CLIReval: Evaluating Machine Translation as a Cross-Lingual Information Retrieval Task

Shuo Sun
Johns Hopkins University
ssun32@jhu.edu

Suzanna Sia
Johns Hopkins University
ssia1@jhu.edu

Kevin Duh
Johns Hopkins University
kevinduh@cs.jhu.edu

Abstract

We present **CLIReval**, an easy-to-use toolkit for evaluating machine translation (MT) with the proxy task of cross-lingual information retrieval (CLIR). Contrary to what the project name might suggest, CLIReval does not actually require any annotated CLIR dataset. Instead, it automatically transforms translations and references used in MT evaluations into a synthetic CLIR dataset; it then sets up a standard search engine (Elasticsearch) and computes various information retrieval metrics (e.g., mean average precision) by treating the translations as documents to be retrieved. The idea is to gauge the quality of MT by its impact on the document translation approach to CLIR. As a case study, we run CLIReval on the "metrics shared task" of WMT2019; while this extrinsic metric is not intended to replace popular intrinsic metrics such as BLEU, results suggest CLIReval is competitive in many language pairs in terms of correlation to human judgments of quality. CLIReval is publicly available at https://github.com/ssun32/CLIReval.

1 Introduction

Machine translation (MT) is the task of automatically translating sentences from a source language to a target language. A natural question that arises is how do we determine whether an MT system is translating sentences well? One answer is that we can engage human translators to evaluate the translated sentences manually. Unfortunately, evaluating translations can be relatively time-consuming and worse, the fact that the quality of translation is inherently subjective can lead to variations among different human translators. The desire for fast and consistent evaluation has led to the emergence of a plethora of automatic evaluation metrics such as BLEU (Papineni et al., 2002), TER (Snover et al., 2006), METOR (Banerjee and Lavie, 2005) and BEER (Stanojević and Sima'an, 2014). Out of the aforementioned metrics, BLEU has become the de facto evaluation metric for machine translation. It calculates the weighted average of n-gram precision between a translated sentence and a reference sentence. Nevertheless, BLEU, too, has its problems. For example, Callison-Burch et al. (2006) showed that an improved BLEU score does not represent an actual improvement in translation quality.

There are also some proposals to evaluate the quality of translations with the help of extrinsic proxy tasks. Berka et al. (2011) collected short English documents from various domains and created yes and no questions in Czech. They then translated the English documents into Czech and evaluated the quality of the MT systems based on human performances on the documents and questions in Czech. Scarton and Specia (2016) translated a dataset of German reading comprehension tests into English with various MT systems such as Google Translate and Bing Translate and judged the quality of translations based on human performances on the translated reading comprehension datasets. Unfortunately, these external tasks suffer from the same scalability and consistency issues as manual evaluation.

One downstream task that relies heavily on MT but has not been used as a method to evaluate MT systems is the task of *Cross-Lingual Information Retrieval (CLIR)*. CLIR is a task in which search queries are issued in one language, and the retrieved relevant documents are written in a different language. Two commonly used methods in CLIR are *query translation*, where queries are translated into the same language as the documents and *document translation* where documents are translated into the same language as the queries (Zhou et al., 2012; Oard, 1998; McCarley, 1999). A monolingual IR system is then used to obtain search results.

CLIR is an active field of research, and previ-

Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, pages 134–141
July 5 - July 10, 2020.

ous works suggest that the performance of CLIR correlates highly with the quality of the MT (Zhu and Wang, 2006; Nie, 2010; Yarmohammadi et al., 2019). Therefore, we expect IR metrics to be good indicators of the quality of translations. Unfortunately, there is currently no publicly available tool to facilitate research in this area, and this motivates us to design and implement CLIReval.

