# Multi-teacher Distillation for Multilingual Spelling Correction

**Jingfen Zhang**
Amazon.com Inc
jingfenz@amazon.com

**Xuan Guo**
Amazon.com Inc
xuangu@amazon.com

**Sravan Bodapati**
Amazon.com Inc
sravanb@amazon.com

**Christopher Potts**
Stanford University
cgpotts@stanford.edu

## Abstract

Accurate spelling correction is a critical step in modern search interfaces, especially in an era of mobile devices and speech-to-text interfaces. For services that are deployed around the world, this poses a significant challenge for multilingual NLP: spelling errors need to be caught and corrected in all languages, and even in queries that use multiple languages. In this paper, we tackle this challenge using multi-teacher distillation. On our approach, a monolingual teacher model is trained for each language/locale, and these individual models are distilled into a single multilingual student model intended to serve all languages/locales. In experiments using open-source data as well as user data from a worldwide search service, we show that this leads to highly effective spelling correction models that can meet the tight latency requirements of deployed services.

## 1 Introduction

Spelling correction is vital to the modern search experience. Users expect it, mobile devices and speech-to-text interfaces make it more crucial than ever, and uncaught spelling errors can lead to urgent problems of security and trust if problematic search results are shown to users. For services with a global reach, this poses a substantial challenge for multilingual NLP: spelling errors must be caught and corrected in any language, and even in queries using multiple languages.

The promise of multilingual language models is that we may be able to meet these challenges with a single spelling correction model serving all languages/locales. In the present paper, we develop and motivate such a multilingual approach relying crucially on multi-teacher distillation. On our approach, an individual teacher model is trained for each language/locale, and these individual models are distilled into a single multilingual student model intended to serve all languages/locales.

Our distillation objective is a purely behavioral one: the multilingual student is trained to mimic the input–output behavior of the individual teachers. This brings a number of key advantages in our setting. First, we can customize the individual teacher models to specific languages/locales, which proves especially useful in the area of tokenization. Second, when we want to add a new language/locale $L$, we train just two models: the new teacher for $L$ and the new multilingual model distilled from the input–output pairs generated by all the teacher models. Third, the individual teacher models are themselves assets that can be distilled into student models; where these are superior (common for data-rich languages), they can be used.

We motivate our approach with a wide range of experiments using open-source data as well as proprietary user data from a worldwide search service. Overall, we find that our multi-teacher distillation approach leads to superior models compared to both individual monolingual student models and multilingual student models distilled from a single multilingual teacher. In addition, we show that we can efficiently add new languages and easily meet the tight latency requirements imposed by industrial search engines. Overall, we suggest that this is a promising modeling approach not only for spelling correction but also for the other services needing to serve numerous languages and locales.

## 2 Related Work

Spelling correction is a widely studied problem (Hládek et al., 2020). Earlier work relied on lexical rules (Meddeb-Hamrouni, 1994; Reynaert, 2004) or language models plus linguistic features (Alkhafaji et al., 2013; Sharma et al., 2023). In more recent work, spelling correction is cast as an encoder–decoder problem (Hasan et al., 2015; Zhou et al., 2017; Kuznetsov and Urdiales, 2021), theoretically making it easier to scale to multilingual settings. However, methods that are efficient

and scalable across numerous languages have been much less explored.

Spelling correction is highly sensitive to different tokenization schemes, since it involves manipulating characters and other subword units. Subword tokenization schemes provide the right balance between operating on subword units and being efficient for training and inference. Popular methods for subword tokenization include Byte Pair Encoding (BPE; (Bostrom and Durrett, 2020)), Byte-Level BPE (BBPE; (Wang et al., 2020)), SentencePiece (Kudo and Richardson, 2018), and Unigram Language Model (Kudo, 2018). BPE splits words into subword units based on their statistical properties and is extensively employed by various Transformer models such as GPT (Radford et al., 2018), RoBERTa (Liu et al., 2019), and BART (Lewis et al., 2020a). BBPE operates at the byte-level, efficiently enabling the encoding and decoding of texts across different languages with non-overlapping character sets. In this paper, we explore BPE and BBPE tokenization schemes in our experiments.

