Revisiting Anwesha: Enhancing Personalised and Natural Search in Bangla

Arup Das
IIT Madras, India,
cs20s016@smail.iitm.ac.in

Joyojyoti Acharya
IIT Madras, India,
cs21m024@smail.iitm.ac.in

Bibekananda Kundu
CDAC Kolkata, India,
bibekananda.kundu@gmail.com

Sutanu Chakraborti
IIT Madras, India,
sutanuc@cse.iitm.ac.in

Abstract

Bangla is a low-resource, highly agglutinative language. Thus it is challenging to facilitate an effective search over Bangla documents. We have created a gold standard dataset containing query document relevance pairs for evaluation purposes. We utilise Named Entities to improve the retrieval effectiveness of traditional Bangla search algorithms. We suggest a reasonable starting model for leveraging implicit preference feedback based on the user search behaviour to enhance the results retrieved by the Explicit Semantic Analysis (ESA) approach. We use contextual sentence embeddings obtained via Language-agnostic BERT Sentence Embedding (LaBSE) to rerank the candidate documents retrieved by the traditional search algorithms (tf-idf) based on the top sentences that are most relevant to the query. This paper presents our empirical findings across these directions and critically analyses the results.

1 Introduction

Owing to India's multilingual diversity, it is important to ensure that a wide gamut of people from diverse backgrounds are able to access the web without any language barrier. Bengali alternatively known as Bangla, has 300 million speakers globally and has witnessed the fastest growth among the other Indic languages in terms of internet usage (KPMG, 2017). Hence there is a pressing need to develop tools that facilitate semantic search over Bangla text documents. Previously, efforts have been put into creating Bangla Search engines. For example, Anwesan[1] was built to search over Rabindra Rachanabali collection[2] (Das et al., 2012). Sandhan[3] is a monolingual domain-specific search engine limited to tourism and health domains in nine Indian languages: Bangla, Hindi, Marathi, Tamil, Telugu, Punjabi, Odiya, Gujarati and Assamese. It is based on the Bag of Words model and focuses more on improving recall than precision (Priyatam et al., 2012). Hence the top results are not always relevant for the query. Pipilika[4], launched on April 13, 2013, is designed for the residents of Bangladesh. It crawls data from Bangla News, Bangla Blogs and Bangla Wikipedia. However, Anwesan and Pipilika[5] are not presently accessible for exploration. This paper reports follow-up work based on (Das et al., 2022) recent work, which introduced an exploration toward building অন্বেষা (Anwesha), a prototype for a search engine in Bangla. Anwesha demonstrated promise in addressing the existing search engines' shortcomings and advanced the research done in the information retrieval (IR) space for the Bangla language. Anwesha incorporated the use of diverse knowledge sources like IndoWordNet[6] (Bhattacharyya, 2010), statistical co-occurrences (by way of Latent Semantic Analysis (LSA) (Deerwester et al., 1990)) and external knowledge sources like Wikipedia (by way of Explicit Semantic Analysis (ESA) (Gabrilovich and Markovitch, 2007)) for facilitating effective retrieval, opening gateways to further improvements in the search quality results. The authors have released a Gold Standard dataset[7] containing 94 query doc-

[1] http://anwesan.iitkgp.ernet.in/
[2] https://rabindra-rachanabali.nltr.org/
[3] http://sandhan.tdil-dc.gov.in/Search
[4] https://pipilika.com/
[5] https://en.wikipedia.org/wiki/Pipilika
[6] The official website and the web interface of IndoWordNet: https://www.cfilt.iitb.ac.in/indowordnet/
[7] https://doi.org/10.5281/zenodo.6583149

Proceedings of the 19th International Conference on Natural Language Processing (ICON), pages 183 - 193
December 15-18, 2022 ©2022 Association for Computational Linguistics

ument relevance pairs over a test collection of 1182 documents. The collection contains 182 short stories, novels and essays written by Rabindranath Tagore[8] and 1000 newspaper articles published in 2013 crawled from the daily newspaper of Bangladesh Prothom Alo[9]. Every query was designed to belong to one of the four different complexity levels, as shown in Table 1. By designing queries of different complexity levels, the effectiveness of Anwesha as the queries become more difficult to resolve could be studied. An approach like tf-idf works best on precise queries that directly match the content of the relevant documents (complexity level 1). For queries which were not precise, query expansion techniques using IndoWordnet helped make a lexical search like tf-idf perform effectively (complexity level 2). LSA performs well when the query and the retrieved relevant documents do not share many words in common; instead, they share a common theme (complexity levels 3 and 4). ESA performs well when the queries require external background knowledge for their intent resolution (complexity level 4). The queries created with different complexity levels demonstrated that there is no silver bullet which works the best across all types of queries. Each of the top ten documents retrieved by their search algorithm was assigned a score of 1 if irrelevant, 2 if partially relevant, and 3 if completely relevant by at least five Bangla users. Further, Anwesha explains the search results by highlighting words from the documents LSA or ESA reckon to be semantically related to the query.

In its present form, Anwesha has been evaluated on a small set of queries. The lack of handling of multi-word expressions has adversely affected Anwesha's performance in several cases, especially in complexity level 1 query. It does not use the implicit feedback the user provides via click preferences to improve the retrieval effectiveness. For best results, users are restricted by the choice of words that exactly match the contents of relevant documents. Any change in word order in the query can potentially lead to contrasting results, which are not captured in Anwesha.

