A Large-Scale Query Spelling Correction Corpus

Matthias Hagen Martin Potthast Marcel Gohsen Anja Rathgeber Benno Stein
Bauhaus-Universität Weimar
99421 Weimar, Germany
⟨firstname⟩.⟨lastname⟩@uni-weimar.de

ABSTRACT

We present a new large-scale collection of 54,772 queries with manually annotated spelling corrections. For 9,170 of the queries (16.74%), spelling variants that are different to the original query are proposed. With its size, our new corpus is an order of magnitude larger than other publicly available query spelling corpora. In addition to releasing the new large-scale corpus, we also provide an implementation of the winner of the Microsoft Speller Challenge from 2011 and compare it on the different publicly available corpora to spelling corrections mined from Google and Bing. This way, we also shed some light on the spelling correction performance of state-of-the-art commercial search systems.

1 INTRODUCTION

Query spelling correction is an important step of the query understanding process at search engine side. When a query is submitted, it is usually first tokenized and "normalized" (e.g., lowercasing), directly followed by a spelling correction. After that, the query might be lemmatized/stemmed, entities might be detected, etc. However, these subsequent steps of understanding a user's query heavily rely on good spelling (e.g., entities with wrong spelling can be very difficult to accurately detect). Thus, spelling correction for queries attracted a lot of attention, both within the Microsoft Speller Challenge 2011 [22] and in subsequent publications on participating approaches as well as improved versions thereof.

Today, commercial search engines typically offer corrections even while the user is typing [5], and they correct misspelled queries very reliably, asking "Did you mean [alternative spelling]," or even directly "Showing results for" their best guess. However, not too many details about the underlying systems are published. Instead, academic research on improved spelling correction algorithms still has to rely on only two publicly available corpora with about 6,000 annotated queries each (16–19% with spelling variants different to the original query), one being a training set of the mentioned Microsoft Speller Challenge 2011 [22], the other being published by the third-placed team who used it as an additional training set for the challenge [7]. To offer an alternative, large-scale resource, we release a corpus of 54,772 web search queries, out of which 16.74% come with spelling variants different to the original query.[1] Along the corpus, we also release the code[2] of a re-implementation of the best-performing approach from the Microsoft Speller Challenge [14] and compare it on our new corpus and the two other publicly available ones against spelling corrections mined from Google's and Bing's search engines.

The analysis of the results shows that our new corpus is a little harder for the spelling correctors, with Precision@1 scores dropping by about 5–10%. Only the Google spelling correction performs better than a baseline that does not change the input query at all. Thus, our new corpus offers a challenging alternative to the two existing corpora. Our re-implementation of Lueck's approach [14], who achieved the best performance within the Microsoft Speller Challenge, also struggles to beat the baseline. This indicates that the version that participated in the challenge probably heavily relied on not fully documented optimizations against the challenge's evaluation metrics that might not help in real-world situations.

2 RELATED WORK

Query spelling correction has been a lively research topic since the mid 2000's, especially in the NLP community [1, 4, 11]. Back in that time an (in)famous slide from some Google presentation presented literally hundreds of misspellings of the then-celebrity Britney Spears (or Brittany Spiers?!).

Most systems for spelling correction from that time (and still today) are based on language models for the a priori probabilities of words and an error model (e.g., noisy channel) to estimate probabilities of misspellings [16]. Especially due to the error models trained on the input of billions of users, today's commercial search engines can provide a spelling performance that seems to "magically" second guess the intended query for most misspellings. In particular, today's search engines go as far as to suggest corrections even while the user is still typing [5], or they try to avoid user misspellings at all by sensible query auto-completions without errors [2], which is also still an ongoing research problem [10].

The problem of query spelling correction attracted a lot of attention around 2010, with the Microsoft Speller Challenge organized in the year 2011 [22] having more than 300 teams participating. With this challenge, a large public set of 5,892 spell-corrected queries sampled from the TREC Million Query track was released for training. The best-performing approach of Gord Lueck [14], based on combining Hunspell[3] suggestions, was followed by Cloudspeller [12] and qSpell [7]. Also the ideas of the other top-performing participants [15, 17, 20] influenced approaches published later [3, 6, 9, 13, 19, 21], indicating that query spelling is not "solved" yet.

