Effects of Query Correction Techniques on Ranking Systems

Tom Aarsen
s1027401
The Netherlands

Tijn Berns
s1027659
The Netherlands

Daan Derks
s1011515
The Netherlands

ABSTRACT

To resolve misspellings in a query, this paper applies a dictionary lookup technique, a word modification technique and context-dependent word corrector. These techniques are used on data retrieved from the document ranking task of the 2021 version of the TREC Deep Learning Track [2, 3, 14]. Two experiments are performed on the document ranking of a query, one to compare the different techniques and one to establish the performance gain or loss when applying a technique. One of the main findings of these experiments is that the techniques are trying to replace entities like names, locations and abbreviations to English words that the technique is aware of. This results in a loss of performance in the ranking.

1 INTRODUCTION

Search engines like Google and Bing are of great importance when it comes to finding and accessing information on the web. The ever-increasing performance of these systems has allowed for the outstanding progress in science. However, queries containing misspellings may pull up documents that the user would deem irrelevant. Research has shown that incorporation a spelling correction mechanism can be beneficial for the results returned by search engines [11].

This research addresses this potential problem by exploring the effects of applying different spelling error correction techniques on queries. To be more specific, this research answers the following research question: What is the effect of spelling error correction of queries on the *MAP*@10, *NDCG*@10, and *MRR*@10 score using the BM25 as the retrieval algorithm? By answering the research question we give a potential direction in which information retrieval systems, like search engines, can be improved.

In this paper, we apply three different methods of spelling correction techniques on data from the 2021 version of the TREC Deep Learning Track[2, 3, 14]. The three methods are a dictionary lookup technique (Section 3.1), a word modification technique and a context-dependent word corrector (Section 3.2). We perform two experiments (Section 4.3) for each of the techniques, allowing us to both compare different techniques with each other, and establish the performance gain or loss when applying a certain technique.

This paper is structured as follows: First, the most relevant related work is covered. Second, a description of the methodology is given. After this, we describe the experimental setup, giving a detailed description of the data that is used and the experiments that are conducted. Finally, we discuss the results from the experiments and give a direction for future work.

2 RELATED WORK

Correcting spelling errors in search queries has been considered in recent research related to information retrieval systems. In this section we cover some of the most important related work w.r.t. spelling errors and their correction in general, and w.r.t spelling correction in search queries.

2.1 Spelling Correction

Spelling correction in general is a complex task. Often, there exist different ways to interpret a sentence and different meanings for the same words. This makes it hard to check whether a sentence, or query, is grammatically and spelled correct. The complex nature of the problem results in active research in this field. Many different techniques have been considered and are under consideration for correcting spelling errors. Closely related to our work, is the dictionary lookup technique. This method compares potentially erroneous words in a sentence with a dictionary, a lexicon, a corpus, or a combination of these collections. Depending on the implementation of the correction method, the resulting collection either consists of existing words, or of known spelling errors [13]. The technique is extremely cost efficient. However, it is challenging to select the best replacement for a misspelled word.

The dictionary lookup technique requires a substantial corpus, and primarily tackles the first of the three progressively difficult problems as described by Kukich [10]: (1) nonword error detection, (2) isolated-word error correction and (3) context-dependent word correction. According to Kukich, the first problem mainly revolves around efficient string comparison techniques for deciding whether a word appears in a dictionary. The second problem involves correction techniques, including studies on spelling errors. Lastly, studies that try to solve the last and thus most difficult problem attempt to use contextual information to aid spelling correction.

2.1.1 Common spelling errors. As per Kukich's second problem [10], studies on spelling errors have been performed to find root causes of spelling errors, in the hopes to improve spelling correction. Damerau [4] performed such a study, showing that 80% of all spelling errors involve exactly one insertion, deletion, substitution or transposition.

2.1.2 The state of the art. Like Kukich predicted, the state-of-the-art spelling correction tools are the ones that can tackle the most difficult spelling correction problem of context-dependent word correction. One such toolkit is NeuSpell [9], which contains ten different models, including a spelling corrector based on BERT [6]. These models are explicitly trained using contextual representations, which proved a key factor in allowing NeuSpell models to outperform existing spelling correction tools.