[8] https://rabindra-rachanabali.nltr.org/

[9] https://www.prothomalo.com/

This paper presents four directions to address the current limitations of Anwesha. First, we expand the Gold Standard dataset by creating an additional 100 query document relevance pairs over a new test collection[10] of 1000 documents for a more exhaustive evaluation and better analysis. Second, we identify the technical terms and named entities in the documents and queries apart from considering only uni-gram word tokens as was done previously. Third, we introduce a novel approach to improve the effectiveness of the IR system by incorporating implicit preference feedback via clickthrough data in an ESA setting. Lastly, we present the usage of contextual vector representations of documents and query using Language-agnostic BERT Sentence Embedding (LaBSE) (Feng et al., 2020) to rerank the documents based on the best sentences from the document that are relevant to the query. We believe that our approaches can be adapted for other low-resource, highly inflected and agglutinative languages similar to Bangla, such as Assamese, Maithili, Oriya and Manipuri (Ray et al., 1966).

The rest of this paper is organized as follows. In Section 2, we position our approaches in the context of background work and relevant research. Section 3 describes the implementation details of our approaches. Section 4 presents a critical analysis of our empirical findings and observations. Section 5 summarizes our key contributions and discusses potential extensions of the work.

2 Background and Literature Survey

This section discusses the concepts that will be used in the rest of the paper.

2.1 Multiword Expressions (MWEs)

MWEs are frequently repeating idiosyncratic phrasal units exhibiting varying degree of semantic compositionality (Chakraborty et al., 2014) (Dandapat et al., 2006). Identifying MWEs is known to play an important role in understanding natural language queries which in turn helps in improving the retrieval effectiveness (Acosta et al., 2011). In the present work, we focus on extracting only the Named Entities (NEs) like names of people (রবীন্দ্রনাথ ঠাকুর (EN: Rabindranath

[10] https://zenodo.org/record/7376906

Query Type	Complexity Level
The query contains exact words, phrases or sentence from the document.	1
The query is not present as it is in the document. There is a slight deviation.	2
The query is a generalised phrase capturing the overall story or the document's theme.	3
It is a general query not related to any specific document.	4

Table 1: Definition of the complexity level of a query

Thakur)), names of locations (পোর্ট ব্লেয়ার (EN: Port Blair)), names of organisations (নাসিরাবাদ পলিটেকনিক ইনস্টিটিউট (EN: Nasirabad Polytechnic Institute)) etc.

In order to detect a multiword NE token in a document or query we used IndicNER (Arnav Mhaske, 2022). IndicNER[11] was trained on the largest publicly available NE Annotated dataset for Indic languages (961679 training instances in the case of Bangla) while the one devised by Sagor Sarker (Sarker, 2021) was trained on a smaller dataset (64155 sentences).

2.2 Explicit Semantic Analysis (ESA)

ESA exploits knowledge of Wikipedia. Terms and documents are expressed in terms of underlying interpretable concepts, where each concept corresponds to a Wikipedia article name. There is an overlap between the concepts shared by similar query terms. For example, the query terms "ক্যান্সার" (EN: cancer) and "কার্সিনোমা" (EN: carcinoma) share লিউকোমিয়া (EN: Leukemia), বায়োপসি (EN: biopsy), সার্ভিকাল (EN: cervical) and ম্যালিগন্যান্ট (EN: malignant) as top concepts. This helps in retrieving relevant documents even when they do not contain the query terms.

2.3 Implicit Preference Feedback

When the information needed is tacit, and the user cannot express her intent, we often do not expect the user to reformulate the query from scratch on search failure. Further, diverse intents can give rise to the same query. Such challenges make it difficult to arrive at an appropriate query representation in the concept space of ESA. However, it becomes easy to implicitly refine the query by unobtrusively studying the user's preference of documents to a query by assessing their interaction with the IR system. Therefore, it is reasonable to engage in implicit iterative query refinement by analyzing the documents selected by the user. The implicit preference feedback system assumes that clicking on a document and viewing it indicates the user's interest in the document's contents (White et al., 2002). Viewing some documents may lead users to refine their understanding of the information they seek and help disambiguate their search requirements (Manning et al., 2008). Explicit feedback can be substituted with implicit feedback in web-based IR, and such feedback represents the user preferences reasonably accurately (Joachims et al., 2005).

2.4 Usage of Language Models for low-resource IR

Queries written in natural language enable better search results when the system can take into account the order of the words in the sentences. Bidirectional Encoder Representations from Transformers (BERT) (Devlin et al., 2018) is a state-of-the-art approach to produce contextual embeddings that have outperformed previously existing methods like Word2vec (Mikolov et al., 2013), GloVe (Pennington et al., 2014), ELMo (Peters et al., 2018) and ULMFiT (Howard and Ruder, 2018) in tasks like question answering, sentence pairs similarity, sentence pair completion, named entity recognition, entailment classification, sentiment classification and several others because of its unsupervised and deeply bidirectional approach. Search engines like Google (Nayak, 2019) and Bing (Zhu, 2019) have been using BERT to understand the context of the query intent better and allow users to ask questions in a way humans ask experts. However, for low-resource languages like Bangla, where limited data is available, traditional vector space approaches like tf-idf or BM25[12] are preferred as these algorithms are computationally less intensive and do not require additional training

[11] https://huggingface.co/ai4bharat/IndicNER

[12] https://kmwllc.com/index.php/2020/03/20/understanding-tf-idf-and-bm-25/

data (Lin, 2019). Therefore, to benefit from the best of both worlds, we utilise BERT, as illustrated in Section 2.5, as a re-ranker over the top documents retrieved by the tf-idf vector space approach.