[1] https://www.uni-weimar.de/medien/webis/corpora/
[2] https://github.com/webis-de/SIGIR-17
[3] http://hunspell.github.io/

SIGIR '17, August 07-11, 2017, Shinjuku, Tokyo, Japan

978-1-4503-5022-8/17/08…$15.00
DOI: http://dx.doi.org/10.1145/3077136.3080749

Still, there are only two publicly available larger query corpora with annotations of potential alternative spellings. First, the aforementioned set of 5,892 queries published by the Microsoft Speller Challenge [22], and a set of 6,000 queries which the qSpell team released as their additional training set [7]. To further add to the publicly available corpora, we publish a set of 54,772 queries and possible alternative spellings for 9,170 of them.

3 OUR NEW CORPUS

We detail the sampling and annotation process of our corpus and compare it to the mentioned other two publicly available ones with regard to error types, error frequencies, etc.

3.1 Query Sampling

For the creation of a previous query segmentation corpus [8], we had sampled 55,555 queries with 3 up to 10 words (i.e., with 2 up to 9 whitespaces) from the AOL query log [18] in three steps: (1) the raw query log was filtered in order to remove ill-formed queries, (2) from the remainder, queries were sampled at random respecting the query length distribution, and (3) the sampled queries were manually checked for anonymity-breaking words, languages other than English, containing child porn intents, etc.

In the first step (filtering), queries were discarded according to the following exclusion criteria:

- Queries comprising remnants of URLs (e.g., .com or http) or URL character encodings (to exclude strictly "navigational" queries caused by confusing the search box with the address bar).
- Queries from searchers having more than 10,000 queries in the logged 3-month period (to exclude some query bots).
- Queries from searchers whose average time between consecutive queries is less than one second (to further exclude query bots).
- Queries from searchers whose median number of letters per query is more than 100 (probably also bots).
- Queries that contain non-alphanumeric characters except for dashes and apostrophes in-between characters.
- Queries from searchers that duplicate preceding queries of themselves (to exclude result page interaction from the query frequency calculation).
- Queries with less than three or more than ten words.

We had a corpus size of more than 50,000 queries in mind and anticipated that the necessary manual cleansing (third step) could reduce the size of any query sample—thus, initially 55,555 queries were drawn to account for up to a potential 10% reduction.

To accomplish the query length distribution sampling (second step), the filtered log was divided into query length classes, where the i-th class contains all queries with i words (i.e., $i-1$ whitespaces), keeping duplicate queries from different searchers. Then, the query length distribution was computed and the amount of each length class to be expected in a 55,555 query sample was determined. Based on these expectations, for each length class, queries were sampled without replacement until the expected amount of distinct queries was reached. Hence, our sample represents the query length distribution of the filtered log. And since each length class in the filtered log contained duplicate entries of queries according to their frequency, our sample also represents the query frequency distribution in the filtered query log. One might argue that our sampling may miss many rare spelling errors but on the other hand, one might also argue that we just favor the more frequent errors whose correction could help many users. Either way, our later analyses of the amount of errors will show that they are similar to the previous corpora.

In the final manual cleansing (third step), we had one annotator go through all the 55,555 queries, labeling those that are non-English (the target language of our corpus), containing child porn intents (to be excluded from our corpus), or containing any potentially anonymity-breaking information (e.g., social security numbers, etc.). After the cleansing, 54,772 queries remained such that our goal of sampling more than 50,000 queries was easily reached. These 54,772 queries then went into manual spelling variant annotation.

Parenthesis: A Word on Anonymity. The AOL query log has been released without proper anonymization (only replacing the searchers' IP addresses with numerical IDs) [18]. This raised a lot of concerns among researchers as well as in the media, since some AOL users could be personally identified by analyzing their queries. We address this problem in our corpus by removing searcher IDs entirely and only publishing query strings without submission times or surrounding interactions. This way, queries from our sample could only be reliably mapped back to some original searcher if they contain user-identifying information or if they were submitted by only one user in the AOL log. With our cleansing step described above, we try to avoid the former potential anonymity breach, while, against the latter, someone would have to actually trace a query back in the AOL log and then be able to de-anonymize the respective user(s).

3.2 Query Spelling Correction

As for the spelling correction, 2 independent annotators went through all the 54,772 queries; allowed to use any tool they wanted to support their work (e.g., Hunspell, aspell, search engines, dictionaries, Wikipedia). For each query, potential alternative spellings (also possibly more than one) had to be annotated. After two months of working on the spelling corrections (not necessarily full-time), both annotators discussed the cases where they disagreed. This typically resulted in different reasonable spelling variants being fed into the final corpus. After this step, three annotators each independently checked one third of the queries that contained alternative spellings from the first iteration and could further add or remove variants if need be—also using tools of their choice. Finally, for 9,170 queries (16.74%) some variant different to the original spelling was annotated in the process.