CLIReval is a lightweight python-based MT evaluation toolkit that consumes the same inputs as other automatic MT evaluation tools such as multi-bleu.perl and SacreBLEU (Post, 2018) and *does not require any additional annotated CLIR data.* Instead, it automatically transforms inputs into a synthetic CLIR dataset on the fly with the help of an Information Retrieval (IR) system. It implements the document translation approach to CLIR, where MT translations are viewed as documents and indexed using a commonly-used search engine (Elasticsearch).

As a case study, we test CLIReval on the metrics shared task of WMT2019 (Ma et al., 2019), which measures the Pearson correlations (r) between automatically generated MT metrics and human judgments. Results show that CLIReval consistently performs at the level of $r \geq 0.9$ and is on par or even outperforms popular metrics such as BLEU on multiple language directions. Further, this is achieved without using external data or doing domain-based parameter tuning. These promising results highlight the potential of CLIR as a proxy task for MT evaluation, and we hope CLIReval can facilitate future research in this area.

Our key contributions in this work can be summarized as follows:

1. We release **CLIReval**,[1] an open-source toolkit that evaluates the quality of MT outputs in the context of a CLIR system, *without the need for any actual CLIR dataset*. The only inputs required to the tool are the translations and the references. It is easy to use in that with a single script, the tool will create a synthetic CLIR dataset, index the translations as documents, and report metrics such as mean average precision.

2. We demonstrate that CLIReval can perform as well as popular intrinsic MT metrics on recent WMT metrics shared task, without supervision from external datasets and domain-based parameter tuning. Results suggest that CLIR is a feasible proxy task for MT evaluation and is worth further exploration in future research.

[1] https://github.com/ssun32/CLIReval

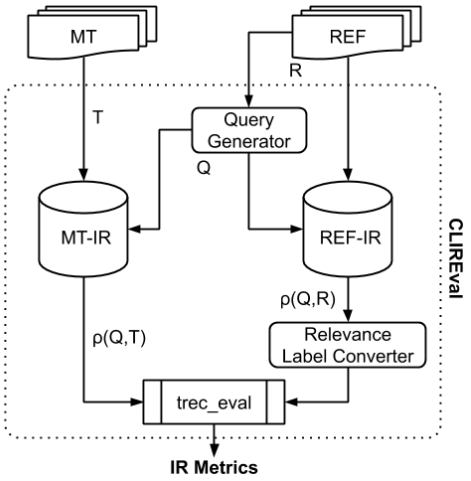

Figure 1: The system architecture of CLIReval. Documents from input files are separately indexed into two instances of IR systems. Generated search queries are used to query both IR instances. Search scores from REF-IR are converted to discrete relevance judgment labels as required by trec_eval. Finally, CLIReval uses trec_eval to calculate IR metrics.

2 Approach

Given a set of source documents S, an MT system ϕ converts S into a set of translated documents, $T = \phi(S)$. Intrinsic MT metrics directly calculate an aggregated score between the sentences in T and sentences in R, where R is a set of reference documents.[2]

We propose an alternative way to evaluate ϕ by first converting it into a proxy CLIR task and then evaluate the MT system with extrinsic IR metrics. First, CLIReval extracts a set of synthetic search queries Q from R. Second, given a monolingual information retrieval (IR) engine ρ, we can run these queries Q over the document collection R to obtain a set of "relevant" documents for Q. We use the notation $\rho(Q, R)$ to refer to this set of desired returned search results.

Now, our goal is to evaluate the quality of the translation $T = \phi(S)$ under the same IR engine ρ. We index the documents T into the IR engine and submit the same queries to obtain the search

[2] When document boundaries are not defined, CLIReval automatically creates artificial document boundaries. The default option is to treat each sentence as a document for retrieval purposes.

results $\rho(Q, T)$. Finally, we can measure the performance of the CLIR system by comparing $\rho(Q, T)$ to $\rho(Q, R)$, and calculating IR metrics such as mean average precision.