2.5 Ad Hoc document retrieval with BIRCH

As per a study (Qiao et al., 2019), MS MARCO[13] passage ranking is closer to the seq2seq task because of its question-answering focus and so pre-trained contextual models like BERT can perform well on them. But for TREC-style ad-hoc document retrieval tasks, we need to fine-tune on user clicks, and the surrounding context is not enough. Further, BERT was not trained with an objective to perform inference on long documents. A simple solution presented by the authors of BIRCH (Akkalyoncu Yilmaz et al., 2019) is to perform sentence-level inference on a candidate document and pick the best sentences (in practice three most relevant sentences) or paragraph in a document which will act as an appropriate proxy for document relevance. This approach, in one way, is a form of passage retrieval, where BERT has already been studied to perform well. The final score of a document to a query is as follows:

$$S_f = a \cdot S_{doc} + (1 - a) \cdot \Sigma_{i=1}^{n} w_i \cdot S_i \quad (1)$$

S_f is the final document score obtained using the BIRCH approach, S_{doc} is the original document score as per a traditional retrieval algorithm like BM25 or tf-idf, S_i is the i^{th} best sentence identified by BERT, a and w_i's are hyperparameters, tuned as per the parameters that gave the highest average precision (AP) score on the training folds.

2.6 Sentence Embeddings using LaBSE

A commonly used approach to obtain a sentence embedding from a BERT Base model is to average the BERT output layer (768 dimensions) or use the output of the [CLS] token from the last transformer block. However, these standard approaches often produce sentence embeddings, that are even worse than those obtained by averaging GloVe embeddings (Reimers and Gurevych, 2019). To obtain good sentence embeddings, we need to fine-tune the BERT output. IndicBERT produces multilingual word embeddings for 12 Indian languages (including Bangla) (Kakwani et al., 2020). LaBSE is a BERT multilingual embedding model developed by Google that generates cross-lingual sentence embeddings for 109 languages (including Bangla). It is trained on 17 billion monolingual sentences and 6 billion bilingual sentence translation pairs using Masked Language Modelling (MLM) and Translation Language Modelling (TLM) pre-training. The empirical benefits in terms of its effectiveness in diverse tasks including retrieval are analysed in (Feng et al., 2020). The sentence embeddings are obtained using l_2 normalized [CLS] token representations from the last transformer block.

3 Proposed Methodology

3.1 Identifying NEs in MWEs

We quantitatively analyzed the NEs detected by (Arnav Mhaske, 2022) and (Sarker, 2021) on two NE recognition datasets ((Karim et al., 2019) and (Pan et al., 2017)). We observed that the NEs detected by the latter were a subset of the NEs detected by IndicNER. Hence, to obtain the multiword NE tokens in a document or query, we used IndicNER.

3.2 Implicit Preference Feedback Strategy

The algorithm for implicit preference feedback with ESA is as follows:

- *Step 1*: The user issues a query.
- *Step 2*: The system returns the top retrieved documents based on the cosine similarity between the vector representation of the document and the query in the concept space.
- *Step 3*: The user clicks and views some of the retrieved documents.
- *Step 4*: The system promotes and demotes the top concepts of the query present in the documents visited by the user and documents viewed by the user but not visited, respectively. The weights for the top concepts of the query, which were common to both the highly preferred and less preferred documents, remain unaffected.
- *Step 5*: The system displays a revised set of retrieved results.
- *Step 6*: Repeat steps 3,4,5 until the user views no new interesting documents.

[13] https://microsoft.github.io/msmarco/

Implicit feedback inference can be made only on the documents the user has observed and assessed. A simple strategy studied in (Radlinski and Joachims, 2005) is adopted to determine the documents observed by a user. As per the study, a user generally follows the results from top to bottom and mostly observes the results from the document at rank one to the document below the one clicked and viewed by the user. A user at least looks at the top two results with equal attention. However, the user is more likely to click on the first result. So if a user only visits the first document presented in the retrieved result, it was assumed that the user had assessed only the first and second documents in the top retrieved documents. Hence the weight updates in the query's concept vector representation were made for the top two retrieved documents. For two documents, both visited, the document visited later should be given a higher preference (Joachims et al., 2005).

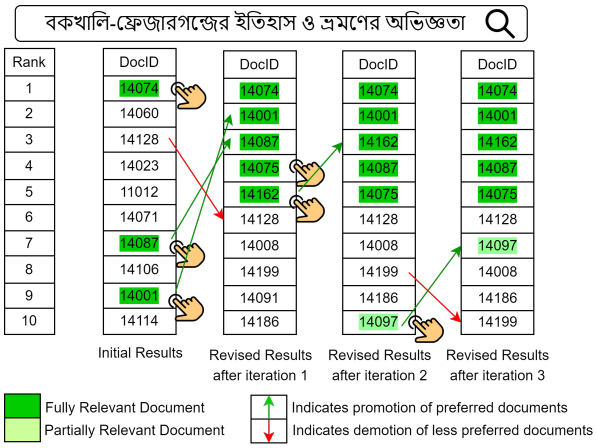

Figure 1: Top 10 retrieved results with implicit preference feedback after every iteration.

