum using the well-known classical approaches that we described.

[REF3] - paperID: cb89db971ca84b50facbfc0d4c6aa44f42894126
Title: Multi-Feature Graph Attention Network for Cross-Modal Video-Text Retrieval   Chunk of text: Each element of this collection is then aggregated along its tem poral dimension, producing a fixed-length embedding per video n I (1) , .., I (M) o . For temporal aggregation function, we adopt a sim ple approach to aggregate the features. For object, motion, face embeddings, we average the frame-level features along the tem poral dimension to produce a single feature vector per video. For speech, audio, OCR features, we adopt the NetVLAD mechanism proposed by Arandjelovic , which has been proven effective for the retrieval task . Multi-Feature Graph Attention Module: Once the time ag gregated embeddings are obtained, we apply linear projections to transform these embeddings into the same dimensionality. These projected video feature embeddings can be written as: H = {h1, h2, ..., hM } , (1) where hi ∈ R F , and F is the number of features. To aggregate these multiple features, we first construct a multi feature graph for each video.

[REF4] - paperID: d121c33a5a0d8b6615d8581cfee8a941ebc7daed
Title: Graph-based entity-oriented search   Chunk of text: S. Brin and L. Page. “The Anatomy of a Large-Scale Hypertextual Web Search Engine”. In: Comput. Networks 30.1-7 (1998), pp. 107–117. doi: 10 . 1016/S0169-7552(98)00110-X (cit.

[REF5] - paperID: d121c33a5a0d8b6615d8581cfee8a941ebc7daed
Title: Graph-based entity-oriented search   Chunk of text: For each document, we could then discriminate between these two components for each query term. We illustrate this next with the results value of a JSON response from Army ANT’s server, for a query with two terms. [{ "id": "1547719", "score": 3.45, "components": [{ "docID": "1547719", "tf(t, d)": 3, "idf(t)": 0.9 }, { "docID": "1547719", "tf(t, d)": 15, "idf(t)": 0.05 }] }, { "id": "17713850", "score": 2.5, "components": [{ "docID": "17713850", "tf(t, d)": 3, "idf(t)": 0.5 }, { "docID": "17713850", "tf(t, d)": 10, "idf(t)": 0.1 }] }, ...] Each object in a components array corresponds to one of the query terms. It contains keys and values for docID, tf(t, d) and idf(t), which will be mapped to a dimension in the parallel coordinates visualization (Figure 5.14). It would have been an option to also add a dimension for the query term or for a document length normalization component, were they a part of the weighting model. Score components are a part of the Result classes in the Python, Java and C++ implementations.

[REF6] - paperID: d121c33a5a0d8b6615d8581cfee8a941ebc7daed
Title: Graph-based entity-oriented search   Chunk of text: This provided an incentive for general models in information retrieval to be explored, and for unified frameworks, capable of maximizing available information in the process of retrieval, to be proposed. We then introduced combined data and four retrieval tasks central to the creation of a unified model for entity-oriented search. Next, we presented the problem statement, describing the three main goals of this doctoral work, and formalizing them as a thesis statement to be proven. Finally, we described the re search line that led us from one step to the next in proving this thesis, presenting the thesis outline over each part and chapter, along

with the main reasons to read this thesis, as well as the main contributions that we produced during this doctoral work. 27Part I STATE OF THE ART2 GRAPH-BASEDENT

[REF7] - paperID: d121c33a5a0d8b6615d8581cfee8a941ebc7daed
Title: Graph-based entity-oriented search   Chunk of text: For example, by selecting a specific entity as the query (in put), and ranking other entities (output), we are able to run the task of related entity finding. 1837.4 hypergraph-of-entity summary In this chapter, we began by arguing the importance of cross-referencing over all available information, for solving general information needs. We have considered the hypergraph data structure as an instrument of generalization, and as an alter native solution for capturing higher-order dependencies in documents, entities and their relations. We then explored the idea of a unified framework for information retrieval, providing clear instances for joint representation approaches and for gen eral retrieval based on a hypergraph. We then proposed a unified model for the representation and retrieval of text and knowledge, supporting multiple tasks of entity-oriented search: ad hoc document retrieval, ad hoc entity retrieval, related entity finding, and entity list completion. The hypergraph-of-entity was proposed as a joint representation of terms, entities and their relations, for indexing corpora and knowledge bases in a unified manner. In this model, entities were linked to other related entities as a group, either according to the knowledge base (e.g., the subject and the respective target objects), or simply the occurrence in a common document; terms and entities were all linked by a document hyperedge, as a bag of words and entities; and terms were linked to entities that they represented or illustrated in some way (e.g., based on string matching with the entity’s name; per haps also good for cross-language retrieval).