Increasing model size and amount of compute used for training will generally improve the performance of neural language models (Kaplan et al., 2020), but the costs might be prohibitive. In model distillation (Hinton et al., 2015), a large teacher model is used to guide the training of a smaller student model. This is a viable solution for developing deployable models with strict production constraints. These approaches have proven highly successful for seq2seq problems in general (Kim and Rush, 2016; Liang et al., 2022). Distillation approaches vary in the degree to which they presuppose access to the teacher model during student model training (Gou et al., 2021). At one extreme, the teacher and student models are trained together (i.e., co-distillation; Chung et al. 2020). At the other extreme, the teacher is simply used to generate output labels for the training data, based on which the student is trained (e.g., Sequence-level Knowledge Distillation (Seq-KD); Kim and Rush 2016). In our multi-teacher distillation, we aim to decouple the teacher and student training regimes in order to train the best model for each language, and so we use Seq-KD. There are existing studies about multiple teacher distillation. For example, You et al. combined multiple teacher networks by averaging the softened output targets and selecting layers in student and teacher networks (You et al., 2017). Yuan et al. selected soft labels from a collection of teacher models, based on the reward signal from performance of distilled student model (Yuan et al., 2021). Both studies use multiple teachers to generate variant candidates and distill knowledge to build a robust and accurate student. In this paper, we apply multi-teachers for multilingual problem, where each teacher specializes in one language and they work together to guide the learning of a multilingual student.

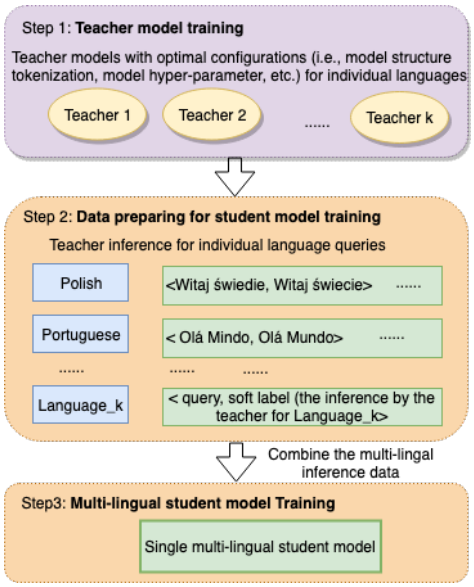


Figure 1: Multi-teacher distillation workflow.

## 3 Methods

We aim to create a high-performing model that can serve multiple languages while satisfying latency restrictions. We propose a multi-teacher distillation method. The main idea is to train teacher models with optimal configurations for individual languages and then build a single student model based on the multi-teacher inference. Figure 1 provides a high-level overview of our multi-teacher distillation approach, which we describe in this section.

### 3.1 Model Architecture

We first formulate the spelling correction task as a text-to-text problem: a query is the input to the model, and the model outputs a correctly spelled query. If the model detects no spelling errors in the input, it outputs a query identical to the input.

We use the Bidirectional Auto Regressive Transformer (BART; Lewis et al. 2020b) architecture for model building. BART is pretrained on a denoising objective mapping corrupted sequences into their

uncorrupted forms, which has many similarities to spelling correction itself. However, our approach is not specific to BART. Indeed, our very lightweight distillation objective even allows different architectures to be used for different languages.

All the spelling correction models reported in this paper are trained from scratch on spelling correction datasets rather than starting from pretrained parameters. This might seem surprising given widespread evidence that pretraining improves models. For example, multilingual BART (mBART) (Liu et al., 2020) is reported as a good pretrained model for many multilingual tasks such as machine translation (Maurya et al., 2021), text generation (Chen et al., 2021), text summarization (Wang et al., 2021), and entity linking (De Cao et al., 2022). However, spelling correction is arguably a different area from the other tasks. First, pretraining objectives tend to serve semantic goals, whereas many aspects of spelling correction are purely form-based (Huang et al., 2023). Second, spelling correction training datasets can be truly massive, since gold behavior data can be expanded with synthetic examples. As a result of these factors, the contributions of pretraining are minimal in practice. For our purposes, this has the advantage of leading to more controlled experimental comparisons, as we do not have to worry about variation in pretraining as a factor in model performance.