In Figure 1, we present the top retrieved results after every iteration of query refinement using the implicit preference feedback from the user for the query "বকখালি-ফ্রেজারগন্েজর ইতিহাস ও ভ্রমণের অভিজ্ঞতা"/bakakhāli-phrejāraganjera itihāsa ō bhramaṇera abhijñatā (EN: History and Travel Experience of Bakkhali-Fraserganj). We can observe that with every iteration, the vector representation of the query gets closer to the relevant documents in the concept space of ESA; as a result, the number of relevant documents in the top ranks increases. Some of the less preferred documents occupy lower ranks in the top results, while many of them disappear from the retrieved list.

3.3 Application of BERT (LaBSE) for document retrieval using BIRCH

IndicBert was trained on MLM task with a word-level objective using a cross-entropy loss function. In contrast, LaBSE was trained on MLM and TLM tasks with a sentence-level objective using additive margin softmax as a contrastive loss function. Since a dual encoder model is trained using a translation ranking loss, the similarity or dissimilarity of sentences in a shared embedding space is more adequately captured by LaBSE than IndicBERT, which uses a single BERT model. Thus, LaBSE can better discriminate amongst the most similar sentences. This was also confirmed through our empirical findings. Therefore we have used LaBSE to obtain the sentence embeddings in the BIRCH approach. We find an initial pool of top 150 candidate documents using the tf-idf approach. We used LaBSE to find the best three sentences that resolve the query in every candidate document. We used a convex combination (as in Equation (1)) of the LaBSE inference scores with retrieved scores from tf-idf to obtain the final document scores. In the original work, the optimal values for a and w_i's in Equation (1) were obtained by fine-tuning using an exhaustive grid search approach. Since we do not have enough data to fine-tune the hyperparameters, we give equal importance to the keyword-based approach (query-document cosine similarity based on tf-idf approach) and to the aggregated sentence level evidence obtained through LaBSE. In the absence of a large amount of training data, we use ESA scores as surrogates to guide the selection of w_i values since ESA scores are expected to correspond to a human assessment of similarities based on familiar background concepts. Interestingly, as highlighted in Section 4.4, this has been empirically found to consistently improve retrieval effectiveness compared to a scheme where the top three sentences are weighed equally. Hence $w_i =$ cosine similarity score of the query and sentence S_i for a candidate document d_j in the concept space defined by ESA. All embeddings for sentences in documents are avail-

able as part of pre-computation. Only query embeddings are created at runtime. Such a standard approach boosts time efficiency of retrieval[14].

4 Results and Analyses

4.1 Gold Standard Dataset Preparation

The Gold Standard dataset created by (Das et al., 2022) contained documents from the prominent dialect variations in Bangla: Sadhu Bhasa[15] and Chalit Bhasa[16]. In view of contributing to the linguistic diversity of the existing test collection, we curated a fresh dataset of 1000 text documents which consists of 642 documents for West Bengal Bangla news readers of Ebela, Zee News and Anandabazar Patrika (Kunchukuttan et al., 2020). One hundred twenty-eight news articles belong to the Entertainment, International and National category, while 129 news articles belong to the Kolkata and sports category. One hundred forty-five articles on the health-specific domain were obtained from Vikaspedia[17], an online information guide launched by the Government of India. The remaining 213 articles were based on the travel domain and were crawled from various Bangla travel blogs. We designed 100 queries each belonging to one of the four complexity levels in Table 1. We obtained graded relevance feedback from at least five Bangla annotators on the top ten documents retrieved by our search algorithms for a given query. A document was rated 1 if irrelevant, 2 if partially or reasonably relevant, and 3 if completely relevant to the query. The final relevance of a document to a query is the mean of the user's relevance scores. Table 2 presents the statistics of the documents in our test collection.

4.2 MWE Evaluation with NEs

We take twenty queries each belonging to one of the four complexity levels (Table 1) from the dataset in (Das et al., 2022) and our Gold Standard dataset to evaluate the effectiveness of grouping NE tokens with respect to mean normalized discounted cumulative gain (nDCG) and mean average precision (MAP). We present our results in Table 3. Using NE-enabled search has boosted the retrieval performance on both datasets i.e., (Das et al., 2022) and our dataset.

Earlier, for the queries like "বাংলাদেশ ইনস্টিটিউট অব ব্যাংক ম্যানেজমেন্ট"(EN: Bangladesh Institute of Bank Management), the documents with a high presence of the individual tokens "বাংলাদেশ" (EN: Bangladesh), "ইনস্টিটিউট" (EN: Institute), "অব" (EN: of), "ব্যাংক" (EN: Bank) and "ম্যানেজমেন্ট" (EN: Management) were prioritised over documents having the query words as a single unit. A query may not precisely contain the named entities as it is present in the relevant documents. So we indexed both the uni-gram tokens and the multiword NE tokens. The revised tokens formed for the example query are: "বাংলাদেশ ইনস্টিটিউট অব ব্যাংক ম্যানেজমেন্ট" (EN: Bangladesh Institute of Bank Management), "বাংলাদেশ" (EN: Bangladesh), "ইনস্টিটিউট" (EN: Institute), "অব" (EN: of), "ব্যাংক" (EN: Bank) and "ম্যানেজমেন্ট" (EN: Management).