[REF8] - paperID: 05109f905d1015a4fa6f8c361128813bf196535b
Title: CapsTM: capsule network for Chinese medical text matching   Chunk of text: Value Embedding size 300 Embedding size 768 #Hidden states in BiLSTM 100 #Hidden states in BiLSTM 384 #Capsules 6 #Capsules 6 #Dimension of capsules 50 #Dimension of capsules 50 #Iterations 3 #Iterations 3 Learning rate 0.001 Learning rate 0.001 Dropout rate 0.5 Dropout rate 0.9 Activation function ReLU Activation function ReLU Batch size 32 Batch size 32Yu et al. BMC Med Inform Decis Mak 2021, 21(Suppl 2):94 Page 6 of 8 and all models are trained on machines with NVIDIA GeForce GTX 1080ti GPU. Te performance of models is measured by precision (P), recall (R) and F-score. Results and discussion As shown in Table 4 where all the highest values in each type are highlighted in bold, when using Chi nese character embeddings initialized by word2vec, CapsTM(word2vec) achieves an F-score of 0.8432, and outperforms other state-of-the-art neural networks except BERT. Te diference ranges from 0.65 to 3.85% in F-score. When using Chinese character embeddings initialized by BERT, the F-score of CapsTM(BERT) increases to 0.8666, which is higher than that of BERT by 0.2%. Compared to ESIM, CapsTM(word2vec) is signif cantly better with an improvement of 1.49% in F-score, indicating that the capsule layer added is efective. In addition to investigate the efect of the attention mechanism used in the representation layer and the dynamic routing algorithm used in the capsule layer, we conduct ablation study on CapsTM.

[REF9] - paperID: d7d46a173fcb6808d1c78734b9d708078a20fc41

Title: A Modern Perspective on Query Likelihood with Deep Generative Retrieval Models Chunk of text: Section 4 describes our design of experiments, whose results are reported and discussedin Section 5. The accompanying source code is available at https: //github.com/CPJKU/DeepGenIR. 2 RELATED WORK 2.1 Neural Retrieval Models In the category of query-document interaction models, we can dis tinguish between three groups of models. The first group captures patterns of similarity values across terms that appear close together within the query and within the document [11, 20, 21, 34]. The second group captures patterns of frequencies across ranges of sim ilarity values [7, 12–14, 50]. The last ones are based on large-scale pre-trained language models, as the use of these models has shown significant performance gains in various IR tasks.

....
....
....

Title: Methods for Text Representation in Neural Information Retrieval

> Entering new chain...

Prompt after formatting:

System: You are a renowned scientist who is writing a survey a section of a survey entitled 'Methods for Text Representation in Neural Information Retrieval'.

Human: Your task is to write the contents of a section of a survey. The title of the section that you are writing is 'Methods for Text Representation in Neural Information Retrieval'.

To complete this task, I will give you a list of documents that you must use as references. Each document has a text and an alphanumeric ID.

When writing the section, you MUST follow this rules:

- be aware of plagiarism, i.e., you should not copy the text, but use them as inspiration.

- when using some reference, you must cite it right after its use. You should use the IEEE citing style (write the id of the text between brackets).

- you are writing the paragraphs of the section. You MUST write only this section.

- you MUST NOT split the section in subsections, nor create introduction and conclusion for it.

- DO NOT write any conclusion in any form for the subsection.

Do you understand your task?