### 3.2 Teacher Training

For teacher training, we train different customized individual teacher models for each language to achieve high performance. For example, we adapt BPE or BBPE tokenization methods according to each language's characteristics, and build both monolingual and multilingual models with different hyper-parameters based on language difficulty and training data availability. The optimal choice varies between languages, and our approach can accommodate this in the teacher creation phase.

### 3.3 Distillation Objective

As discussed in Section 2, we use the Seq-KD method of Kim and Rush (2016), in which the teacher is simply used to generate "soft" labels for student training. This has led to exceptional spelling correction models in practice, in the context we describe in Section 4. In addition, it is extremely efficient in terms of overall system development, and it allows the teacher and student models to have different sizes, tokenization schemes, and other architectural features.

In our experiments, we explore a range of options: (1) a multilingual teacher distilled into a multilingual student model; (2) multi-teacher distillation using each of the monolingual teachers; and (3) multi-teacher distillation from the best teacher for each language, selected from the set of monolingual and multilingual teachers available. It turns out that the option (3) provides the best results.

A guiding hypothesis for our method is that our distillation process can lead to individual models that are not only capable of serving all languages/locales, but also superior to monolingual models due to knowledge sharing across languages. We expect to see the largest gains in low-resource languages, and this is indeed what we find experimentally.

### 3.4 Evaluation Metric

Our evaluations are based in exact match (after punctuation removal) between gold and predicted outputs, and we focus on cases involving corrections to avoid inflating our scores with inputs that contain no spelling errors. Thus, precision is the percentage of model-predicted corrections that are correct according to the gold data, and recall is the percentage of cases requiring corrections that the model predicts correctly. We report the F1 score of these two values. Appendix A provides additional details on score calculation.

## 4 Experiments on User Data

In this section, we report on experiments with user data from a large, global search service, and the user data are their search queries. In Section 5, we report on experiments with open source data that are natural language sentences. With the open-source data, we can be completely open about the findings, with some costs in terms of realism. With the user data, we are required to conceal some details, but the findings themselves still provide a clear picture of how our approach fares in the real world.

### 4.1 Training Data

Our user data are derived from a global search service. For a proof of concept, we focus on six languages: Portuguese, Dutch, Turkish, Swedish, Polish, and Arabic. These cover eight locales: Brazil (BR), Netherlands (NL), Turkey (TR), Swe-

| Language | Locale | Teachers: Multi-BPE [a] teacher | Teachers: Multi-BBPE teacher | Teachers: Monolingual teacher | Teachers: Best teacher | Single-teacher distillation: Multi-BPE student | Single-teacher distillation: Multi-BBPE student | Multi-teacher distillation: Matched mono student | Multi-teacher distillation: Best mono student |
|---|---|---|---|---|---|---|---|---|---|
| Portuguese | BR | – | −1.2% | 3.3% | 3.3% | −3.7% | −2.8% | −2.9% | −1.1% |
| Dutch | NL | – | 5.5% | −5.2% | 5.5% | −0.6% | −0.3% | 0.9% | 3.4% |
| Turkish | TR | – | −1.1% | −2.0% | 0 | −0.4% | −6.6% | −3.2% | 0.8% |
| Swedish | SE | – | 1.1% | −6.0% | 1.1% | −4.7% | −1.2% | 3.3% | 2.0% |
| Polish | PL | – | −4.0% | −0.5% | 0 | −5.3% | 1.6% | 2.5% | 7.2% |
| | AE | – | 6.3% | 12.5% | 12.5% | −0.4% | −0.6% | 14.3% | 20.4% |
| Arabic | SA | – | 8.7% | 16.6% | 16.6% | −0.1% | 6.5% | 25.1% | 28.5% |
| | EG | – | 4.9% | 16.5% | 16.5% | 1.0% | 1.3% | 23.1% | 32.5% |
| Avg across locales | | – | 1.7% | 2.2% | 5.2% | −2.2% | −1.0% | 4.7% | 8.0% |

