Trustful Test Suites for Natural Language Processing

Mariana Cabeça[1]*, Marianna Buchicchio*[2]*, Helena Moniz*[3]

[1] Faculdade de Letras da Universidade de Lisboa / Unbabel
[2] Unbabel
[3] Faculdade de Letras da Universidade de Lisboa / INESC-ID

Abstract

Machine Translation (MT) research has witnessed continuous growth, accompanied by an increasing demand for automated error detection and correction in textual content. In response, Unbabel has developed a hybrid approach that combines machine translation with human editors in post-edition (PE) to provide high-quality translations. To facilitate the tasks of post-editors, Unbabel has created a proprietary error detection tool named Smartcheck, designed to identify errors and provide correction suggestions. Traditionally, the evaluation of translation errors relies on carefully curated annotated texts, categorized based on error types, which serve as the evaluation standard or Test Suites for assessing the accuracy of machine translation systems. However, it is crucial to consider that the effectiveness of evaluation sets can significantly impact the outcomes of evaluations. In fact, if evaluation sets do not accurately represent the content or possess inherent flaws, the decisions made based on such evaluations may inadvertently yield undesired effects. Hence, it is of utmost importance to employ suitable datasets containing representative data of the structures needed for each system, including Smartcheck. In this paper we present the methodology that has been developed and implemented to create reliable and revised Test Suites specifically designed for the evaluation process of MT systems and error detection tools. By using these meticulously curated Test Suites to evaluate proprietary systems and tools, we can ensure the trustworthiness of the conclusions and decisions derived from the evaluations. This methodology accomplished robust identification of problematic error types, grammar-checking rules, and language- and/or register-specific issues, leading to the adoption of effective production measures. With the integration of Smartcheck's reliable and accurate correction suggestions and the improvements made to the post-edition revision process, the work presented herein led to a noticeable improvement in the translation quality delivered to customers.

Keywords: Grammar Error Detection, performance assessment, Test Suites, NLP systems evaluation

Resumo

À medida que o estudo da Tradução Automática (TA) tem vindo a expandir-se ao longo do tempo, a necessidade de detetar e corrigir erros em textos tem também aumentado. Neste sentido, a Unbabel combina tradução automática com pós-edição feita por tradutores e linguistas, para, assim, obter traduções de boa qualidade. De modo a assistir os editores nas suas tarefas, foi desenvolvida uma ferramenta proprietária de deteção de erros denominada de *Smartcheck*, que identifica erros e sugere correções para os mesmos. O método mais recente de identificação de erros de tradução baseia-se em textos previamente pós-editados e anotados (categorizando cada erro de acordo com as suas características), que são fornecidos aos sistemas de tradução automática como sendo o padrão de avaliação ou o *corpus* de teste para avaliar a precisão dos sistemas de tradução. Contudo, é de extrema importância considerar que a eficácia dos *corpora* de teste pode ter um impacto significativo nos resultados das avaliações. De facto, se

estes *corpora* não representarem de forma precisa e representativa o conteúdo, as decisões tomadas com base nas avaliações podem inadvertidamente produzir efeitos indesejados. Assim, é de extrema importância criar *corpora* de teste adequados, cujos dados sejam representativos das estruturas necessárias para cada sistema, incluindo ferramentas como o *Smartcheck*. Neste sentido, o presente trabalho permitiu criar e implementar uma nova metodologia de criação de *corpus* de teste bem fundamentada, que pode ser aplicada no processo de avaliação de sistemas de tradução automática e de ferramentas de deteção de erros. Recorrendo à aplicação deste *corpus* de avaliação, tornou-se possível confiar nas conclusões e ilações obtidas posteriormente. Esta metodologia possibilitou também que todo o processo de identificação de erros e avaliação de regras gramaticais se tornasse mais robusto, bem como o de deteção de problemas específicos por língua e/ou registo, permitindo, assim, adotar diversas medidas necessárias em produção. Por meio de sugestões de correção de erros válidas do *Smartcheck* e das melhorias aplicadas ao processo de pós-edição, o presente trabalho demonstrou ser possível aferir a qualidade das traduções que são entregues a diferentes clientes de forma mais cuidada e consistente.