4.3 Implicit Preference Feedback Results

To fully utilise the benefit of ESA, it is necessary that the concepts chosen are representative of the underlying text semantics. Since Bangla is a resource-constrained language, we could not find enough relevant concepts about health and travel from Wikipedia alone. We supplemented this gap with articles from reliable sources like Vikaspedia and various travel blogs. We represented the complete test collection using 9349 articles. Due to the absence of articles related to the literary works of Rabindranath Tagore, we have not used ESA on the 182 documents from Rabindranath Tagore's work.

We present the performance of Implicit preference feedback on 40 queries in Figure 2. We observed that implicit feedback on the initial results for queries from complexity levels 1 and 2 did not generate any interesting document in the revised result set. We expected this behaviour as the queries were clear and precise in intent and did not require any refinement in its vector representation. On the average, queries from complexity levels 3 and 4 produced improved results after

[14] https://github.com/huggingface/transformers/issues/876#issuecomment-514948425

[15] https://en.banglapedia.org/index.php/Sadhu_Bhasa

[16] https://en.banglapedia.org/index.php/Chalita_Bhasa

[17] https://bn.vikaspedia.in/

Parameters	Entire Test Collection	Entertainment	Health	International	Kolkata	National	Sports	Travel
Tokens (words)	433553	34629	91084	29506	25182	30699	31005	191444
Types (unique words)	53831	9707	14650	9436	7671	8927	8599	22622
Sentences	34553	3644	6439	2879	2814	2730	3214	12833
Average number of sentences per document	34.553	28.468	44.715	22.492	21.813	21.328	24.914	59.967
Average number of tokens per sentence	12.547	9.503	14.145	10.248	8.948	11.245	9.648	14.918
Average number of tokens per document	433.553	270.539	632.527	230.515	195.209	239.835	240.379	894.598

Table 2: Statistics of our Gold Standard Dataset

Dataset	Model	Mean nDCG@10	MAP@10	Mean Precision@10
(Das et al., 2022)	tf-idf	0.741	0.453	0.62
	tf-idf + NE tokens	0.842	0.519	0.66
Ours	tf-idf	0.76	0.57	0.37
	tf-idf + NE tokens	0.92	0.8	0.49

Table 3: Performance of Anwesha across different query complexity levels containing NEs.

1 and 2.3 iterations of query refinement using implicit preference feedback. Complexity level 4 queries required at most three iterations of query refinement. These queries were not a precise articulation of query intent. Hence, the query was initially not well represented in the concept space. After the user implicitly provided the IR system with the click-through information, the concept representation of the query improved.

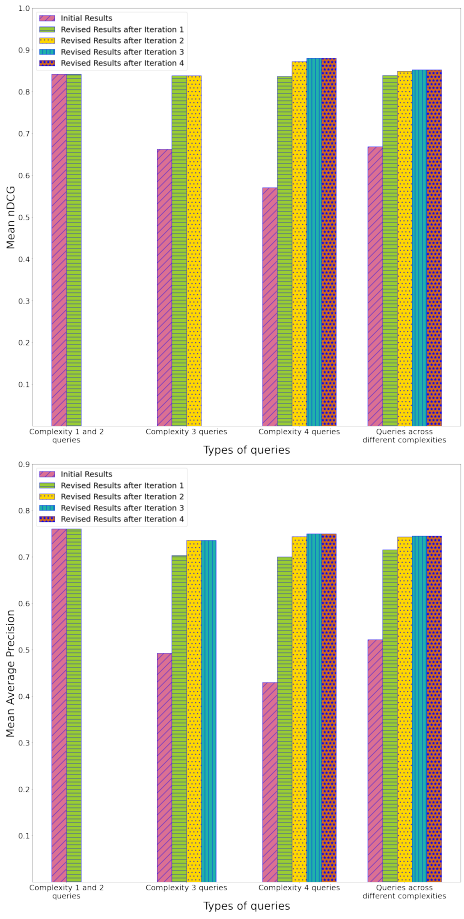

Figure 2: Results of Implicit Preference Feedback across different query complexity levels measured using mean nDCG and MAP @K=10.

4.4 Evaluation of LaBSE reranker using BIRCH approach

We present the results of applying LaBSE to document retrieval using the BIRCH approach in Table 4. We have studied the effect of LaBSE reranker on candidate documents retrieved by the tf-idf vector space algorithm with uniform weights ($w_1 = w_2 = w_3 = 1$) and weights set to cosine similarity score of the best sentences with the query in the concept space of ESA. Interestingly, we observe ESA weighted sentence scores perform the best. We present the top two documents retrieved using BIRCH in Figure 4 and the most relevant sentences related to the query "যমুনা নদীর দক্ষিণে সপ্তম আশ্চর্য্য"/ yamunā nadīra dakṣiṇe saptama āścaryya (EN: Seventh wonder to the south of Yamuna river). The query contains geospatial details that LaBSE could capture. So it picked the sentences from the documents related to the Taj Mahal with such information. The same query was issued in Sandhan (as of: 1-Oct-2022) with the intent to seek documents related to the "Taj Mahal"; the top 10 documents retrieved by Sandhan are not in line with the goal. Due to the Bag of words nature of Sandhan, the documents that contain a high presence of the individual terms "দক্ষিণ"/ dakṣiṇa(EN: south), "নদী"/ nadī(EN: river) and "যমুনা"/ yamunā(EN: Yamuna) are deemed to be given higher preference. The semantic relationships among the query words are not captured.