Table 1: F1 scores on user data. Due to external constraints, we report only percentage-wise changes relative to the Multi-BPE model, whose absolute performance we cannot disclose. The multi-teacher students (far right two columns) yield the best results. Here, 'Matched mono' is the multilingual model distilled from the column of models represented under 'Monolingual teachers', whereas 'Best mono' is the the multilingual model distilled from the column of models represented under 'Best teachers'. Overall, these results indicate that multi-teacher distillation is an effective strategy for industrial spelling correction, and that the flexibility afforded by our lightweight distillation strategy pays off.

[a] The baseline model.

den (SE), Poland (PL), United Arab Emirates (AE), Saudi Arabia (SA), and Egypt (EG). We collected two years of historical behavior data (2021 to 2023), comprising <input query, label query> pairs. In this context, the input query refers to the user's initial search query, while the label query represents the prediction made by our production speller model as validated by user data (successful completion of a search as measured by clicks and other behavior). We have millions to hundreds of millions of examples, with imbalanced size across locales. For example, the data for PL is less than 1/20 the size of BR.

### 4.2 Evaluation Data

We collected human annotations of search queries to serve as ground truth in model evaluations. For each locale, there are 10K human-annotated queries that reflect the spelling correction distribution in production. These examples are collected from a different time window than the training data collection, and they are carefully sampled to balance cases where misspellings could have been corrected and where good spellings should not have been over-corrected.

### 4.3 Monolingual Teachers

The first step of our method is to train individual teacher models, including both monolingual and multilingual models with optimal configurations to reach high performance in each language. A major advantage of our approach is that we can train a diverse set of models, using choices that are tightly aligned with what we know about individual languages. We heuristically explored different configurations for different languages. This led us to use a full-size BART-large model with a 128K BPE vocabulary (480M parameters) for BR, and a 6-layer BART model with 32K BBPE vocabulary (211M parameters) for the other locales.

### 4.4 Multilingual Teachers

We trained two multilingual teacher models with the full-size BART-large architecture. The **Multi-BPE** model uses BPE tokenization and has a 128K vocabulary (490M parameters). This model serves as a baseline for all our comparative reporting. The **Multi-BBPE** model uses BBPE tokenization and has a 32K vocabulary (471M parameters).

### 4.5 Multi-Teacher Distillation

We distill teacher models into student models according to the methods described in Section 3.3. All student models are BART-base models with 2 layers, trained with 25 training epochs. Each epoch contains 200 millions of training data. These models are small compared to the teacher model due to our latency requirements (Section 4.8).

### 4.6 Results

Our results are summarized in Table 1. Due to external constraints, we can show only percentage-wise gains and losses relative to the Multi-BPE teacher model, rather than reporting raw F1 scores. Nonetheless, the findings are very clear: our multi-teacher distillation approach is superior, leading

to solid gains in nearly every locale and a very large average improvement across locales. The best students are those distilled from the best teacher for each language (rightmost column).

Some variation is observed across different languages and locales. For example, a significant difference of 8.7% is observed in SA and a 5.5% difference is observed in NL. When comparing the monolingual models with the multilingual models, a similar pattern is observed, with comparable overall performance but even larger variation across languages and locales. On our approach, we can embrace this variation and choose the best teacher for each language to obtain a better multi-lingual student model.

Multi-teacher distillation out-performs the corresponding monolingual teacher in all locales except BR. BR is the largest of these locales, and it is common for large locales to support very strong monolingual models; the strengths of multi-teacher distillation are usually most apparent in low-resource locales. During multilingual student training, we observed differences across languages. While the training for most languages achieved convergence, the training on Portuguese data did not converge optimally. Although we could achieve better performance on Portuguese by doing more training, over-fitting could result in a sacrifice of performance on the other languages. In the future, we plan to address this by treating different languages as different tasks and developing a multi-task approach that dynamically allocates computing effort to different languages (Ruder, 2017; Duong et al., 2015; Baxter, 1997).