2.2 Errors in Search Queries

Fixing spelling errors in search queries brings additional challenges compared to traditional spelling correction. New search queries emerge constantly as new research and news messages are published. This makes it less useful to use a human-compiled lexicon for spelling correction. Therefore, previous research has focused on

using Web corpora and logs instead [15]. Other methods adapting to the rapidly changing spelling in search queries focus on training n-gram models capturing user behavior [1, 7].

3 METHOD

The main problem this research aims to solve is how to improve generated rankings for search queries containing spelling errors. We tackle this problem by applying three different spelling correction techniques, one for each of the three progressively difficult problems described by Kukich [10].

For each of the techniques we check the performance on a index built on large corpus of documents. Having this index, we can rank the our different sets of queries using the `Anserini` implementation of BM25. For a baseline we apply BM25 on the queries where no spelling correction is applied. Using this baseline we are able to check what the effects are of applying the spelling correction technique on the quality of the ranking.

3.1 Dictionary Lookup Technique

The first approach to the problem is applying the dictionary lookup technique. This technique corresponds to the first level of problems mentioned in Section 2.1, nonword error detection. For our implementation we used multiple collections of spelling errors with corresponding correct spellings. The misspellings can have one or multiple different correct spellings. The first suggested correction is usually the most common. This rule-based spelling error correction is applied on every single word of every query individually. We check whether the word is in the collection. If this is the case, we replace the misspelled word with the first suggested correctly spelled word. Eventually, there are no queries that contain any of the common misspellings.

3.2 Existing Spelling Correction Tools

A different approach to the problem is to correct spelling errors in queries using existing spelling correction tools. We considered two tools: The Python `autocorrect` module and `BertChecker` from Neuspell [9].

The Python `autocorrect`[1] module creates a set of all possible words that are constructed by applying two alterations for every word in a given query. Where one alterations is either a deletion, a transposition, a substitution, or an insertion. The module corrects a word in the query by replacing it with the most common existing word in the constructed set according to a dictionary of words and word counts. The module is based on the fact that the four alterations account for approximately 80% of the spelling errors [4]. Therefore, this module corresponds to the second level of problem difficulty mentioned in Section 2.1, isolated error correction.

`BertChecker` from Neuspell [9] implements BERT [6], which is a state of the art transformer based machine learning technique for natural language processing. When correcting spelling errors, BERT takes the context of the query into account. Therefore, this spelling correction technique corresponds the the third level of problem difficulty mentioned in Section 2.1, context-dependent word correction. Take for example a misspelling of the word `'boat'`, e.g. `'bot'`. If the query containing this misspelling contain words contextually related to boats, it is more likely that the word the word should be corrected to `'boat'`, than that it should be corrected to `'boot'` or `'bat'`.

[1] https://github.com/filyp/autocorrect

3.3 Evaluation Metrics

For evaluating generated rankings we use three metrics: Mean Average Precision (*MAP*), Normalized Discounted Cumulative Gain (*NDCG*) and Mean Reciprocal Rank (*MRR*). *MAP* takes the mean over the average precision (*AP*) of all documents. *AP* is applied on a single query:

$$AP@K = \sum_{k=1}^{K} (Recall@k - Recall@k-1) \cdot Precision@k \tag{1}$$

$$MAP@K = \frac{1}{K} \sum_{k=1}^{K} AP@k \tag{2}$$

where K is the top-K number of ranked documents by our algorithm given a query.

NDCG is the normalized version *DCG*, which divides the *DCG* by the ideal *DCG* (*IDCG*). *DCG* is used to measure the usefulness of a document given their position in the ranking.

$$DCG@K = \sum_{i=1}^{K} \frac{rel@i}{log_2(i+1)} \tag{3}$$

$$IDCG@K = \sum_{i=1}^{|rel@K|} \frac{rel@i}{log_2(i+1)} \tag{4}$$

$$NDCG@K = \frac{DCG@K}{IDCG@K} \tag{5}$$

where $rel@K$ is the ideal top-K relevant documents in the corpus given a query.

The third metric is *MRR*, where we take the mean of the reciprocal rank. The reciprocal rank of a query is the multiplicative inverse of the rank in our ranking of the actual most relevant document.

$$MRR@K = \frac{1}{K} \sum_{i=1}^{K} \frac{1}{rank_i} \tag{6}$$

where $rank_i$ is the rank in our ranking of the actual most relevant document.

4 EXPERIMENTAL SETUP

This section describes the experimental setting of the experiments that have been conducted. First, a description of the considered datasets and how they are used in the experiments is given. After this, we describe how we generated rankings for our experiments, and how these rankings are evaluated.