AI: Sure, send me a list of text and I will write a section about Methods for Text Representation in Neural Information Retrieval using them as references. I am aware that I should use the IEEE citing style.

Human: ID: REF0

Text: Wa ∈ RM×SM, ba and bk are learnable parameters. The (*) T is the transpose operation and the output of the layer is O = [h0, ..., ht , ..., hN], a concatenation of the hidden states produced by the AttLSTM at each time step. As mentioned before, we propose a

feature ensemble by using an interpretable multi-source fusion component (IMF). The IMF aims to combine features from different sources. A naive way of doing this is concatenating the vector representations into a single vector. This scheme considers all sources equally, but one source may yield a better result than others.

ID: REF1
*Text: Introduction Information retrieval (IR) systems traditionally comprise of two stages: retrieval and ranking. Given a user query, the role of the retrieval stage is to quickly retrieve a set of candidate documents *Both authors contributed equally to the paper. †Work carried out while working at Amazon. 10 3 10 4 10 5 10 6 Document corpus size (MB) 0.340 0.345 0.350 0.355 0.360 0.365 0.370 0.375 0.380 Ranking quality (MRR@10) BERTSPLIT (uncompressed) SDR SDR (float16) Baseline Figure 1: MRR@10 performance vs. document corpus size tradeoff, measured on the MSMARCO-DEV dataset. BERTSPLIT is a distilled late-interaction model with reduced vector width and no compression (§ 4.2). For MRR@10 above 0.35, SDR is 4x-11.6x more efficient compared to the baseline.*

ID: REF2
Text: (a) Transformer : it has achieved very good results in the field of machine translation. (b) SVM : it was first used to solve the two-classification problem in pattern recognition, and it has achieved good classification results in the fields of text classification, handwriting recognition, and image processing. (c) TextRnn : it is a model that uses RNN for text classification. (d) FastText : its biggest feature is that the model is simple, the training speed is very fast, and it is widely used in the field of text classification.

ID: REF3
Text: 0.684 clean tf-idf 0.881 0.706 0.791 0.669 0.688 clean, glove 0.847 0.722 0.766 0.618 0.67 clean spacy 0.846 0.746 0.784 0.655 0.7 nps, clean, tf-idf 0.876 0.714 0.79 0.654 0.684 nps, clean, spacy 0.837 0.728 0.768 0.646 0.687 fct words 0.672 0.628 0.614 0.564 0.596 POS n-grams 0.754 0.573 0.723 0.607 0.59 pronouns 0.594 0.596 0.656 0.636 0.616 filter POS 0.832 0.731 0.765 0.657 0.694 Table 1: Cross-validation and test-set results of logistic regression classifier with different feature extraction processes. generalize to the new texts. The second run (run2, 0.666 macro F1) uses the glove pre-trained embeddings for English. This result represents the biggest surprise of the three since it did not use Italian embeddings. We observe that the English glove representations cover more than 60 Cleaned texts aid the classifier by a significant threshold.

ID: REF4
Text: For each social media platform, training and test data were provided. Furthermore, two cross-platform sub-tasks were introduced to test the systems' ability to generalize across platforms. The ultimate goal of HaSpeeDe 2 at EVALITA 2020 (Basile et al., 2020) is to take a step further in state-of-the-art HS detection for Italian. By doing this, we also intend to explore other side phenomena and see the extent to which they can be automatically distinguished from HS. We propose a single training set made of tweets, but two separate test sets within two different domains:

tweets and news headlines. While social media are still one of the main channels used to spread hateful content online (Alkiviadou, 2019; Wodak, 2018), an important role in this respect is also played by traditional media, and newspapers in particular. Furthermore, we chose to include another HS related phenomenon, namely the presence of stereotypes referring to one of the targets identified within our dataset (i.e., muslims, Roma and immigrants).