### 4.7 Adding New Languages

The results in Table 1 shows that multilingual student training has the capacity to transfer knowledge among languages. In addition, the approach makes it easy to include new languages or data in the future with minimal effort: we simply add the new monolingual teacher model inference data into the distillation process and expand the multilingual student model without having to retrain the entire set of teacher models for all languages from scratch.

To illustrate, we trained a multilingual student model using monolingual teacher inference data from three languages: Portuguese, Dutch, and Polish. We obtained an improvement of 4.7% in the average F1 score of the student model compared to the average F1 score of the teachers. We then added two more languages (Turkish and Swedish) and obtained similar F1 scores for Portuguese, Dutch, and Polish while achieving better performance for Turkish and Swedish than their respective monolingual teachers. Table 2 summarizes these experiments.

| Locale | Monolingual teacher | Distill on 3 languages | Distill on 5 languages |
|---|---|---|---|
| BR | – | −2.0% | −2.2% |
| NL | – | 9.0% | 9.2% |
| PL | – | 10.0% | 9.0% |
| TR | – | | 4.5% |
| SE | – | | 20.6% |
| Avg (3) | – | 4.7% | 4.4% |
| Avg (5) | – | | 7.0% |

Table 2: Model performance (F1) after adding new languages to the multi-teacher distillation process.

### 4.8 Latency

Industrial search technologies operate under very tight latency requirements. We have demonstrated that our multi-teacher distilled student model outperforms the larger teacher models (Table 1), but we have not so far quantified the latency gains that this brings.

In this section, we evaluate the impact of multi-teacher distillation on online deployment by conducting a real traffic load test to measure throughput per second (TPS) and P99 latency. A higher TPS enables a reduction in the number of GPU instances needed to handle the same volume of service requests, thereby lowering the overall Initial Margin Requirement (IMR) costs of the inference fleet. In addition, improvements in the P99 latency will allow for more spelling corrections that would otherwise result in "no corrections" due to timeouts. This ensures that the online F1 performance is consistent with the offline F1, leading to a better user experience.

Table 3 lists the comparison of TP99 latency and TPS by the multi-teacher distilled student model on six locales against the multilingual teacher model reported in Table 1. For these experiments, we first convert the model to an ONNX (Open Neural Network eXchange) model graph (ONNX 2023) and then optimize the serialized ONNX graph using TensorRTframework (Vanholder, 2016). Here, all latency numbers are based on the TensorRT serialized model on an AWS g5.2xlarge GPU instance. We observe that the TPS of the student model is

| | P99 | TPS | | P99 | TPS |
|---|---|---|---|---|---|
| BR | 37.9% | +2.1x | TR | 37.1% | +2.1x |
| PL | 43.7% | +2.1x | SE | 33.9% | +2.3x |
| NL | 31.6% | +2.3x | AE | 50.1% | +2.6x |

Table 3: Improvement in TP99 latency and throughput for the student model vs. the baseline teacher model shown in Table 1.

double that of the teacher model, and so we can save more than half of IMR costs by deploying the student models.

## 5 Experiments on Open-Source Data

To supplement our experiments on user data, we also conducted experiments with open-source data for which we can supply absolute performance numbers.

### 5.1 Data

A few spelling datasets have been proposed (Hagiwara and Mita, 2020; Rothe et al., 2021), but most of these focus primarily on English. In this paper we use the large multilingual dataset from the Workshop on Statistical Machine Translation (WMT) website.[1] We downloaded the europarl, news-commentary, and news- 2007 to news-2011 corpora for five languages, English (EN), Germany (DE), Czech(CS), French(FR) and Spanish(ES). We then injected synthetic noise into these sequence to get <noise inserted sequence, original sequence> pairs as our training data. The operations used in noise injection include inserting, deleting, and replacing random characters at random locations. For each original sentence, we generated 8 noised sentences for training set, and one noised sentence for evaluation set. For evaluation data, we filtered out trivial cases and sequence less than 6 words, and then randomly selected 10,000 as the evaluation data for each language. Table 4 provides an overview of these resources.