4.1 Data

In all of the experiments we at least make use of the following three datasets, all extracted from the document ranking task of the 2021 variant of the TREC Deep Learning Track [2, 3, 14]. The three datasets are a documents dataset, a dataset of queries, and the corresponding QREL data. For the document dataset we use the MS MARCO dataset. This is a corpus consisting of 11,959,635 records, adding up to a total of 32.3 GB of document data. The query

set corresponding to MS MARCO, consists of 322,196 records. The corresponding QREL data consists of 331,956 records.

For the experiments w.r.t. the dictionary lookup technique we make use of four additional datasets, or dictionaries, containing spelling errors and their corresponding corrected words. With these four dictionaries we consider different spelling errors collected from various sources. The dictionaries are as follows:

(1) **Aspell:** Consists of 531 misspellings of 240 different words, and thereby is the smallest dictionary we use. The misspellings are collected for testing the Aspell spellchecker, the default spellchecker of GNU operating system. The paper "Correcting spelling errors by modelling their causes" [5], is closely related to this spellchecker.
(2) **Holbrook:** This dictionary contains 1,791 misspellings of 1,200 words. The misspellings in this set are taken from the book “English for the Rejected” [8], which contains writings of secondary school children in their next-to-last year. Note that this book is from 1,964 and therefore does not contain any modern terms.
(3) **Birkbeck:** Consists of 36,133 misspellings of 6,136 words, and thereby is the largest dictionary we use. The misspellings are taken from the Birkbeck spelling error corpus[12]. This corpus is a collection of errors taken from free writing, mostly of school children, university students, and adult literacy students. It must be noted that this dictionary contains misspellings which are very different from their target word.
(4) **Wikipedia:** This dictionary contains 2,455 spelling errors of 1,922 different words. The list of misspellings has been constructed by Wikipedia editors [2].

When applying the dictionary lookup technique we use every possible combination of the four spelling error datasets as a dictionary to fix the original query set. This results in 15 dictionaries, which when applied to the query set, gives 15 different sets of corrected queries.

4.2 Ranking & Evaluation

In order to be able to generate a ranking we must first build an index of our document dataset, MS MARCO. The indexing is done using `Anserini`. All parameters of the build in method are set to the default values. The generated index is used in the BM25 retrieval algorithm from `Anserini` to generate a ranking of ten documents for all queries in the considered query set. We first do this on the original query set to create a ranking serving as the baseline. This ranking will be denoted be $\mathcal{R}_b$. After generating $\mathcal{R}_b$, we generate a ranking for the 15 dictionary based corrected query sets, for the query set corrected using `BERT`, and for the query set corrected using `autocorrect`. Hence, in total, we generate 18 different rankings.

We evaluate the quality of the rankings using the *MAP*@10, the *NDCG*@10, and the *MRR*@10 as described in Section 3.3. These metrics together give a good representation of the quality of the given ranking.

[2]https://en.wikipedia.org/wiki/Wikipedia:Lists_of_common_misspellings

4.3 Experiments

The goal of this research is to test the effect of applying different spelling correction techniques to correct queries on the quality of generated rankings. To achieve this, we designed two experiments which we conduct on each of 15 fixed query sets from Section 4.1, as well as the two fixed query sets derived by applying the two spelling correction tools from Section 3.2 on the original query set. The code that is used for the experiments can be found at GitHub[3]. The two experiments are as follows:

(1) We generate a ranking for the set of fixed queries including queries that have not been corrected by the spelling correction technique. We compare the quality of the resulting ranking with that of $\mathcal{R}_b$ using *MAP*@10, *NDCG*@10, and *MRR*@10 (Section 3.3). This experiment allows us to compare the quality of rankings when using different dictionaries, and different spelling correction techniques.
(2) We generate a ranking for the set of fixed queries, in which we do not consider queries that have not been altered using the spelling correction technique. We compare this ranking with the subset of $\mathcal{R}_b$ corresponding to the set of altered queries. The comparison is done by considering the increase in *MAP*@10, *NDCG*@10, and *MRR*@10 relative to $\mathcal{R}_b$, defined as Δ*MAP*@10, Δ*NDCG*@10, and Δ*MRR*@10, respectively. This experiment gives us a better understanding how the different spelling correction techniques affect the quality of the rankings.