ID: REF5
Text: StdDev. .004 .005 .008 .003 .001 .001 determines the indexing throughput and query encoding latency, as well as the training batch size which influences the GPU memory requirements. The TREC-DL'20 query set was recently released, therefore most related work is missing results on these queries. We observe that the methods not using knowledge distillation and larger encoders (ANCE, LTRe) are outperformed on TREC-DL'19 by those that do use teachers (TCT, Margin-MSE), however on the sparse MSMARCO-DEV the result trend turns around. RocketQA

ID: REF6
Text: As for the noun war, its nearest neighbors in the distributional semantic model are conflict, warfare, invasion, 1991-95 Serbo-Croatian, Israel Hezbollah, genocide, Bosnia war, Jehad, civil war, Croatia war, Cold War, Iran-Iraq, wartime, Vietnam-like, etc. – that is, rather abstract concepts. The only more concrete words referring to physical combat action that occur in the distributional semantic neighborhood of war are battlefield and bloodshed, but this is not enough to outweigh the abstract terms. Thus, the distributional semantic model models warfare in terms of abstract rather than concrete (such as names of weapons, military equipment, military personnel, etc.) concepts. As a result, military action is not sufficiently juxtaposed to the metaphoric meaning of war as 'a situation in which two people or groups of people fight, argue, or are extremely unpleasant to each other'. In the case of answer and agreement, their nearest distributional semantic neighbors in the model are fairly abstract concepts: explanation, answer, reply, solution, unanswerable, query, TV-talkback answer, question, and yes (for answer), and accord, pact, deal, treaty, initial, negotiation, memorandum, compromise, and negotiate (for agreement). Meanwhile, human subjects rank answer and agreement rather high in concreteness; presumably, this is a consequence of conflating the mental representations of the action of answering / reaching an agreement with their two modes – the spoken and the written, i.e. with the physical actions of speaking and writing. This conflation is not reflected in discourse – it largely exists in the mental representations of answer and agreement and, therefore, is not very distinguishable on the level of linguistic representation.

ID: REF7
Text: Sentence embeddings produced by the UmBERTo NLM are concatenated to metadata and image embeddings produced by three popular pre-trained vision modals. The three resulting multimodal embeddings are fed separately to feedforward networks, and the final outcome is selected through majority voting. task of DANKMEMES, aimed at discriminating memes from standard images

```
containing actors from the Italian political scene. Task organizers
extracted a total of 1600 training images from the Instagram
platform, and data available from each dataset entry – text, actors
and user engagement, among others – were leveraged to train an
ensem ble of multimodal models performing meme de tection through
majority-vote. The following sec tions present our approach in
detail, first showing our preliminary evaluation of multiple
modeling approaches and then focusing on the final system’s main
modules and the features we leverage from the dataset. Finally,
results are presented, and we conclude by discussing the problems
we faced with some inconsistencies in the data. Our code is made
available at https://github.com/ jinensetpal/ArchiMeDe 2 System
Description ArchiMeDe is composed of a multimodal learn ing
ensemble, with the final output being the re sult of a majority
vote.

ID: REF8
Text: — RQ7: How does the number of important tokens m affect the
PRADA performance? 6.1 Baseline Comparison To answer RQ1, we
compare PRADA with different baselines under both the automatic
evaluations and human evaluations. Automatic evaluation. The
performance comparisons between our model and the baselines are
shown in Table 2. For the MS-MARCO-Doc, we have the following
observations: (1) Step-wise ACM Transactions on Information
Systems, Vol. 41, No. 4, Article 89. Publication date: April
2023.PRADA: Practical Black-box Adversarial Attacks against Neural
Ranking Models 89:17 Table 3.

ID: REF9
Text: The paradigm words are automatically selected  from the MRC
Psycholinguistic Database Ma chine Usable Dictionary (Coltheart,
1981), a col lection of 4,295 English words rated with degrees  of
abstractness by human subjects in psycholin guistic experiments.
Tsvetkov et al. (2013) also compute the con creteness indexes of
English words by using a dis tributional semantic model and the MRC
data base. They train a logistic regression classifier on  1,225
most abstract and 1,225 most concrete  words from MRC; the degree
of concreteness of a  word is the posterior probability produced by
the  classifier. The Tsvetkov et al. system for meta phor
identification with concreteness indexes is  based on cross-lingual
model transfer, when the  model is trained on English data, and
then the  classification features are translated into other
lan guages by means of an electronic dictionary. 335
yuliya.badryzlova@gmail.com  Badryzlova (2020) explores
concreteness and  abstractness indexes for linguistic metaphor
iden tification in Russian and English. The paradigm  words are
selected in a semi-automatic fashion:  the Russian paradigm is
derived from the Open  Semantics of the Russian Language, the
semanti cally annotated dataset of the KartaSlov database (Kulagin,
2019); the English paradigm is selected  from the MRC database
(Coltheart, 1981). The in dexes of concreteness and abstractness
are com puted for large sets of Russian and English words  (about
18,000 and 17,000 lexemes, respectively).


> Finished chain.
Methods for Text Representation in Neural Information Retrieval
```