This approach makes several assumptions. First, CLIReval implements the *document translation approach* to CLIR and evaluates MT quality in that context; additionally, we assume that ρ is a robust and reasonable IR engine that can be used across a wide range of situations. Second, we assume R contains the "correct" translations of S, and that $\rho(Q, R)$ is a good approximation of the optimal search results. Third, we assume that automatically-generated Q can mimic that actual information needs of manually-crafted queries. If these caveats are acknowledged, then CLIReval is a reasonable tool for MT evaluation.

3 Design and Implementation Details

Figure 1 presents the system architecture of CLIReval. The only necessary inputs are 1) a system output translation (MT) file and 2) a reference (REF) file. CLIReval executes the following steps:

1. Separately index documents in MT and REF files into two instances of the Information Retrieval (IR) system, we refer to them as MT-IR and REF-IR.

2. Convert text in the REF file into search queries with the *Query Generator* module.

3. Query both instances of IR system with the same set of generated search queries.

4. Convert search scores from REF-IR to discrete relevance labels with the *Relevance Label Converter*.

5. Finally, CLIReval evaluates the search results from MT-IR and relevance judgment labels from REF-IR with trec_eval,[3] a standard evaluation toolkit used by the information retrieval community.

We emphasize that the above steps are achieved with a single easy-to-use script: CLIReval is as simple as executing the following command:

```
python evaluate.py [ref file] [mt file]
```

where the inputs are standard text files that one might pass to `multi-bleu.perl`, or standard SGML files that one might pass to `mteval-v13a.pl`, both of which are common BLEU scripts for MT.[4]

3.1 Input files

CLIReval ingests a system output translation (MT) file which contains documents translated by an MT system and a reference (REF) file, which contains reference translations of the same source documents. Our system supports two input file formats:

1. The *SGML format* commonly used by the news translation shared task from the annual conference on machine translation (Barrault et al., 2019). This is also the input format required by the NIST BLEU scoring tool.[5] In a SGML file, every translated sentence segment is placed in a <seg> tag, and sentence segments belonging to the same document are placed in the same <doc> tag. Every <doc> tag must also contain a unique document id attribute used to identify the document.

2. A *text file* where each line contains a sentence. A user can supply an optional mapping file that maps a line number to a (document id and, segment id) tuple. If a mapping file is not specified, CLIReval will create an artificial document boundary every N sentences.[6]

For either format, the number of documents in the MT file must be equivalent to the number of documents in the REF file. Further, the number of sentence segments in a machine translated document must also match the number of sentence segments in the corresponding reference document.

3.2 Query Generator

The query generator module ingests data in the REF file and automatically generates search queries. CLIReval has two modes for query generation, which can be specified with the query_mode argument:

1. In **sentences** mode, the query generator extracts all reference sentences from the input

[3] https://github.com/usnistgov/trec_eval

[4] https://github.com/moses-smt/mosesdecoder/blob/master/scripts/generic/

[5] ftp://jaguar.ncsl.nist.gov/mt/resources/mteval-v13.pl

[6] N can be specified with the doc_length argument. The default value is 1, which means every sentence is treated as a document.

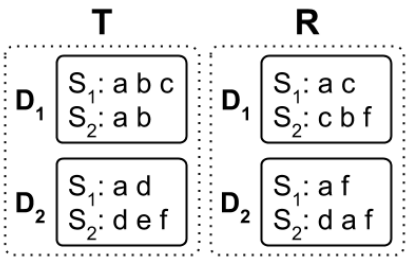

Figure 2: R is a set of sample reference documents and each document contains two sentences, while T is a set of sample translated documents.

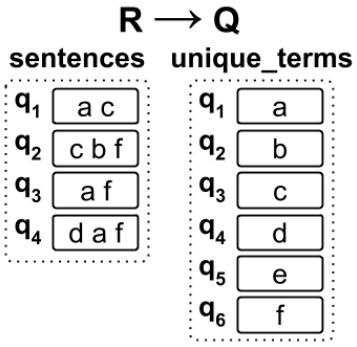

Figure 3: Sample outputs from the *query generator*. In *sentences* mode, all sentences from R (Figure 2) are used as search queries while in the *unique_terms* mode, the unique terms in R are the search queries.