5 RESULTS AND DISCUSSION

We applied three types of techniques in our research: a dictionary lookup technique, a more established technique involving applying insertions, deletions, substitutions and transpositions implemented in `autocorrect`, and a state-of-the-art BERT-based spelling corrector from NeuSpell[9]. As mentioned in Section 4.3, we conducted two experiments for each of these techniques:

5.1 Results of Experiment 1

In our first experiment we ranked all queries using BM25, allowing us to compare the global performance of the different spelling correction techniques on the entire query set.

5.1.1 Dictionary lookup technique. The results of experiment (1) applied on all 15 query sets generated via the dictionary lookup technique can be found in Appendix A, Table 1. Note that this table has 16 entries, as the top row represents the evaluation scores of BM25 on the original queries. The remainder of the 15 rows represent all possible combinations of datasets used in the dictionary lookup technique. The cells containing the best scores have been given a darker background. For all evaluation metrics, a higher score is better.

In this table, it is immediately apparent that the first row, containing the scores of unchanged queries, has the highest values. These scores are equaled by using Wikipedia data, using Aspell data, or by using both of those corpora in the dictionary lookup method. Table 3 helps explain why this might be - it shows that only 1534, 1661 and 1887 out of 322,197 queries were modified using these corpora,

[3]https://github.com/tomaarsen/IRSpellingCorrection

respectively. Regardless of how the dictionary lookup technique modifies the performance for these queries, this effect would be too small to be noticeable in the evaluation scores over the entire query set. Experiment (2) will help give more details on the effect of the dictionary lookup technique, as it only considers the scores of the modified queries.

Beyond the small differences between the top 4 dataset combinations in Appendix A, Table 1, we can also see that any combination of dictionaries containing Birkbeck results in a poor ranking. One major difference of Birkbeck relative to Aspell, Holbrook and Wikipedia is that it is by far the largest dictionary (36,133 misspellings). Table 3 shows that Birkbeck modifies 15,509 queries, much more than Holbrook, Aspell and Wikipedia, which individually modify 3004, 1661 and 1534 queries, respectively. Whenever a query is modified, it can happen that a word deemed as a misspelling is actually existing English word. For example, Birkbeck claims that the word `'effect'` is a misspelling of the word `'affect'`, while both are correct words on their own. Which word ought to be used depends on the context, which is the most difficult type of word misspelling problem according to Kukich [10].

Another explanation of the poor performance of Birkbeck could be that it consist of misspellings of school children, meaning that some misspellings can be very different from the associated correct word. Such large misspellings are very unlikely to re-occur in the exact same way in practice [4].

5.1.2 Existing spelling correction tools. We have applied the same experiment (1) to the two query sets obtained by applying the existing spelling correction tools, resulting in Table 4 in Appendix B. These spelling correction methods are performing much worse than the dictionary lookup technique from Table 4. That said, these methods also affect significantly more queries (See Table 6), which helps to explain the significant decrease in performance.

We believe the poor performance of `autocorrect` can be explained by that its corpus of known words is simply too small. It contains roughly 91000 words, gathered from Wikipedia data, and will frequently change one valid English word into another valid English word. For example, in the first 10 queries alone, `autocorrect` modifies into `'fixes'`, `'clots'` into `'lots'` and `'firefly'` into `'briefly'`.

The BERT-based spelling correction by NeuSpell seems to fall victim to this, too, although to a lesser extent. It will also commonly modify entities such as names or locations into nouns. For example, `'rice eccles stadium'` is converted to `'race reckless stadiums'`, `'texas'` to `'taxes'`, `'maryland'` into `'mainland'`, `'maranda name'` becomes `'veranda name'` and `'perez hilton'` is changed to `'per religion'`.

Here we can tell that applying a general-purpose spelling corrector does not perform well for search queries that contain entities. We believe that there is a lot of performance to be gained here.

5.2 Results of Experiment 2

In our second experiment we ranked only the modified queries per technique. This experiment gives us more insight in how the specific spelling correction techniques affect the rankings, because every technique modifies a different amount of queries (Table 3 and Table 6).