In the field of Neural Information Retrieval (NIR), text representation plays a crucial role in the retrieval and ranking stages of IR systems [REF1]. The goal of text representation is to transform textual data into a numerical format that can be effectively processed by neural models. In this section, we discuss various methods for text representation in NIR, drawing inspiration from the following references: [REF0], [REF2], [REF3], [REF6], and [REF9].

One commonly used method for text representation in NIR is the use of vector representations. These representations aim to capture the semantic meaning of words and phrases in a continuous vector space. One approach is to use pre-trained word embeddings, such as GloVe or Word2Vec, which map words to dense vectors based on their co-occurrence statistics [REF3]. These embeddings can be directly used to represent individual words in a text. Another approach is to use contextualized word embeddings, such as ELMo or BERT, which capture the contextual meaning of words in a sentence [REF1]. These embeddings provide more nuanced representations by considering the surrounding context.

In addition to word-level representations, neural models can also benefit from higher-level representations that capture the semantics of longer text segments, such as sentences or documents. One popular method for sentence representation is the use of recurrent neural networks (RNNs), such as TextRnn [REF2]. RNNs process sequential data by maintaining hidden states that capture the context of previous words. These hidden states can be used as representations of sentences or documents. Another approach is the use of transformer models, which have achieved state-of-the-art results in various natural language processing tasks [REF2]. Transformers leverage self-attention mechanisms to capture global dependencies in the input text, enabling them to generate informative representations.

In the context of NIR, it is often necessary to combine information from multiple sources or modalities. One way to achieve this is through feature fusion, where vector representations from different sources are concatenated into a single vector [REF0]. However, this naive approach treats all sources equally, without considering the varying importance or relevance of each source. To address this, interpretable multi-source fusion components (IMFs) have been proposed [REF0]. IMFs aim to combine features from different sources in a way that assigns appropriate weights to each source, allowing for more effective fusion.

In conclusion, text representation is a crucial aspect of Neural Information Retrieval. Various methods, such as word embeddings, contextualized embeddings, recurrent neural networks, and transformer models, have been employed to capture the semantic meaning of text. Additionally, techniques like feature fusion and interpretable multi-source fusion components have been proposed to combine information from multiple sources. These methods contribute to the effectiveness of neural models in retrieving and ranking relevant information in NIR tasks.

References:

[REF0] - Wa ∈ RM×SM, ba and bk are learnable parameters. The (*) T is the transpose operation and the output of the layer is O = [h0, ..., ht , ..., hN], a concatenation of the hidden states produced by the AttLSTM at each time step. As mentioned before, we propose a feature ensemble by using an interpretable multi-source fusion component (IMF). The IMF aims to combine features from different sources. A naive way of doing this is concatenating the vector representations into a single vector. This scheme considers all sources equally, but one source may yield a better result than others.
[REF1] - Introduction Information retrieval (IR) systems traditionally comprise of two stages: retrieval and ranking. Given a user query, the role of the retrieval stage is to quickly retrieve a set of candidate documents. BERTSPLIT is a distilled late-interaction model with reduced vector width and no compression (§ 4.2). For MRR@10 above 0.35, SDR is 4x–11.6x more efficient compared to the baseline.
[REF2] - Transformer: it has achieved very good results in the field of machine translation. SVM: it was first used to solve the two-classification problem in pattern recognition, and it has achieved good classification results in the fields of text classification, handwriting recognition, and image processing. TextRnn: it is a model that uses RNN for text classification. FastText: its biggest feature is that the model is simple, the training speed is very fast, and it is widely used in the field of text classification.
[REF3] - Cross-validation and test-set results of logistic regression classifier with different feature extraction processes. Generalize to the new texts. The second run (run2, 0.666 macro F1) uses the glove pre-trained embeddings for English. We observe that the English glove representations cover more than 60 Cleaned texts aid the classifier by a significant threshold.
[REF6] - As for the noun war, its nearest neighbors in the distributional semantic model are conflict, warfare, invasion, 1991-95 Serbo-Croatian, Israel-Hezbollah, genocide, Bosnia war, Jehad, civil war, Croatia war, Cold War, Iran-Iraq, wartime, Vietnam-like, etc. – that is, rather abstract concepts. The only more concrete words referring to physical combat action that occur in the distributional semantic neighborhood of war are battlefield and bloodshed, but this is not enough to outweigh the abstract terms.
[REF9] - The paradigm words are automatically selected from the MRC Psycholinguistic Database Machine Usable Dictionary (Coltheart, 1981), a collection of 4,295 English words rated with degrees of abstractness by human subjects in psycholinguistic experiments. Tsvetkov et al. (2013) also compute the concreteness indexes of English words by using a distributional semantic model and the MRC database. They train a logistic regression classifier on 1,225 most abstract and 1,225 most concrete words from MRC; the degree of concreteness of a word is the posterior probability produced by the classifier.

References given to GPT:
[REF0] - paperID: bd23ce64a6422c1f73acf51675e53b7a06547da3
Title: UOBIT @ TAG-it: Exploring a Multi-faceted Representation for Profiling Age, Topic and Gender in Italian Texts
Chunk of text: Wa ∈ RM×SM, ba and bk are learnable

parameters. The (*) T is the transpose operation and the output of the layer is O = [h0, ..., ht , ..., hN], a concatenation of the hidden states produced by the AttLSTM at each time step. As mentioned before, we propose a feature ensemble by using an interpretable multi-source fusion component (IMF). The IMF aims to combine features from different sources. A naive way of doing this is concatenating the vector representations into a single vector. This scheme considers all sources equally, but one source may yield a better result than others.

[REF1] - paperID: 8a6125562341d9a839006a23b48c870504810a27
Title: SDR: Efficient Neural Re-ranking using Succinct Document Representation Chunk of text: Introduction Information retrieval (IR) systems traditionally comprise of two stages: retrieval and ranking. Given a user query, the role of the retrieval stage is to quickly retrieve a set of candidate documents *Both authors contributed equally to the paper. †Work carried out while working at Amazon. 10 3 10 4 10 5 10 6 Document corpus size (MB) 0.340 0.345 0.350 0.355 0.360 0.365 0.370 0.375 0.380 Ranking quality (MRR@10) BERTSPLIT (uncompressed) SDR SDR (float16) Baseline Figure 1: MRR@10 performance vs. document corpus size tradeoff, measured on the MSMARCO-DEV dataset. BERTSPLIT is a distilled late-interaction model with reduced vector width and no compression (§ 4.2). For MRR@10 above 0.35, SDR is 4x–11.6x more efficient compared to the baseline.

[REF2] - paperID: c537c75fa35d4060474862b82f31523811ae84da
Title: LawRec: Automatic Recommendation of Legal Provisions Based on Legal Text AnalysisChunk of text: (a) Transformer : it has achieved very good results in the field of machine translation. (b) SVM : it was first used to solve the two-classification problem in pattern recognition, and it has achieved good classification results in the fields of text classification, handwriting recognition, and image processing. (c) TextRnn : it is a model that uses RNN for text classification. (d) FastText : its biggest feature is that the model is simple, the training speed is very fast, and it is widely used in the field of text classification.