REF file and treats every sentence as a search query string. This is inspired by Sasaki et al. (2018), who use the first sentences of documents as queries.

2. In **unique_terms** mode, the query generator treats all unique terms as queries. For Elasticsearch, these terms can be obtained from the term vectors of all indexed documents.

We recognize that using sentences or unique terms as queries might be less ideal than using real search queries, but getting relevant human-generated queries can be time-consuming and expensive. Our query generation methods are cheap and fast, which enables quick experimentation. Examples of R and T are shown in Figure 2, and the resulting queries generated from R are shown in Figure 3. In the example, we have two documents D_1 and D_2 each with two sentences S_1 and S_2. In the **sentence** mode for query generation, each of the four sentences in R are used as queries; in the **unique_terms** mode, the 6 vocabulary words are extracted as query.

3.3 Information Retrieval (IR) System

To ensure consistent and reproducible results, we choose Elasticsearch[7] as the default backend IR system for CLIReval and adopt well-tested search configurations.[8] Elasticsearch is an open-source, lightweight, and fast search engine written in Java. We pick Elasticsearch for three reasons:

First, Elasticsearch has built-in analyzers for a wide variety of languages, which allows CLIReval to support many translation tasks beyond English as the target language. Analyzers are Elasticsearch modules that preprocess and tokenize queries and documents according to language-specific rules. It also implements stopwords removal and stemming. These are important operations that affect the quality of search results.

Second, Elasticsearch implements many competitive retrieval models used by IR researchers and practitioners. By default, CLIReval uses the Okapi BM25 (Robertson et al., 2009) score to measure the degree of similarity of documents to a given search query. Note that BM25 shows strong performances on many datasets (Chapelle and Chang, 2011; McDonald et al., 2018) and frequently outperforms newer "state of the art" methods (Guo et al., 2016). It is also fast to compute, allowing CLIReval to run in a highly efficient manner.

Third, Elasticsearch is a widely used search engine solution that is supported on various platforms. This increases the ease of installation for users of CLIReval.

CLIReval separately indexes the documents from MT and REF files into two instances of Elasticsearch. It then queries the Elasticsearch instances with the generated query strings. For every query, Elasticsearch returns the top 100 documents ranked by BM25 scores. Since trec_eval only accepts discrete relevance judgment labels, the relevance label converter module is used to convert search scores from REF-IR into discrete labels.

3.4 Relevance Label Converter

We implement three ways of converting raw BM25 scores of REF-IR into discrete *relevance judgment labels*:

The **query_in_document** method (Schamoni et al., 2014; Sasaki et al., 2018) assigns 1 to a document if and only if the given search query

[7] https://www.elastic.co/

[8] CLIReval is flexible and users can easily replace Elasticsearch with their own IR system.

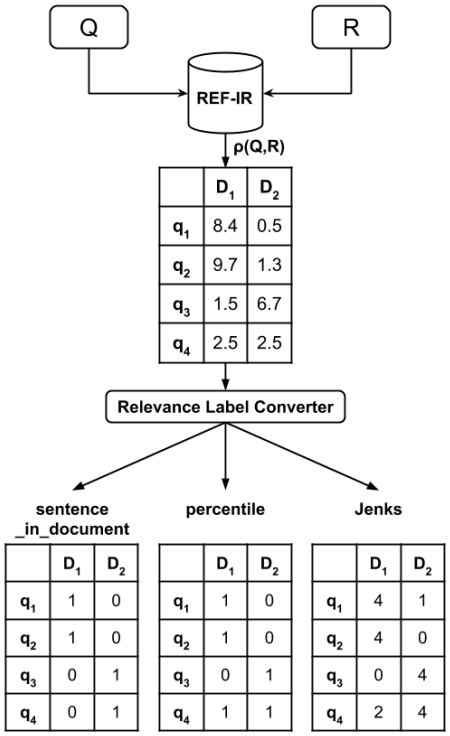

Figure 4: Given queries from the query generator and documents from R, we can obtain relevance scores from an IR system. The *relevance label converter* then converts those relevance scores into discrete relevance labels via different conversion modes.

is extracted from that document. Consequently, there will only be one relevant document per search query.