Of course, this annotation process is not perfect and some spelling errors might have been missed or even been introduced. Hence, correcting the queries will remain an ongoing task with potential future corpus updates. For instance, after the corpus release, the community working with the corpus may submit further spelling variants that will then be included and also made publicly available.

3.3 Corpora Analysis and Comparison

Table 1 contains the characteristics of the two previously available spelling correction corpora and our new corpus. The typical spelling error types reported in the table are deletion (entertaner

Table 1: Corpora characteristics (MS = Microsoft Speller Challenge, JDB = qSpell corpus, Ours = our new corpus).

	MS	JDB	Ours
Corpus size			
Queries	5,892	6,000	54,772
w/ alternative spellings	1,121 (19.04%)	983 (16.38%)	9,170 (16.74%)
Error type frequency (percentage of all queries with alternative spellings)			
Deletion	308 (27.45%)	226 (22.99%)	3,054 (33.30%)
Insertion	163 (14.53%)	235 (23.91%)	1,688 (18.41%)
Space	625 (55.70%)	497 (50.56%)	2,821 (30.76%)
Special character	0 (0.00%)	0 (0.00%)	3,229 (35.21%)
Substitution	135 (12.03%)	118 (12.00%)	1,751 (19.09%)
Transposition	31 (2.76%)	27 (2.75%)	386 (4.21%)

→ entertainer), insertion (baseballl → baseball), missing or added spaces (e.g., sponge bob → spongebob), missing or wrong special characters (e.g., noahs ark → noah's ark), substitution (canfederate → confederate), and transposition (chevorlet → chevrolet). Note that the numbers per error type do not necessarily add up to the number of queries with alternative spellings since some queries might contain more than one error type (the percentages indicate the ratio of queries with spelling variants that have a particular error in some variant).

As can be seen, the overall ratio of queries with alternative spellings is similar in all corpora. However, per error type, it is obvious that our annotators were the only ones who also annotated special characters as possible spelling variants; although we did not instruct them to do so. Since spelling correction often takes place after query normalization (i.e., after removal of special characters), we added respective variants without special characters in a post-processing. This ensures compatibility of our corpus with any ordering of the query understanding pipeline (i.e., normalization before or after spelling correction). On average, the number of spelling variants per query is around 1.05–2.36 for different query classes in the corpora (i.e., most corrected queries have just one or two spelling variants) while the average Levenshtein distance from the original query to its closest variant is around 0.3–1.5 for queries with alternative spellings (especially in the Microsoft Speller Challenge corpus, the original spelling often is among the alternative spellings).

Altogether, our new corpus has similar error characteristics as the smaller previous corpora with the potential additional bonus of also including corrections with special characters.

4 EVALUATION

To compare the different corpora not just based on the annotated errors but also with respect to how hard it is for state-of-the-art query spelling correction to handle the ones contained, we conduct a pilot experiment on all three corpora. As a baseline, we choose the approach that does nothing, which turns out to be a rather strong competitor due to the large number of queries not containing any error (>80%) or having the original spelling as one variant. This baseline is contrasted with a re-implementation of Gord Lueck's approach that won the Microsoft Speller Challenge; basing the re-implementation solely on Lueck's publication for the challenge to also conduct a small-scale reproducibility study. To also include current search systems, we submitted all the queries from the three corpora to the Bing Spell Check API[4] and also to the Google search engine checking whether corrections are suggested ("Showing results for," "Containing results for," "Did you mean," etc.). Table 2 contains the results of the three aforementioned approaches, and the baseline that does nothing.

Table 2: Query spelling correction performance.

	Prec@1	EF_1	EP	ER
Microsoft Corpus				
Google	0.962	0.892	0.961	0.833
Bing	0.948	0.865	0.928	0.810
Lueck	0.650	0.854	0.887	0.823
Baseline	0.947	0.873	0.947	0.810
JDB Corpus				
Google	0.947	0.914	0.941	0.888
Bing	0.929	0.888	0.918	0.860
Lueck	0.619	0.877	0.900	0.855
Baseline	0.906	0.870	0.906	0.836
Our Corpus				
Google	0.912	0.904	0.905	0.903
Bing	0.851	0.833	0.833	0.833
Lueck	0.541	0.836	0.812	0.863
Baseline	0.851	0.842	0.851	0.833