[REF3] - paperID: bd23ce64a6422c1f73acf51675e53b7a06547da3
Title: UOBIT @ TAG-it: Exploring a Multi-faceted Representation for Profiling Age, Topic and Gender in Italian Texts
Chunk of text: 0.684 clean tf-idf 0.881 0.706 0.791 0.669 0.688 clean, glove 0.847 0.722 0.766 0.618 0.67 clean spacy 0.846 0.746 0.784 0.655 0.7 nps, clean, tf-idf 0.876 0.714 0.79 0.654 0.684 nps, clean, spacy 0.837 0.728 0.768 0.646 0.687 fct words 0.672 0.628 0.614 0.564 0.596 POS n-grams 0.754 0.573 0.723 0.607 0.59 pronouns 0.594 0.596 0.656 0.636 0.616 filter POS 0.832 0.731 0.765 0.657 0.694 Table 1: Cross-validation and test-set results of logistic regression classifier with different feature extraction processes. generalize to the new texts. The second run (run2, 0.666 macro F1) uses the glove pre-trained embeddings for English. This result represents the biggest surprise of the three since it did not use Italian embeddings. We observe that the English glove representations cover more than 60 Cleaned texts aid the classifier by a significant threshold.

[REF4] - paperID: bd23ce64a6422c1f73acf51675e53b7a06547da3
Title: UOBIT @ TAG-it: Exploring a Multi-faceted

Representation for Profiling Age, Topic and Gender in Italian Texts
Chunk of text: For each social me dia platform, training and test data were provided. Furthermore, two cross-platform sub-tasks were introduced to test the systems’ ability to generalize across platforms. The ultimate goal of HaSpeeDe 2 at EVALITA 2020 (Basile et al., 2020) is to take a step further in state-of-the-art HS detection for Italian. By do ing this, we also intend to explore other side phe nomena and see the extent to which they can be automatically distinguished from HS. We propose a single training set made of tweets, but two separate test sets within two different do mains: tweets and news headlines. While social media are still one of the main channels used to spread hateful content online (Alkiviadou, 2019; Wodak, 2018), an important role in this respect is also played by traditional media, and newspapers in particular. Furthermore, we chose to include another HS related phenomenon, namely the presence of stereotypes referring to one of the targets identi fied within our dataset (i.e., muslims, Roma and immigrants).

[REF5] - paperID: 4deed74a3eee7e629dce2b8ef1e437ca74b2e64a
Title: Efficiently Teaching an Effective Dense Retriever with Balanced Topic Aware Sampling   Chunk of text: StdDev. .004 .005 .008 .003 .001 .001 determines the indexing throughput and query encoding latency, as well as the training batch size which influences the GPU memory requirements. The TREC-DL’20 query set was recently released, therefore most related work is missing results on these queries. We observe that the methods not using knowledge distillation and larger encoders (ANCE, LTRe) are outperformed on TREC-DL’19 by those that do use teachers (TCT, Margin-MSE), however on the sparse MSMARCO-DEV the result trend turns around. RocketQA

[REF6] - paperID: bd23ce64a6422c1f73acf51675e53b7a06547da3
Title: UOBIT @ TAG-it: Exploring a Multi-faceted Representation for Profiling Age, Topic and Gender in Italian Texts
Chunk of text: As for the noun war, its nearest neighbors in the distributional semantic model are conflict, war fare, invasion, 1991-95 Serbo-Croatian, Israel Hezbollah, genocide, Bosnia war, Jehad, civil war, Croatia war, Cold War, Iran-Iraq, wartime, Vietnam-like, etc. – that is, rather abstract con cepts. The only more concrete words referring to physical combat action that occur in the distribu tional semantic neighborhood of war are battle field and bloodshed, but this is not enough to out weigh the abstract terms. Thus, the distributional semantic model models warfare in terms of ab stract rather than concrete (such as names of weapons, military equipment, military personnel, etc.) concepts. As a result, military action is not sufficiently juxtaposed to the metaphoric meaning of war as ‘a situation in which two people or groups of people fight, argue, or are extremely un pleasant to each other’. In the case of answer and agreement, their near est distributional semantic neighbors in the model are fairly abstract concepts: explanation, answer, reply, solution, unanswerable, query, TV-talkback answer, question, and yes (for answer), and ac cord, pact, deal, treaty, initial, negotiation, mem orandum, compromise, and negotiate (for agree ment). Meanwhile, human subjects rank answer and agreement rather high in concreteness; pre sumably, this is a consequence of conflating the mental representations of the action of answer ing

/ reaching an agreement with their two modes – the spoken and the written, i.e. with the physical actions of speaking and writing. This conflation is not reflected in discourse – it largely exists in the mental representations of answer and agreement and, therefore, is not very distinguishable on the level of linguistic representation.