### 5.2 Models

We conduct both monolingual and multilingual teacher training. For all teacher models, we use a BART-large model structure with 6-layer Transformers and a 32K BBPE vocabulary. As before, we compare three student models: (1) a model distilled from the single multilingual teacher, (2) a model distilled from the monolingual teachers, and (3) a model distilled from the best teacher for each language, which can be either the monolingual model for that language or the multilingual teacher.

[1] https://www.statmt.org/wmt19/translation-task.html

| Language | Train | Eval | Overlap |
|---|---|---|---|
| EN | 181,597,816 | 10,000 | 393 |
| DE | 133,116,472 | 10,000 | 418 |
| CS | 71,469,552 | 10,000 | 475 |
| FR | 47,164,952 | 10,000 | 480 |
| ES | 9,215,136 | 10,000 | 515 |

Table 4: Open source data: training and evaluation data size. The overlap between the evaluation data and the training data ranges from 3.93% to 5.15% (denominator is evaluation size).

| | Teachers | | Students | | |
|---|---|---|---|---|---|
| | Multi | Mono | S-T | M-T | B-T |
| EN | 76.0 | 77.4 | 71.6 | 72.9 | 73.0 |
| DE | 90.3 | 92.0 | 85.2 | 87.4 | 87.5 |
| CS | 85.0 | 85.3 | 77.7 | 80.2 | 80.7 |
| FR | 44.8 | 44.4 | 42.2 | 42.8 | 43.0 |
| ES | 85.1 | 81.9 | 81.4 | 81.0 | 82.6 |
| Avg. | 76.3 | 76.2 | 71.6 | 72.9 | 73.4 |

Table 5: Open-source data experiment results (F1 scores). Here, 'Multi' and 'Mono' are multilingual and monolingual teacher models, whereas 'S-T', 'M-T' and 'B-T' are distilled from the single multilingual teacher, multi-monolingual teachers and the best teachers, respectively. Our multi-teacher distillation approach is superior for all languages, with the best results emerging where the best teacher for each language is used.

### 5.3 Results

Table 5 summarizes our findings. In terms of performance, the student model distilled from the multi-monolingual teachers outperforms the student model distilled from the single multilingual model, achieving an F1 score of 72.9 versus 71.6. The student model distilled from the best teachers surpasses both, achieving the highest F1 score of 73.4. Detailed F1 scores for different models are listed in Table 5. Note that the training data size and training epochs for different methods are equivalent, to make sure that the performance differences do not trace to these factors.

## 6 Conclusion

We developed and motivated a multi-teacher distillation approach for multilingual spelling correction. On our approach, teacher models for individual languages are used to distill a single multilingual student model. By focusing on improving the performance of teacher models for specific languages, we can enhance the overall performance of the student model. Additionally, our approach allows for the inclusion of new monolingual teacher model inference data into the distillation process, enabling the expansion of the multilingual student model without the need to retrain the entire set of teacher models for all languages. We believe that this modeling approach holds promise not only for spelling correction services but also for other services needing to serve numerous languages and locales. We will release our code for data preparation, model training, inference, and evaluation upon publication.

## Ethics Statement

We hereby acknowledge that all of the co-authors of this work are aware of the provided ACL Code of Ethics and honor the code of conduct.

In this paper, we are focused on situations involving people from diverse linguistic and cultural backgrounds, spread all around the world. This is a very challenging context for any NLP system, and it raises the concern that models might be overfit to specific groups (usually the largest and most influential) at the expense of other groups. We certainly do not claim to have solved this problem, but we do view our proposed approach as an attempt to make cautious progress here. In particular, since we train a monolingual model for every language/locale, we can always fall back to that model if the multilingual one shows problematic transfer that degrades performance. On the other hand, we expect that, on average, the multilingual models will help to make up for data scarcity problems for specific languages, which improves the experiences of those users on our site, and that they will also allow users the freedom to use multilingual queries if they wish. Also, considering the popularity and relevance of our service, we anticipate that over time, our traffic will attract individuals from diverse linguistic and cultural backgrounds, thereby partially mitigating the issue.