The **percentile** method assigns 1 to documents with BM25 scores in the top 25 percentile of all document scores returned by the IR system and 0 otherwise. The cutoff percentile value can be adjusted with the *n_percentile* argument.

Th **Jenks** methods uses Jenks natural breaks optimization[9] to automatically break a list of BM25 scores into different classes. This is achieved by minimizing the variance of BM25 scores within a class and at the same time maximize the variance of average BM25 scores between classes (McMaster and McMaster, 2002). Following the conventions of publicly available IR datasets (Chapelle and Chang, 2011; Qin and Liu, 2013), we break the BM25 scores into 5 relevance judgment classes, where 4 indicates that a document is highly relevant to a given query and 0 indicates that a document is not relevant to a given query. For each query, CLIReval normalizes the BM25 scores of retrieved documents to the range [0, 1] and use Jenks natural breaks optimization to convert the BM25 scores into discrete relevance judgment labels. Users can specify the number of classes with the *jenks_nb_class* argument.

Figure 4 illustrates an example of how relevance labels are generated for each query-document pair using the generated query Q (see Section 3.2 and the reference documents R provided by the user. First, raw BM25 scores are obtained by indexing R in an IR system and searching with Q. These scores are then converted to discrete labels in one of three ways.

3.5 IR Metrics

To summarize: after the queries and relevance labels are prepared (as in Section 3.2 and 3.4), the MT output T (e.g. Figure 2, left) is indexed into another IR system. Finally, we run the queries Q through this MT-IR system to obtain document scores $\rho(Q, T)$ (e.g. Figure 1, left branch), which can be evaluated with respect to the relevance labels. We do this final evaluation with the standard trec_eval toolkit.

The trec_eval toolkit returns a large number of IR metrics but CLIReval is configured to return only two of the most popular IR metrics by default:

- **Mean average precision (MAP)** is the mean of the average precision scores for each query (Buckley and Voorhees, 2005).
- **Normalized discounted cumulative gain (NDCG)** is a metric that measures the usefulness of documents based on their ranks in the search results (Järvelin and Kekäläinen, 2002) and is normalized to [0, 1].

We choose MAP because it is a widely understood metric, and NDCG because it allows for multiple levels of relevance labels. We follow standard practice in IR benchmark datasets such as Chapelle and Chang (2011) and calculate both metrics at the cutoff threshold of 10 documents. We name these metrics as MAP@10 and NDCG@10.

3.6 Installation

CLIReval is written in Python 3 and works on Python 3.5 and later. Elasticsearch requires at least Java 8. We provide a shell script that automatically downloads and installs Elasticsearch 6.5.3 and the latest version of trec_eval. It also installs additional

[9] https://en.wikipedia.org/wiki/Jenks_natural_breaks_optimization

Elasticsearch plugins that support additional languages. In total, CLIReval has built-in support for 36 languages and for unsupported languages, it will fall back to the default standard analyzer, which is based on the Unicode text segmentation algorithm.[10] We tested CLIReval extensively in the Unix/Linux environment, but it should work in other environments with minimal modification.

4 Case Study

4.1 WMT metrics shared task

To demonstrate the utility of CLIReval, we test it on the metrics shared task of WMT2019.[11] The metrics task (Ma et al., 2019) is designed to evaluate outputs from automatic MT metrics against actual human ratings on machine translation systems. The goal is to find evaluation methods that have high Pearson correlations with human judgments. For every system in every language direction, we compute multiple system-level scores (different IR metrics) with CLIReval.