[REF7] - paperID: bd23ce64a6422c1f73acf51675e53b7a06547da3
Title: UOBIT @ TAG-it: Exploring a Multi-faceted Representation for Profiling Age, Topic and Gender in Italian Texts
Chunk of text: Sentence embeddings produced by the UmBERTo NLM are concatenated to metadata and image embeddings produced by three popular pre-trained vi sion modals. The three resulting multimodal embeddings are fed separately to feedforward networks, and the final outcome is selected through majority voting. task of DANKMEMES, aimed at discriminating memes from standard images containing actors from the Italian political scene. Task organizers extracted a total of 1600 training images from the Instagram platform, and data available from each dataset entry – text, actors and user engagement, among others – were leveraged to train an ensem ble of multimodal models performing meme de tection through majority-vote. The following sec tions present our approach in detail, first showing our preliminary evaluation of multiple modeling approaches and then focusing on the final system's main modules and the features we leverage from the dataset. Finally, results are presented, and we conclude by discussing the problems we faced with some inconsistencies in the data. Our code is made available at https://github.com/ jinensetpal/ArchiMeDe 2 System Description ArchiMeDe is composed of a multimodal learn ing ensemble, with the final output being the re sult of a majority vote.

[REF8] - paperID: 21ee4b66ce53de6b7b23c23cae0885bf5c96ad78
Title: PRADA: Practical Black-Box Adversarial Attacks against Neural Ranking Models Chunk of text: — RQ7: How does the number of important tokens m affect the PRADA performance? 6.1 Baseline Comparison To answer RQ1, we compare PRADA with different baselines under both the automatic evaluations and human evaluations. Automatic evaluation. The performance comparisons between our model and the baselines are shown in Table 2. For the MS-MARCO-Doc, we have the following observations: (1) Step-wise ACM Transactions on Information Systems, Vol. 41, No. 4, Article 89. Publication date: April 2023.PRADA: Practical Black-box Adversarial Attacks against Neural Ranking Models 89:17 Table 3.

[REF9] - paperID: bd23ce64a6422c1f73acf51675e53b7a06547da3
Title: UOBIT @ TAG-it: Exploring a Multi-faceted Representation for Profiling Age, Topic and Gender in Italian Texts
Chunk of text: The paradigm words are automatically selected from the MRC Psycholinguistic Database Ma chine Usable Dictionary (Coltheart, 1981), a col lection of 4,295 English words rated with degrees of abstractness by human subjects in psycholin guistic experiments. Tsvetkov et al. (2013) also compute the con creteness indexes of English words by using a dis tributional semantic model and the MRC data base. They train a logistic regression classifier on 1,225 most abstract and 1,225 most concrete words from MRC; the degree of concreteness of a word is the posterior probability produced by the classifier. The Tsvetkov

et al. system for metaphor identification with concreteness indexes is based on cross-lingual model transfer, when the model is trained on English data, and then the classification features are translated into other languages by means of an electronic dictionary. 335 yuliya.badryzlova@gmail.com Badryzlova (2020) explores concreteness and abstractness indexes for linguistic metaphor identification in Russian and English. The paradigm words are selected in a semi-automatic fashion: the Russian paradigm is derived from the Open Semantics of the Russian Language, the semantically annotated dataset of the KartaSlov database (Kulagin, 2019); the English paradigm is selected from the MRC database (Coltheart, 1981). The indexes of concreteness and abstractness are computed for large sets of Russian and English words (about 18,000 and 17,000 lexemes, respectively).

....
....
....