As evaluation measures, we employ the ones from the Microsoft Speller Challenge (EF_1, EP, ER), and additionally Precision@1 to check how good an approach's candidate with the highest confidence actually is. For the Microsoft Speller Challenge, the spell correction approaches could submit a set C of potential correction candidates for each query q from the query set Q of the corpus that also contains the gold standard corrections $G(q)$ for every query. A correction candidate c from the derived correction set $C(q)$ of query q has to come with a likelihood or confidence $P(c|q)$ that c actually is a valid spelling for q; the $P(c|q)$ values have to sum up to 1 for each query. The "expected precision" EP and "expected recall" ER of a spelling correction approach then are defined as follows:

$$\text{EP} = \frac{1}{|Q|} \sum_{q \in Q} \sum_{c \in C(q) \cap G(q)} P(c|q), \qquad \text{and}$$

$$\text{ER} = \frac{1}{|Q|} \sum_{q \in Q} \frac{|C(q) \cap G(q)|}{|G(q)|}.$$

The combined EF_1 score is defined as $0.5 \cdot (1/\text{EP} + 1/\text{ER})$. Note that with the above definitions, a successful strategy can be to include many potential corrections with low confidence scores in order to increase ER without harming EP too much. To somewhat counter this possibility, we also report Precision@1, which is simply the average over all queries of the precision at the first rank given the confidence scores (i.e., "simulating" the real-world scenario that a search system has to actually decide whether to correct a query

[4] https://azure.microsoft.com/en-us/services/cognitive-services/spell-check/

or not, whereas giving tens of possible candidates is not supporting a user). In case of ties at the first rank (i.e., same confidence scores), we checked whether one of these top-ranked corrections is in the gold standard (i.e., in doubt, favor the approach). To compute the confidence scores for Bing we simply equally weighted the suggestions—hardly ever more than two—and for Google, we resorted to a simple heuristic: (a) if just results for an alternative spelling are shown, this variant gets a confidence of 1 ("Showing results for"), (b) if results for an alternative and the original spelling are shown, the alternative gets 0.55 and the original 0.45 ("Containing results for"), (c) if only results for the original spelling are shown but an alternative is suggested, the original gets a confidence of 0.75 and the alternative of 0.25 ("Did you mean").

As can be seen from Table 2, only Google reliably outperforms the do-nothing baseline. It is also particularly striking how low the Prec@1 scores of Lueck's approach are. In fact, we also could not really reproduce the performance of $EF_1 > 0.9$ that was reported for Lueck's approach on the Microsoft Corpus [6]. We tried to follow Lueck's description of his approach [14] as closely as possible but some parts of the scoring scheme might not have been described and also some "optimizations" targeting the Speller Challenge's evaluation measures might not have been reported. Still, even the Bing system struggles to improve upon the baseline.

To further analyze the problems of Bing and Lueck's approach, we take a closer look on the error classes and on queries without spelling problems. While an in-depth analysis is beyond the scope of this paper, we summarize some particularly interesting insights with a focus on Prec@1 since the top rank would probably be the basis for retrieving search results. On queries with no errors, only Google and Bing achieve Prec@1 close to 0.99 while Lueck's approach for about every second or third such query suggests a top-variant that is not in the ground truth. The only approach achieving Prec@1 above 0.5 for most classes (error types and without error) is the Google system (except space and special character). Bing and Lueck's approach for many error classes like insertion or deletion only perform around 0.1–0.2 for Prec@1 (only at most one to two out of ten rank 1 suggestions actually are in the gold standard). On such cases, Lueck's approach rather achieves better EF_1 scores than Bing on our corpus. This is probably due to the many reported possible candidates (the Bing Spell Check API usually reported one or two candidates). However, on the Microsoft and the JDB corpora, the EF_1 scores of Bing on error classes are about twice as large as the ones from Lueck; both still being below 0.4, though.

Our brief experimental study shows that Google actually seems to have the most useful spelling corrections (high Prec@1 for almost all classes and also highest EF_1 scores) while Bing is somewhat behind and the many suggestions produced by Lueck's approach do not help in the practically important Prec@1 category.

5 CONCLUSION AND OUTLOOK

Our new freely available corpus of query spelling corrections is about an order of magnitude larger than the two previously available corpora. As future work, we plan to include entity linking and maybe related queries to provide a large-scale corpus that supports research on several components of the query understanding pipeline. In fact, as a first step, we will link the spelling corrections to our previously collected query segmentations [8].

The portion of queries with alternative spellings in our new corpus is similar to the previous corpora (16.74%). However, our corpus is the only one containing spelling variants with special characters—providing a testbed for query spelling before or after normalization (i.e., before or after treating special characters).