In total, there are 18 language directions, and for every language direction, a reference file and 11 to 22 system generated translation files are provided. In every reference file, there are around 1000 to 2000 sentences in 70 to 140 documents. The only exceptions are French-German and German-French, where all sentences are placed in the same document. Since document boundaries are not clearly defined in these language directions, we are excluding them from this case study.

4.2 Run Time

We used an Intel Xeon E5 Linux server with 64GB RAM. For every language direction, CLIReval runs consistently at the rate of around 0.2 to 0.3 seconds per document and it takes less than a minute to get results.

4.3 Results

We use the official evaluation scripts[12] to compute linear correlations between IR metrics and human judgments.

Table 1 presents the results for 16 language directions and IR metrics perform well. On Jenks mode, NDCG@10 outperforms BLEU and NIST on 10 out of the 16 language directions. Further, the 4 IR metrics collectively hold the top scores for 6 language directions. BEER seems to be a little bit better than the IR metrics, claiming the top spot for 7 language directions. Note that the participating BEER system is trained on provided in-domain data, while we are getting comparable results without any tuning. It is also worth pointing out that the intrinsic MT metrics work at sentence level while in comparison, CLIReval works at the document-level. Nonetheless, the results are encouraging and show the potential of CLIR as a proxy task for MT evaluation.

4.4 Analysis: BLEU vs. NDCG

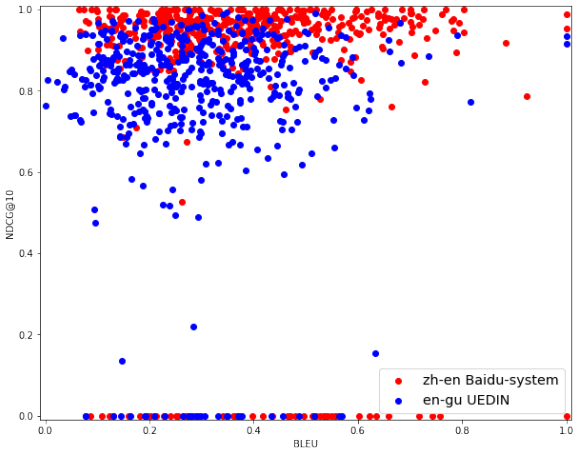

Figure 5: Scatterplot of sentence-level NDCG@10 vs sentence-level BLEU on zh-en and en-gu. For better visualization, only 300 random samples from each language direction are shown.

To get a deeper comparison between CLIReval and the most popular MT metric, BLEU, we randomly select two systems (Baidu-system for zh-en and UEDIN for en-gu) and calculate sentence-level BLEU and sentence-level NDCG@10 scores[13] on both systems. As we can see in Figure 5, there is no clear correlation between sentence-level NDCG@10 and sentence-level BLEU scores. To be more exact, the Pearson correlations between the two metrics is almost non-existent, at -0.021 and -0.032 for zh-en and en-gu respectively. This shows that the two metrics are qualitatively different and contribute different perspectives to MT evaluation.

5 Conclusions

We present CLIReval, an open-source python-based evaluation toolkit for machine translation.

[10] https://unicode.org/reports/tr29/

[11] http://www.statmt.org/wmt19/

[12] http://ufallab.ms.mff.cuni.cz/~bojar/wmt18-metrics-task-package.tgz

[13] calculated with CLIReval using default arguments.