## References


Hussein Alkhafaji, Suhail Abdullah, and Hanaa Merza. 2013. A new algorithm to design and implementation of multilingual spellchecker and corrector. *Journal of Al-Rafidain University College For Sciences (Print ISSN: 1681-6870, Online ISSN: 2790-2293)*, 2:32–49.

Jonathan Baxter. 1997. A Bayesian/information theoretic model of learning to learn via multiple task sampling. *Machine learning*, 28:7–39.

Kaj Bostrom and Greg Durrett. 2020. Byte pair encoding is suboptimal for language model pretraining. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 4617–4624, Online. Association for Computational Linguistics.

Yiran Chen, Zhenqiao Song, Xianze Wu, Danqing Wang, Jingjing Xu, Jiaze Chen, Hao Zhou, and Lei Li. 2021. Mtg: A benchmark suite for multilingual text generation. *arXiv preprint arXiv:2108.07140*.

Inseop Chung, SeongUk Park, Jangho Kim, and Nojun Kwak. 2020. Feature-map-level online adversarial knowledge distillation. In *International Conference on Machine Learning (ICML)*, pages 2006–2015. PMLR.

Nicola De Cao, Ledell Wu, Kashyap Popat, Mikel Artetxe, Naman Goyal, Mikhail Plekhanov, Luke Zettlemoyer, Nicola Cancedda, Sebastian Riedel, and Fabio Petroni. 2022. Multilingual autoregressive entity linking. *Transactions of the Association for Computational Linguistics*, 10:274–290.

Long Duong, Trevor Cohn, Steven Bird, and Paul Cook. 2015. Low resource dependency parsing: Cross-lingual parameter sharing in a neural network parser. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 845–850, Beijing, China. Association for Computational Linguistics.

Jianping Gou, Baosheng Yu, Stephen J Maybank, and Dacheng Tao. 2021. Knowledge distillation: A survey. *International Journal of Computer Vision*, 129:1789–1819.

Masato Hagiwara and Masato Mita. 2020. GitHub typo corpus: A large-scale multilingual dataset of misspellings and grammatical errors. In *Proceedings of the Twelfth Language Resources and Evaluation Conference*, pages 6761–6768, Marseille, France. European Language Resources Association.

Saša Hasan, Carmen Heger, and Saab Mansour. 2015. Spelling correction of user search queries through statistical machine translation. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 451–460.

Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. 2015. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*.

Daniel Hládek, Ján Staš, and Matúš Pleva. 2020. Survey of automatic spelling correction. *Electronics*, 9(10):1670.

Jing Huang, Zhengxuan Wu, Kyle Mahowald, and Christopher Potts. 2023. Inducing character-level structure in subword-based language models with type-level interchange intervention training. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 12163–12180, Toronto, Canada. Association for Computational Linguistics.

Jared Kaplan, Sam McCandlish, Tom Henighan, Tom B Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeffrey Wu, and Dario Amodei. 2020. Scaling laws for neural language models. *arXiv preprint arXiv:2001.08361*.

Yoon Kim and Alexander M. Rush. 2016. Sequence-level knowledge distillation. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 1317–1327, Austin, Texas. Association for Computational Linguistics.

Taku Kudo. 2018. Subword regularization: Improving neural network translation models with multiple subword candidates. *arXiv preprint arXiv:1804.10959*.

Taku Kudo and John Richardson. 2018. SentencePiece: A simple and language independent subword tokenizer and detokenizer for neural text processing. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 66–71, Brussels, Belgium. Association for Computational Linguistics.

Alex Kuznetsov and Hector Urdiales. 2021. Spelling correction with denoising transformer. *arXiv preprint arXiv:2105.05977*.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2020a. BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7871–7880, Online. Association for Computational Linguistics.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2020b. BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7871–7880, Online. Association for Computational Linguistics.

Xiaobo Liang, Lijun Wu, Juntao Li, Tao Qin, Min Zhang, and Tie-Yan Liu. 2022. Multi-teacher distillation with single model for neural machine translation. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 30:992–1002.