Palavras-chave: Sistemas de Deteção Automática de Erros, avaliação de desempenho, *corpus* de teste, avaliação de sistemas de PLN

1. **Introduction**

In recent years, there has been a significant surge in interest in the automation of Machine Translation (MT). While MT offers faster, more efficient, and cost-effective translation, it has not yet achieved the quality standard set by human translations. In light of this limitation, the technologies developed at Unbabel have successfully addressed this issue by effectively combining the advantages of MT with the high-quality assurance provided by post-editing.

Evaluation plays a crucial role in the life cycle of any system. Whether conducted manually or automatically through the use of metrics such as BLUE (Papineni et al., 2002), METEOR (Lavie & Denkowski, 2009) or COMET (Rei et al., 2020), it is essential to have a means of measuring the quality of the output. However, one aspect of evaluation that often goes overlooked is the validity and trustworthiness of the conclusions drawn from these assessments. Hence, it is of utmost importance to establish a robust evaluation standard that encompasses representative, relevant, and accurate data pertaining to the content being translated. This approach enables the identification of limitations and facilitates their resolution.

In an attempt to further enhance the quality of translations, Unbabel created Smartcheck, a proprietary Grammar Error Detection tool that aids post-editors during the post-edition stage in order to improve their performance in terms of efficiency and accuracy. Smartcheck highlights possible errors and provides suggestions, thereby enabling post-editors to complete their tasks quicker. In essence, it assists them in error detection and correction, ensuring the delivery of high-quality translations to clients in a timely manner.

The work described hereafter aims to improve Smartcheck's performance and underscore the significance of fairness and quality in evaluation data. Specifically, it emphasizes the creation of reliable Test Suites based on a robust methodology and representative data of high quality. Although Smartcheck has several distinct modules, this study will focus on the assessment of rules and spell checkers. Consequently, the objectives of this study were threefold: i) implementation of a methodology on creating reliable Test Suites for testing a proprietary tool on error detection and editing suggestions; ii) evaluation of the performance of this tool; and iii) contribution to the improvement of quality based on the edits suggested.

2. State of the art

MT research has witnessed remarkable growth in recent years, despite encountering certain challenges along the way. However, there remains a great amount of work to be done to ensure high-quality outputs and complete system automation. When a new MT system is trained, it is crucial to analyze its output to detect errors. These errors serve as valuable opportunities to improve the system's output, as they allow for a deeper understanding of the most recurrent errors. This analysis enables the identification of specific linguistic structures that require greater attention to further refine the MT system.

Current systems still struggle to ensure fluency and coherence in their translated sentences. Consequently, numerous attempts have been made to evaluate and monitor translation quality. These efforts encompass both manual revision and automated metrics. In terms of manual evaluation, the state-of-the-art approach relies on the Multidimensional Quality Metrics (MQM) framework (Lommel et al., 2014). The MQM metric is devised based on error annotations with different degrees of severity and aims to achieve a high level of granularity in the evaluation of translations. However, the manual process of annotating errors is inherently time-consuming, as every translation error in a given text must be identified and classified according to this typology.

On the other hand, automated metrics provide a general score for a given system's output within a matter of seconds. Nevertheless, these metrics are considered to have a low level of correlation with human judgment. Despite their efficiency in terms of speed, their ability to accurately align with human assessment is frequently questioned.

The demand for a more detailed and time-efficient evaluation process has significantly increased. As such, the MT community began exploring alternative evaluation approaches as early as the 1990s. Approaches such as the utilization of test *corpora* and Test Suites, which are distinct techniques that should not be confused with one another. The main distinction lies in the fact that test *corpora* serve as repositories of extensive and potentially unrefined data, while Test Suites consist of carefully curated sets of tests that accurately represent the linguistic structures under analysis.