					query_in_document		Jenks	
LD	**BLEU**	**NIST**	**TER**	**BEER**	**MAP@10**	**NDCG@10**	**MAP@10**	**NDCG@10**
de→cs	0.941	0.954	0.890	0.978	0.971	0.968	0.965	**0.991**
de→en	0.849	0.813	0.874	**0.906**	0.865	0.869	0.654	0.858
en→cs	0.897	0.896	0.980	**0.990**	0.882	0.889	0.909	0.983
en→de	0.921	0.321	0.969	**0.983**	0.953	0.953	0.977	0.982
en→fi	0.969	0.971	0.981	**0.989**	0.915	0.906	0.927	0.944
en→gu	0.737	0.786	0.865	0.829	**0.912**	0.909	0.833	0.847
en→kk	0.852	0.930	0.940	0.971	**0.982**	**0.982**	0.963	0.968
en→lt	0.989	0.993	**0.994**	0.982	0.776	0.791	0.903	0.916
en→ru	0.986	0.988	**0.995**	0.977	0.865	0.862	0.980	0.953
en→zh	0.901	0.884	0.856	0.803	0.928	**0.930**	0.772	0.902
fi→en	0.982	0.986	0.984	**0.993**	0.956	0.955	0.944	0.960
gu→en	0.834	0.930	0.890	**0.952**	0.814	0.809	0.782	0.824
kk→en	0.946	0.942	0.799	**0.986**	0.970	0.968	**0.986**	0.983
lt→en	**0.961**	0.944	0.960	0.947	0.636	0.612	0.929	0.865
ru→en	0.879	0.925	0.917	0.915	0.922	0.920	0.866	**0.961**
zh→en	0.899	0.921	0.840	0.942	0.930	0.922	0.622	**0.957**

Table 1: Pearson correlations (r) of various metrics against human judgments. Best scores for every language direction are highlighted in bold. Note that BEER is trained on in-domain resources from the WMT2019 metrics task. We show MAP@10 and NDCG@10 scores for CLIReval with two relevance label conversion settings.

Rather than directly evaluating translated sentences against reference sentences, CLIReval transforms the inputs into the closely related task of CLIR, without the need for annotated CLIR dataset.

The aim of this project is not to replace current automatic evaluation metrics or fix the limitations in those metrics, but to bridge the gap between machine translation and cross-lingual information retrieval and to show that CLIR is a feasible proxy task for MT evaluation.

Our case study on the WMT2019 metrics shared task further highlights the potential of CLIR as a proxy task for MT evaluation, and we hope that CLIReval can facilitate future research in this area.

Acknowledgement

We want to thank Sorami Hisamoto and Muhammad Mahbubur Rahman for their initial code and guidance on best practices for Elasticsearch.

References

Satanjeev Banerjee and Alon Lavie. 2005. Meteor: An automatic metric for mt evaluation with improved correlation with human judgments. In *Proceedings of the acl workshop on intrinsic and extrinsic evaluation measures for machine translation and/or summarization*, pages 65–72.

Loïc Barrault, Ondřej Bojar, Marta R Costa-jussà, Christian Federmann, Mark Fishel, Yvette Graham, Barry Haddow, Matthias Huck, Philipp Koehn, Shervin Malmasi, et al. 2019. Findings of the 2019 conference on machine translation (wmt19). In *Proceedings of the Fourth Conference on Machine Translation (Volume 2: Shared Task Papers, Day 1)*, pages 1–61.

Jan Berka, Martin Černỳ, and Ondřej Bojar. 2011. Quiz-based evaluation of machine translation. *The Prague Bulletin of Mathematical Linguistics*, 95:77–86.

Chris Buckley and Ellen Voorhees. 2005. Retrieval system evaluation. *TREC: Experiment and Evaluation in Information Retrieval*, pages 53–75.

Chris Callison-Burch, Miles Osborne, and Philipp Koehn. 2006. Re-evaluation the role of bleu in machine translation research. In *11th Conference of the European Chapter of the Association for Computational Linguistics*.

Olivier Chapelle and Yi Chang. 2011. Yahoo! learning to rank challenge overview. In *Proceedings of the Learning to Rank Challenge*, pages 1–24.

Jiafeng Guo, Yixing Fan, Qingyao Ai, and W Bruce Croft. 2016. A deep relevance matching model for ad-hoc retrieval. In *Proceedings of the 25th ACM International on Conference on Information and Knowledge Management*, pages 55–64.