5.2.1 Dictionary lookup technique. Table 2 contains the differences between the baseline and all the Dictionary Lookup combinations. Since all values int this table (with the exception of one) are negative, we can conclude that the dictionary look-up technique does not result in better rankings. Queries corrected by the dictionary constructed by Wikipedia come closest to the original queries. The $\Delta NDCG@10$ score of 0.0006 is a bit higher than the baseline. This can mean that by applying the Wikipedia dataset the ranking system can rank the most relevant documents higher than the also relevant, but less relevant documents. The dictionary lookup table of Wikipedia is of much higher quality, because the misspellings are aggregated from many resolved misspellings on Wikipedia, as opposed to just one school child who made an error once. We believe this is why Wikipedia outperforms the other corpora.

Aspell, Holbrook, and Birkbeck all performed worse than the baseline. Out of these three the ranking based on queries fixed using the Birkbeck dataset has the lowest quality. This is not unexpected, as we saw similar results in the first experiment.

5.2.2 Existing spelling correction tools. We conducted the same experiment using the considered existing spelling correction tools. Table 5 contains the differences in scores between the baseline ranking and the rankings fixed using `BERT` or `autocorrect`. We can see that both tools result is worse rankings than the baseline. The argumentation to why this might be the case is the same as the argumentation given in Section 5.1.2; Both techniques often change correct words into other existing words, and thereby change the context of the query.

5.3 Conclusion

This research aimed to test the effects of applying different spelling correction techniques on search queries and their produced rankings. We considered three different techniques, one for each of the problem levels as described by Kukich [10]: (1) nonword-error detection, (2) isolated-word error correction, and (3) context-dependent word correction. Unfortunately, none of the considered techniques resulted in better rankings. The main issue encountered by all spelling correction techniques was that new terms, names, and abbreviations make it hard to apply spelling correction on the queries. We noticed that the correction techniques often change these terms, and other already existing words, to words which are more commonly used. By making these incorrect changes, the context of the query might completely change, and therefore the quality of the ranking might be much worse.

This problem could be countered by using different techniques to check for spellings errors in search queries. Future work could tackle this problem by using knowledge graphs. By incorporating knowledge graphs we have information about names of entities, abbreviations, and newer terms. This could then be combined with a technique focusing heavily on the context of the query.

In conclusion, the $MAP@10$, $NDCG@10$, and $MRR@10$ scores of queries ranked by BM25 will decrease when corrected using an entity-unaware spelling error correction technique.

REFERENCES

[1] Thorsten Brants, Ashok Popat C, Peng Xu, Franz J Och, and Jeffrey Dean. 2007. Large language models in machine translation. (2007).

[2] Nick Craswell, Bhaskar Mitra, Emine Yilmaz, and Daniel Campos. 2021. Overview of the TREC 2020 deep learning track. (2021).

[3] Nick Craswell, Bhaskar Mitra, Emine Yilmaz, Daniel Campos, and Ellen M. Voorhees. 2020. Overview of the TREC 2019 deep learning track. (2020).

[4] Fred J. Damerau. 1964. A Technique for Computer Detection and Correction of Spelling Errors. *Commun. ACM* 7, 3 (1964), 171–176.

[5] Sebastian Deorowicz and Marcin Ciura. 2005. Correcting spelling errors by modelling their causes. *International Journal of Applied Mathematics and Computer Science* 15, 2 (2005), 275–285.

[6] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. (2019).

[7] Huizhong Duan and Hsu (Paul) Bo-June. 2011. Online Spelling Correction for Query Completion. In *Proceedings of the 20th International Conference on World Wide Web*. 117–126.

[8] David Holbrook. 1964. *English for the Rejected.* Cambridge University Press.

[9] Sai Muralidhar Jayanthi, Danish Pruthi, and Graham Neubig. 2020. NeuSpell: A Neural Spelling Correction Toolkit. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*. Association for Computational Linguistics, Online, 158–164.

[10] Karen Kukich. 1992. Techniques for Automatically Correcting Words in Text. *ACM Comput. Surv.* 24, 4 (1992), 377–439.

[11] Bruno Martins and Mario J. Silva. 2004. Spelling Correction for Search Engine Queries. In *Advances in Natural Language Processing*. Springer Berlin Heidelberg, 372–383.

[12] Roger Mitton. [n. d.]. Birkbeck spelling error corpus. Oxford Text Archive.

[13] Joseph J Pollock and Antonio Zamora. 1984. Automatic spelling correction in scientific and scholarly text. *Commun. ACM* 27, 4 (1984), 358–368.

[14] Ellen M Voorhees, Donna K Harman, et al. 2005. *TREC: Experiment and evaluation in information retrieval*. Vol. 63. MIT press Cambridge, MA.