Test *corpora* often lack meticulousness, making it challenging to isolate specific linguistic phenomena being tested. Furthermore, the absence of annotations in most *corpora* further complicates the evaluation process. Conversely, Test Suites, the focus of our work, comprise lists of sentences deliberately compiled to create a controlled *corpus* of exemplary data, referred to as gold standard data. These Test Suites facilitate diagnostic evaluation of an MT system, as the input used for testing is pre-checked, allowing for control over the vocabulary and the specific linguistic phenomena being tested. This evaluation method proves particularly valuable when it is necessary to present language phenomena in a comprehensive and systematic manner or generate various combinations of phenomena (Balkan et al., 1994).

To properly evaluate a system using Test Suites, the set of tests must be constructed in a methodological way by following a specific approach that is considered appropriate for the evaluation's primary objective (King & Falkedal, 1990). This is crucial because, as Dale et al. (2012, p. 58) state, "if we cannot entirely trust our gold-standard data, then we cannot place too much trust in the results of evaluations carried out using that data". Therefore, the creation of high-quality Test Suites demands meticulous planning, coherence, and systematic execution.

According to Balkan (1994), there are different approaches when constructing Test Suites, one of which is the bottom-up approach. In the bottom-up approach, the system is tested and its functions are analyzed, treating them as attributes. For instance, consider the case of spell checkers. Their functions consist of detecting misspelled words and providing plausible corrections; therefore, their reportable attributes consist of detected misspelled words and of the corresponding correct form, found among the corrections the system proposes. Subsequently, each attribute

is associated with a value, usually a percentage, which is calculated by comparing it to a standard, such as Test Suites.

However, for Test Suites to serve their purpose in evaluating spell checkers or MT systems in general, it is essential that the included phenomena are specifically relevant to the intended application (Balkan, 1994). In other words, Test Suites must comprise representative examples of the phenomena or content that one aims to evaluate. Within the bottom-up approach, there are multiple options for constructing system-specific Test Suites, depending on the type of evaluation required. Nonetheless, two distinct evaluation scenarios can be identified: the "black box" scenario, wherein the evaluator lacks access to the internal workings of the system but can still test hypotheses about its internal mechanisms, and the "glass box" scenario, wherein the evaluator has access to the system's rules. In the latter scenario, the writer of the Test Suite can tailor it to the system's rules, enabling a diagnostic evaluation to identify the root causes of system errors through a root-cause analysis.

Test Suites play a vital role in conducting meticulous and comprehensive evaluations, and their applicability extends beyond machine translation systems (Avramidis et al., 2018). They can be employed to monitor and evaluate other complex systems, including grammatical error detection (GED) and correction (GEC) tools. These tools aim to improve the overall quality of final translations, necessitating thorough evaluation procedures.

GED systems receive potentially erroneous sentences as input and identify tokens that violate specific linguistic rules. On the other hand, GEC systems are tasked with accurately correcting the identified errors while preserving the original meaning of the sentence, thereby optimizing translation quality. However, the process of error detection is intricate, involving various Natural Language Processing (NLP) tools and complex dependencies between tokens. Unlike humans, whose inherent linguistic competence enables them to intuitively identify errors in sentences without explicit knowledge of grammar and language rules, systems lack this intuitive decision-making ability. Instead, systems rely on explicit universal dependencies, word classifications, and language rules to detect errors. Given the complexity of these systems, their evaluation must be detailed and precise.

To achieve an accurate evaluation, one effective approach involves using trustworthy Test Suites as a complementary component of the evaluation process. Test Suites help ensure the evaluation is conducted with the necessary level of scrutiny and accuracy, thereby improving the reliability of the findings. By relying on Test Suites, a comprehensive framework that accounts for various linguistic phenomena can be established, facilitating a thorough examination of the system's performance.