Figure 3 shows a snapshot of the revised user interface of Anwesha where the user can choose one of the six options (tf-idf, IndoWordNet based query expansion, LSA, ESA, LaBSE reranker on tf-idf with uniform sentence weights and LaBSE reranker on tf-idf with sentence weighted by ESA scores) for re-

Domain	Model	Mean nDCG@10	MAP@10	Mean precision@10
Travel	tf-idf	0.811	0.69	0.535
	ESA	0.824	0.716	0.56
	LaBSE reranker on tf-idf	0.908	0.82	0.635
	LaBSE reranker on tf-idf weighted by ESA scores	0.952	0.889	0.69
Health	tf-idf	0.772	0.621	0.38
	ESA	0.826	0.704	0.4
	LaBSE reranker on tf-idf	0.878	0.777	0.46
	LaBSE reranker on tf-idf weighted by ESA scores	0.912	0.825	0.47

Table 4: Performance of LaBSE as a reranker on tf-idf retrieved candidate documents.

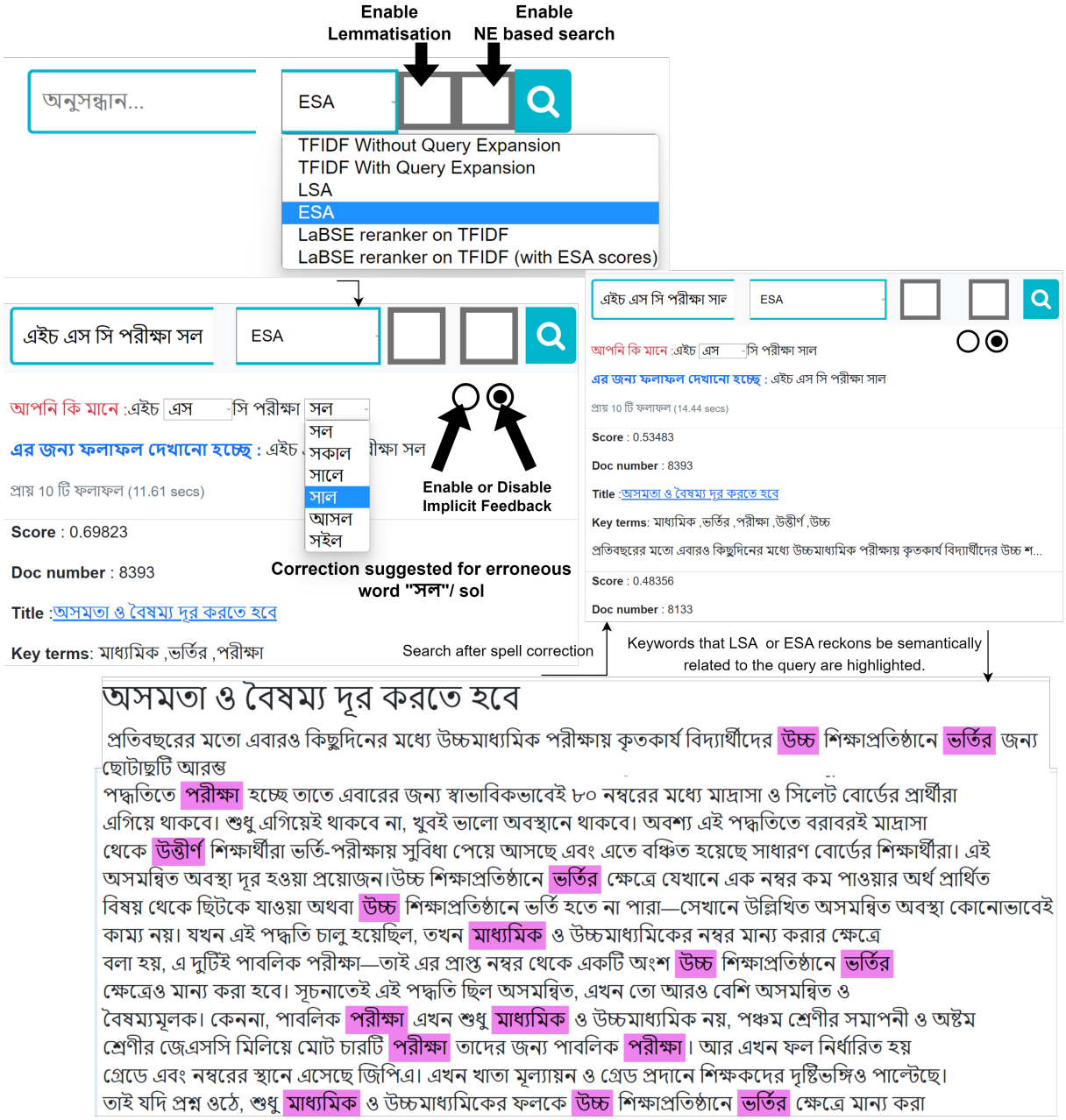

Figure 3: Revised user interface of Anwesha and explanation of search results by highlighting keywords