[15] Casey Whitelaw, Ben Hutchinson, Grace Y Chung, and Gerard Ellis. 2009. Using the Web for Language Independent Spellchecking and Autocorrection. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing*. 890–899.

A RESULTS DICTIONARY LOOKUP

A	H	B	W	*MAP*@10	*NDCG*@10	*MRR*@10
0	0	0	0	0.1323	0.1689	0.1341
0	0	0	1	0.1323	0.1689	0.1341
0	0	1	0	0.1301	0.1662	0.1319
0	0	1	1	0.1301	0.1662	0.1319
0	1	0	0	0.1322	0.1687	0.1340
0	1	0	1	0.1322	0.1687	0.1340
0	1	1	0	0.1300	0.1660	0.1318
0	1	1	1	0.1300	0.1660	0.1318
1	0	0	0	0.1323	0.1689	0.1341
1	0	0	1	0.1323	0.1689	0.1341
1	0	1	0	0.1301	0.1661	0.1319
1	0	1	1	0.1301	0.1661	0.1319
1	1	0	0	0.1321	0.1686	0.1339
1	1	0	1	0.1321	0.1686	0.1339
1	1	1	0	0.1300	0.1660	0.1318
1	1	1	1	0.1300	0.1660	0.1318

Table 1: Evaluation scores of BM25 rankings based on queries corrected using the dictionary lookup technique. A is for Aspell, H for Holbrook, B for Birkbeck and W for Wikipedia.

A	H	B	W	Δ*MAP*@10	Δ*NDCG*@10	Δ*MRR*@10
0	0	0	0	0	0	0
0	0	0	1	-0.0003	0.0006	-0.0003
0	0	1	0	-0.0456	-0.0574	-0.0459
0	0	1	1	-0.0451	-0.0568	-0.0454
0	1	0	0	-0.0184	-0.0233	-0.0183
0	1	0	1	-0.0174	-0.0213	-0.0173
0	1	1	0	-0.0447	-0.0561	-0.0450
0	1	1	1	-0.0443	-0.0556	-0.0446
1	0	0	0	-0.0102	-0.0141	-0.0103
1	0	0	1	-0.0096	-0.0126	-0.0097
1	0	1	0	-0.0452	-0.0571	-0.0455
1	0	1	1	-0.0448	-0.0564	-0.04511
1	1	0	0	-0.0213	-0.0275	-0.0213
1	1	0	1	-0.0205	-0.0259	-0.0205
1	1	1	0	-0.0445	-0.0558	-0.0447
1	1	1	1	-0.0441	-0.0553	-0.0443

Table 2: Difference of evaluation scores on the subsection of the queries that were modified by the dictionary lookup technique, relative to the same subsection of unmodified queries. A is for Aspell, H for Holbrook, B for Birkbeck and W for Wikipedia.

A	H	B	W	Nr. modified queries
0	0	0	0	0
0	0	0	1	1534
0	0	1	0	15509
0	0	1	1	15651
0	1	0	0	3004
0	1	0	1	3238
0	1	1	0	16619
0	1	1	1	16759
1	0	0	0	1661
1	0	0	1	1887
1	0	1	0	15765
1	0	1	1	15892
1	1	0	0	3383
1	1	0	1	3589
1	1	1	0	16874
1	1	1	1	16999

Table 3: Number of modified queries using the dictionary lookup technique, out of a total of 322197 queries. A is for Aspell, H for Holbrook, B for Birkbeck and W for Wikipedia.

B RESULTS EXISTING SPELLING CORRECTION TOOLS

Spelling correctors	*MAP*@10	*NDCG*@10	*MRR*@10
`autocorrect`	0.0939	0.1203	0.0953
`BERT`	0.0876	0.1127	0.0889

Table 4: Evaluation scores of BM25 rankings based on queries corrected using the existing spelling correction tools.

Spelling correctors	Δ*MAP*@10	Δ*NDCG*@10	Δ*MRR*@10
`autocorrect`	-0.1543	-0.1954	-0.1560
`BERT`	-0.0998	-0.1253	-0.1008

Table 5: Difference of evaluation scores on the subsection of the queries that were modified by the existing spelling correction tools, relative to the same subsection of unmodified queries.

Spelling correctors	Nr. modified queries
`autocorrect`	80161
`BERT`	144476

Table 6: Number of modified queries using the existing spelling correction tools, out of a total of 322197 queries.