Yinhan Liu, Jiatao Gu, Naman Goyal, Xian Li, Sergey Edunov, Marjan Ghazvininejad, Mike Lewis, and Luke Zettlemoyer. 2020. Multilingual denoising pre-training for neural machine translation. *Transactions of the Association for Computational Linguistics*, 8:726–742.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. RoBERTa: A robustly optimized BERT pretraining approach. *arXiv preprint arXiv:1907.11692*.

Kaushal Kumar Maurya, Maunendra Sankar Desarkar, Yoshinobu Kano, and Kumari Deepshikha. 2021. Zmbart: An unsupervised cross-lingual transfer framework for language generation. *arXiv preprint arXiv:2106.01597*.

Boubaker Meddeb-Hamrouni. 1994. Logic compression of dictionaries for multilingual spelling checkers. In *COLING 1994 Volume 1: The 15th International Conference on Computational Linguistics*.

ONNX. 2023. ONNX: The open standard for machine learning interoperability [An open ecosystem that empowers AI model developments]. `https://github.com/onnx/onnx`. (Accessed: 1 May 2023).

Alec Radford, Karthik Narasimhan, Tim Salimans, Ilya Sutskever, et al. 2018. Improving language understanding by generative pre-training. OpenAI.

Martin Reynaert. 2004. Multilingual text induced spelling correction. In *Proceedings of the Workshop on Multilingual Linguistic Resources*, pages 110–117, Geneva, Switzerland. COLING.

Sascha Rothe, Jonathan Mallinson, Eric Malmi, Sebastian Krause, and Aliaksei Severyn. 2021. A simple recipe for multilingual grammatical error correction. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 702–707, Online. Association for Computational Linguistics.

Sebastian Ruder. 2017. An overview of multi-task learning in deep neural networks. *arXiv preprint arXiv:1706.05098*.

Sanat Sharma, Josep Valls-Vargas, Tracy Holloway King, Francois Guerin, and Chirag Arora. 2023. Contextual multilingual spellchecker for user queries. *arXiv preprint arXiv:2305.01082*.

Han Vanholder. 2016. Efficient inference with tensorrt. In *GPU Technology Conference*, volume 1, page 2.

Changhan Wang, Kyunghyun Cho, and Jiatao Gu. 2020. Neural machine translation with byte-level subwords. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, pages 9154–9160.

Danqing Wang, Jiaze Chen, Hao Zhou, Xipeng Qiu, and Lei Li. 2021. Contrastive aligned joint learning for multilingual summarization. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 2739–2750.

Shan You, Chang Xu, Chao Xu, and Dacheng Tao. 2017. Learning from multiple teacher networks. In *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 1285–1294.

Fei Yuan, Linjun Shou, Jian Pei, Wutao Lin, Ming Gong, Yan Fu, and Daxin Jiang. 2021. Reinforced multi-teacher selection for knowledge distillation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 14284–14291.

Yingbo Zhou, Utkarsh Porwal, and Roberto Konow. 2017. Spelling correction as a foreign language. *arXiv preprint arXiv:1705.07371*.

## Supplementary Materials

## A Evaluation Metrics

We use precision and recall as the offline spelling correction performance metrics, defined as follows:

- *Exact match*(*): String identity after punctuation removal (e.g., "women's" and "womens" as equal).

$$\text{precision} = \frac{\text{action}_e = \text{action}_s = \text{AUTO} \wedge \text{query}_e \simeq \text{query}_s}{\text{action}_s = \text{AUTO}}$$

$$\text{recall} = \frac{\text{action}_e = \text{action}_s = \text{AUTO} \wedge \text{query}_e \simeq \text{query}_s}{\text{action}_e = \text{AUTO}}$$

- Subscript $s$: the model output.
- Subscript $e$: the gold (human-judged) output.
- action: the suggested action. The possible values are AUTO (auto correction) and NONE (no correction).
- query: the corrected query in the case of auto correction
- query_$s$ $\simeq$ query_$e$: query_$s$ is an exact match with query_$e$.