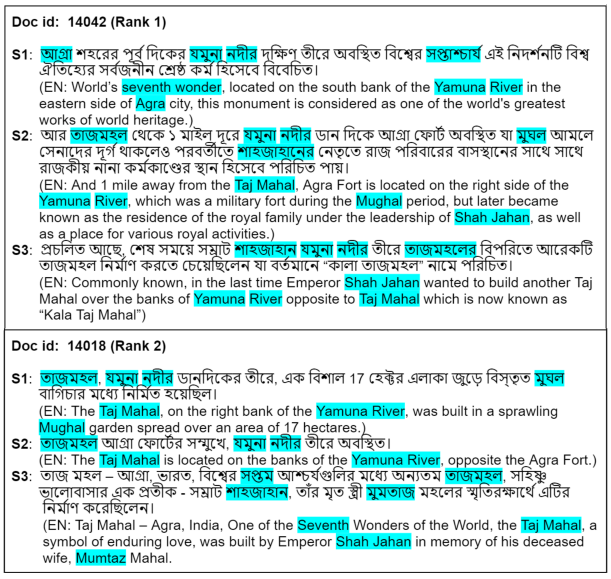

Figure 4: Best three sentences (S1, S2, S3) considered relevant by LaBSE in the top two retrieved documents ranked by the BIRCH approach. The words in a sentence deemed relevant to the query intent resolution by ESA are highlighted by the system.

trieval. The user can disable sending implicit feedback to the IR system, apply lemmatisation and NE based search. Anwesha receives the query "এইচ এস সি পরীক্ষা সল"/ ēica ēsa si parīkṣā sala(EN: HSC Exam Year) and performs spelling correction on the query word সল/ sala → সাল/ sāla. The search results are explained by highlighting the words relevant to the query in a top retrieved document. Such explanations are useful for techniques like LSA and ESA where the documents not having the query words are retrieved.

5 Conclusion and Future Work

We have enabled NE-based search to obtain single-unit NE tokens. To the best of our knowledge, ours is the first effort that delivers personalized results from diverse background knowledge sources via the clickthrough information of the user. We are also not aware of the past work that applies BERT models and it's advancement on Bangla IR. We have extended the previously existing Gold standard dataset for diverse evaluation of search results and used this to systematically study the improvement of Anwesha while addressing its current limitations. Our technique can inspire research in IR for other low-resource, highly inflected languages. As part of future work, we plan to handle different forms of MWEs like conjunct verbs (example "অনুভব করা"/ anubhaba karā (EN: to feel)), noun-verb collocations (example "খেতে যাওয়া"/khete yāōẏā (EN: go eat)), reduplicated terms (example "ছোট ছোট"/choṭa choṭa (EN: small small)), idiomatic compound nouns (example: "ভাই বোন"/bhāi bona (EN: brother sister)) etc (Chakraborty et al., 2014) (Dandapat et al., 2006). In future, we can incorporate actions like bookmarking, saving a document, and time of viewing a document as other contributors to determining user preferences. ESA is a recall-centric approach while LaBSE, as a reranker on the tf-idf vector space approach, is precision oriented. Studying the effect of the ESA cosine similarity scores used as surrogates to weigh the LaBSE sentence scores will be interesting. In some cases, where the queries were not precise, tf-idf could not include the relevant candidate documents in the initial pool of candidates. So LaBSE could not perform well on them, suggesting that a recall-centric algorithm could plausibly be used to determine the initial candidate set. Estimating the sentence-level relevance of a document to the query is a reasonable approach because LaBSE is trained with a sentence-level objective, making it suitable to find the relationship between a query sentence and a sentence from the document. However, in this approach, we are losing context information. Hence a more robust language model that could encode the "long" documents while effectively gauging their relevance to a "short" query would be a better alternative to our existing approach using BIRCH. In future, we plan to conduct our experiments over a wider range of queries. The dataset and code of our work are present here: https://github.com/ArupDas15/Revisiting_Anwesha.

Acknowledgements

This work would not have been possible without the contributions of the members of the Artificial Intelligence and Database Lab affiliated with the Department of Computer Science and Engineering, Indian Institute of Technology Madras Lokasis Ghorai and Arjun Kumar Gupta in the early stages of development of Anwesha.

References

Otavio Acosta, Aline Villavicencio, and Viviane Moreira. 2011. Identification and treatment of multiword expressions applied to information retrieval. In *Proceedings of the Workshop on Multiword Expressions: from Parsing and Generation to the Real World*, pages 101–109, Portland, Oregon, USA. Association for Computational Linguistics.

Zeynep Akkalyoncu Yilmaz, Shengjin Wang, Wei Yang, Haotian Zhang, and Jimmy Lin. 2019. Applying BERT to document retrieval with birch. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP): System Demonstrations*, pages 19–24, Hong Kong, China. Association for Computational Linguistics.

Rudramurthy. V Anoop Kunchukuttan Pratyush Kumar Mitesh Khapra Arnav Mhaske, Harshit Kedia. 2022. Naamapadam: A large-scale named entity annotated data for indic languages.

Pushpak Bhattacharyya. 2010. Indowordnet. In *Proceedings of the Seventh International Conference on Language Resources and Evaluation (LREC'10)*, pages 3785–3792.

Tanmoy Chakraborty, Dipankar Das, and Sivaji Bandyopadhyay. 2014. Identifying bengali multiword expressions using semantic clustering. *ArXiv*, abs/1401.6122.

Sandipan Dandapat, Pabitra Mitra, and Sudeshna Sarkar. 2006. Statistical investigation of bengali noun-verb (nv) collocations as multi-word-expressions. *Proceedings of Modeling and Shallow Parsing of Indian Languages (MSPIL)*, pages 230–233.