Kalervo Järvelin and Jaana Kekäläinen. 2002. Cumulated gain-based evaluation of ir techniques. *ACM Transactions on Information Systems (TOIS)*, 20(4):422–446.

Qingsong Ma, Johnny Wei, Ondřej Bojar, and Yvette Graham. 2019. Results of the wmt19 metrics shared task: Segment-level and strong mt systems pose big challenges. In *Proceedings of the Fourth Conference on Machine Translation (Volume 2: Shared Task Papers, Day 1)*, pages 62–90.

J Scott McCarley. 1999. Should we translate the documents or the queries in cross-language information retrieval? In *Proceedings of the 37th annual meeting of the Association for Computational Linguistics on Computational Linguistics*, pages 208–214. Association for Computational Linguistics.

Ryan McDonald, George Brokos, and Ion Androutsopoulos. 2018. Deep relevance ranking using enhanced document-query interactions. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 1849–1860.

Robert McMaster and Susanna McMaster. 2002. A history of twentieth-century american academic cartography. *Cartography and Geographic Information Science*, 29(3):305–321.

Jian-Yun Nie. 2010. Cross-language information retrieval. *Synthesis Lectures on Human Language Technologies*, 3(1):1–125.

Douglas W Oard. 1998. A comparative study of query and document translation for cross-language information retrieval. In *Conference of the Association for Machine Translation in the Americas*, pages 472–483. Springer.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting on association for computational linguistics*, pages 311–318. Association for Computational Linguistics.

Matt Post. 2018. A call for clarity in reporting bleu scores. In *Proceedings of the Third Conference on Machine Translation: Research Papers*, pages 186–191.

Tao Qin and Tie-Yan Liu. 2013. Introducing LETOR 4.0 datasets. *CoRR*, abs/1306.2597.

Stephen Robertson, Hugo Zaragoza, et al. 2009. The probabilistic relevance framework: Bm25 and beyond. *Foundations and Trends® in Information Retrieval*, 3(4):333–389.

Shota Sasaki, Shuo Sun, Shigehiko Schamoni, Kevin Duh, and Kentaro Inui. 2018. Cross-lingual learning-to-rank with shared representations. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pages 458–463.

Carolina Scarton and Lucia Specia. 2016. A reading comprehension corpus for machine translation evaluation. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, pages 3652–3658.

Shigehiko Schamoni, Felix Hieber, Artem Sokolov, and Stefan Riezler. 2014. Learning translational and knowledge-based similarities from relevance rankings for cross-language retrieval. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 488–494.

Matthew Snover, Bonnie Dorr, Richard Schwartz, Linnea Micciulla, and John Makhoul. 2006. A study of translation edit rate with targeted human annotation.

Miloš Stanojević and Khalil Sima'an. 2014. Beer: Better evaluation as ranking. In *Proceedings of the Ninth Workshop on Statistical Machine Translation*, pages 414–419.

Mahsa Yarmohammadi, Xutai Ma, Sorami Hisamoto, Muhammad Rahman, Yiming Wang, Hainan Xu, Daniel Povey, Philipp Koehn, and Kevin Duh. 2019. Robust document representations for cross-lingual information retrieval in low-resource settings. In *Proceedings of Machine Translation Summit XVII Volume 1: Research Track*, pages 12–20, Dublin, Ireland. European Association for Machine Translation.

Dong Zhou, Mark Truran, Tim Brailsford, Vincent Wade, and Helen Ashman. 2012. Translation techniques in cross-language information retrieval. *ACM Computing Surveys (CSUR)*, 45(1):1–44.

Jiang Zhu and Haifeng Wang. 2006. The effect of translation quality in mt-based cross-language information retrieval. In *Proceedings of the 21st International Conference on Computational Linguistics and the 44th annual meeting of the Association for Computational Linguistics*, pages 593–600. Association for Computational Linguistics.