In a first study, we have compared the spelling corrections from the commercial search engines Google and Bing to a re-implementation of the best performing approach from the Microsoft Speller Challenge 2011. Our results on all corpora indicate that only Google is able to substantially improve upon a simple do-nothing baseline, while the other two approaches often perform worse. But even the Google system is not able to always correct a typo and for some of the queries without errors suggests different spellings. Hence, query spelling correction is still not a "solved" problem.

REFERENCES

[1] Farooq Ahmad and Grzegorz Kondrak. 2005. Learning a spelling error model from search query logs. In *Proceedings of HLT/EMNLP 2005*.

[2] Fei Cai and Maarten de Rijke. 2016. A survey of query auto completion in information retrieval. *Foundations and Trends in Information Retrieval* 10 (2016), 273–363.

[3] Ishan Chattopadhyaya, Kannappan Sirchabesan, and Krishanu Seal. 2013. A fast generative spell corrector based on edit distance. In *Proceedings of ECIR 2013*, 404–410.

[4] Silviu Cucerzan and Eric Brill. 2004. Spelling correction as an iterative process that exploits the collective knowledge of web users. In *Proceedings of EMNLP 2004*, 293–300.

[5] Huizhong Duan and Bo-June Paul Hsu. 2011. Online spelling correction for query completion. In *Proceedings of WWW 2011*, 117–126.

[6] Huizhong Duan, Yanen Li, ChengXiang Zhai, and Dan Roth. 2012. A discriminative model for query spelling correction with latent structural SVM. In *Proceedings of EMNLP-CoNLL 2012*, 1511–1521.

[7] Yasser Ganjisaffar, Andrea Zilio, Sara Javanmardi, Inci Cetindil, Manik Sikka, Sandeep Paul Katumalla, Narges Khatib-Astaneh, Chen Li, and Cristina Lopes. 2011. qSpell: Spelling correction of web search queries using ranking models and iterative correction. In *Spelling Alteration for Web Search Workshop 2011*.

[8] Matthias Hagen, Martin Potthast, Benno Stein, and Christof Bräutigam. 2011. Query segmentation revisited. In *Proceedings of WWW 2011*, 97–106.

[9] Sasa Hasan, Carmen Heger, and Saab Mansour. 2015. Spelling correction of user search queries through statistical machine translation. In *Proceedings of EMNLP 2015*, 451–460.

[10] Liangda Li, Hongbo Deng, Jianhui Chen, and Yi Chang. 2017. Learning parametric models for context-aware query auto-completion via Hawkes processes. In *Proceedings of WSDM 2017*, 131–139.

[11] Mu Li, Muhua Zhu, Yang Zhang, and Ming Zhou. 2006. Exploring distributional similarity based models for query spelling correction. In *Proceedings of ACL 2006*.

[12] Yanen Li, Huizhong Duan, and ChengXiang Zhai. 2012. CloudSpeller: Query spelling correction by using a unified hidden Markov model with web-scale resources. In *Proceedings of WWW 2012*, 561–562.

[13] Yanen Li, Huizhong Duan, and ChengXiang Zhai. 2012. A generalized hidden Markov model with discriminative training for query spelling correction. In *Proceedings of SIGIR 2012*, 611–620.

[14] Gord Lueck. 2011. A data-driven approach for correcting search queries. In *Spelling Alteration for Web Search Workshop 2011*.

[15] Peter Nalyvyko. 2011. A REST-based online English spelling checker "Pythia". In *Spelling Alteration for Web Search Workshop 2011*.

[16] Peter Norvig. 2007. How to write a spelling corrector. http://norvig.com/spell-correct.html. (2007).

[17] Yoh Okuno. 2011. Spelling generation based on edit distance. In *Spelling Alteration for Web Search Workshop 2011*.

[18] Greg Pass, Abdur Chowdhury, and Cayley Torgeson. 2006. A picture of search. In *Proceedings of Infoscale 2006*, 1.

[19] Jason J. Soo. 2013. A non-learning approach to spelling correction in web queries. In *Proceedings of WWW 2013*, 101–102.

[20] Dan Stefanescu, Radu Ion, and Tiberiu Boros. 2011. TiradeAI: An ensemble of spellcheckers. In *Spelling Alteration for Web Search Workshop 2011*.

[21] Xu Sun, Anshumali Shrivastava, and Ping Li. 2012. Fast multi-task learning for query spelling correction. In *Proceedings of CIKM 2012*, 285–294.

[22] Kuansan Wang and Jan Pedersen. 2011. Review of MSR-Bing web scale speller challenge. In *Proceedings of SIGIR 2011*, 1339–1340.