Arup Das, Bibekananda Kundu, Lokasis Ghorai, Arjun Kumar Gupta, and Sutanu Chakraborti. 2022. Anwesha: A tool for semantic search in bangla. https://github.com/ArupDas15/Anwesha. Accepted in The International Conference on Agglutinative Language Technologies as a challenge of Natural Language Processing; Conference date: 07-06-2022 Through 08-06-2022.

Suprabhat Das, Shibabroto Banerjee, and Pabitra Mitra. 2012. Anwesan: A search engine for bengali literary works. *World Digital Libraries*, 5(1):11–18.

Scott Deerwester, Susan T. Dumais, George W. Furnas, Thomas K. Landauer, and Richard Harshman. 1990. Indexing by latent semantic analysis. *Journal of the American Society for Information Science*, 41(6):391–407.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding.

Fangxiaoyu Feng, Yinfei Yang, Daniel Cer, Naveen Arivazhagan, and Wei Wang. 2020. Language-agnostic BERT sentence embedding. *CoRR*, abs/2007.01852.

Evgeniy Gabrilovich and Shaul Markovitch. 2007. Computing semantic relatedness using wikipedia-based explicit semantic analysis. In *IJCAI*.

Jeremy Howard and Sebastian Ruder. 2018. Universal language model fine-tuning for text classification.

Thorsten Joachims, Laura Granka, Bing Pan, Helene Hembrooke, and Geri Gay. 2005. Accurately interpreting clickthrough data as implicit feedback. In *Proceedings of the 28th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '05, page 154–161, New York, NY, USA. Association for Computing Machinery.

Divyanshu Kakwani, Anoop Kunchukuttan, Satish Golla, Gokul N.C., Avik Bhattacharyya, Mitesh M. Khapra, and Pratyush Kumar. 2020. IndicNLPSuite: Monolingual Corpora, Evaluation Benchmarks and Pre-trained Multilingual Language Models for Indian Languages. In *Findings of EMNLP*.

Redwanul Karim, M. A. Muhiminul Islam, Sazid Rahman Simanto, Saif Ahmed Chowdhury, Kalyan Roy, Adnan Al Neon, Md Hasan, Adnan Firoze, and Rashedur M. Rahman. 2019. A step towards information extraction: Named entity recognition in bangla using deep learning. *J. Intell. Fuzzy Syst.*, 37:7401–7413.

KPMG. 2017. Indian languages- defining india's internet. https://assets.kpmg/content/dam/kpmg/in/pdf/2017/04/Indian-languages-Defining-Indias-Internet.pdf, Accessed: 2022-09-09.

Anoop Kunchukuttan, Divyanshu Kakwani, Satish Golla, Gokul N.C., Avik Bhattacharyya, Mitesh M. Khapra, and Pratyush Kumar. 2020. Ai4bharat-indicnlp corpus: Monolingual corpora and word embeddings for indic languages. *arXiv preprint arXiv:2005.00085*.

Jimmy Lin. 2019. The neural hype and comparisons against weak baselines. *SIGIR Forum*, 52(2):40–51.

Christopher D. Manning, Prabhakar Raghavan, and Hinrich Schütze. 2008. *Introduction to Information Retrieval*. Cambridge University Press.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient estimation of word representations in vector space.

P. Pandurang Nayak. 2019. Understanding searches better than ever before. https://blog.google/products/search/search-language-understanding-bert/. Accessed: 2022-08-03.

Xiaoman Pan, Boliang Zhang, Jonathan May, Joel Nothman, Kevin Knight, and Heng Ji. 2017. Cross-lingual name tagging and linking for 282 languages. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1946–1958, Vancouver, Canada. Association for Computational Linguistics.

Jeffrey Pennington, Richard Socher, and Christopher Manning. 2014. GloVe: Global vectors for word representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543, Doha, Qatar. Association for Computational Linguistics.

Matthew E. Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep contextualized word representations.

Pattisapu Nikhil Priyatam, Srikanth Reddy Vaddepally, and Vasudeva Varma. 2012. Domain specific search in indian languages. In *Proceedings of the First Workshop on Information and Knowledge Management for Developing Region*, IKM4DR '12, page 23–30, New York, NY, USA. Association for Computing Machinery.

Yifan Qiao, Chenyan Xiong, Zhenghao Liu, and Zhiyuan Liu. 2019. Understanding the behaviors of bert in ranking.

Filip Radlinski and Thorsten Joachims. 2005. Query chains: Learning to rank from implicit feedback. In *Proceedings of the ACM Conference on Knowledge Discovery and Data Mining (KDD)*. ACM.

Punya Sloka Ray, Muhammad Abdul Hai, and Lila Ray. 1966. *Bengali Language Handbook*. Center for Applied Linguistics.

Nils Reimers and Iryna Gurevych. 2019. Sentence-bert: Sentence embeddings using siamese bert-networks. *arXiv preprint arXiv:1908.10084*.

Sagor Sarker. 2021. Bnlp: Natural language processing toolkit for bengali language. *ArXiv*, abs/2102.00405.

Ryen White, Ian Ruthven, and Joemon Jose. 2002. The use of implicit evidence for relevance feedback in web retrieval. pages 93–109.

Jeffrey Zhu. 2019. Bing delivers its largest improvement in search experience using azure gpus. https://azure.microsoft.com/en-us/blog/bing-delivers-its-largest-improvement-in-search-experience-using-azure-gpus/. Accessed: 2022-08-21.