Test Suites, according to Balkan (1994, p. 1), are carefully put-together collections of sentences. They're often created artificially, with each sentence crafted to specifically check how a system deals with specific linguistic phenomena or sets thereof. These inputs can assume various forms, such as complete sentences, sentence fragments, or even sequences of sentences. When these test inputs are carefully chosen ahead of time, it becomes possible to control both the words used and the specific language aspects being tested. This control lets the evaluator focus entirely on how the system handles the specific language tasks at hand, without being distracted by issues related to vocabulary, as explained by Balkan et al. (1994, p. 53).

This evaluation methodology proves especially advantageous when addressing the following three significant objectives, as delineated by Balkan et al. (1994): i) presenting linguistic phenomena in a comprehensive and methodical manner; ii) generating various potential combinations of linguistic phenomena; and iii) systematically deriving negative data from positive data, achieved by deliberately contravening grammatical constraints associated with the positive data set (Balkan et al., 1994, p. 53). Thus, the use of Test Suites serves as an invaluable means to meticulously evaluate systems. Ultimately, they diver from test sets in the sense that they are not corseted to a certain percentage of the total amount of data in a *corpus*, but rather tailored to provide examples of linguistic structures to be consistently tested.

3. **Methodology**

As previously mentioned, ensuring the quality of MT involves employing post-edition. Therefore, post-editors play a vital role in the quality assurance process and in order to facilitate their work and motivate them to accomplish tasks, it is of utmost importance to provide them assistance so as to minimize errors while considering time constraints. This is where Smartcheck assumes a crucial role in supporting post-editors. The primary objective of Smartcheck is to assist post-editors in two fundamental ways: first, by detecting potential errors, and second, by automatically suggesting corrections aligned with the customer's style guidelines and many other requirements. Smartcheck examines errors such as inconsistencies in register and formality, adherence to specific client rules, and overall text coherence. Meanwhile, the task of spelling verification is delegated to external NLP services, such as a word aligner, a syntax parser, and a spell checker. One key feature of Smartcheck lies in its integration of custom language rules implemented through a proprietary programming language known as SURF.[1] These rules address various types of issues, including style, fluency, grammar-related errors, dependency problems, among others. Consequently, Smartcheck represents the culmination of multiple NLP modules and hardcoded rules tailored to different language pairs and can be regarded as an augmented multilingual version of a spell checker, as it not only analyzes grammar and orthography but also delves into morphology and style-adapted client rules.

It is important to note that Smartcheck does not substitute the erroneous form with the correct one. Hence, it should be classified as a GED tool rather than a GEC tool. The ultimate decision rests with the post-editor, who has the final say in accepting or rejecting the suggestions. Thanks to Smartcheck, post-editors can achieve higher translation quality, enabling these refined texts to be utilized as training data for MT systems and driving their continual improvement. This is due to the fact that MT systems operate on the principles of machine learning, wherein the quality of translations relies heavily on the training data provided. When high-quality translations are used as training data, the machine learns to emulate the desired output. Conversely, if the training data is of low quality, the resulting output is likely to be inadequate. In essence, the quality of the input data directly affects the quality of the output produced by the MT system. Therefore, it is crucial to ensure the use of reliable and accurate data during the training phase to achieve desirable translation quality.

The methodology presented in this section was established to evaluate the performance of Smartcheck. The initial step involved conducting a baseline analysis to establish a benchmark for future comparisons. Upon gathering these results, we formulated a hypothesis that attributed the low metric values to the evaluation dataframe employed for assessing the system's performance, rather than the actual grammar-checking rules. Consequently, the initial dataframe, referred to as the prior Evaluation Dataframe (prior EDF), underwent a meticulous examination, leading to the conclusion that revisions and updates were necessary. To address this issue, new Test Suites were methodically and systematically developed, incorporating representative examples of the typical content handled by the systems. These modifications were guided by meticulous typologies to ensure the highest possible quality. Shortly after the application of this methodology, the annotation typology and evaluation standard were updated, leading to further revisions in the rules. Some rules were too extensive and new rules were added to address client requirements. The most ignored rules were identified, prompting necessary changes and discarding unhelpful rules. The following and final step involved evaluating the revised rules with the updated evaluation standard to ensure accurate error detection.

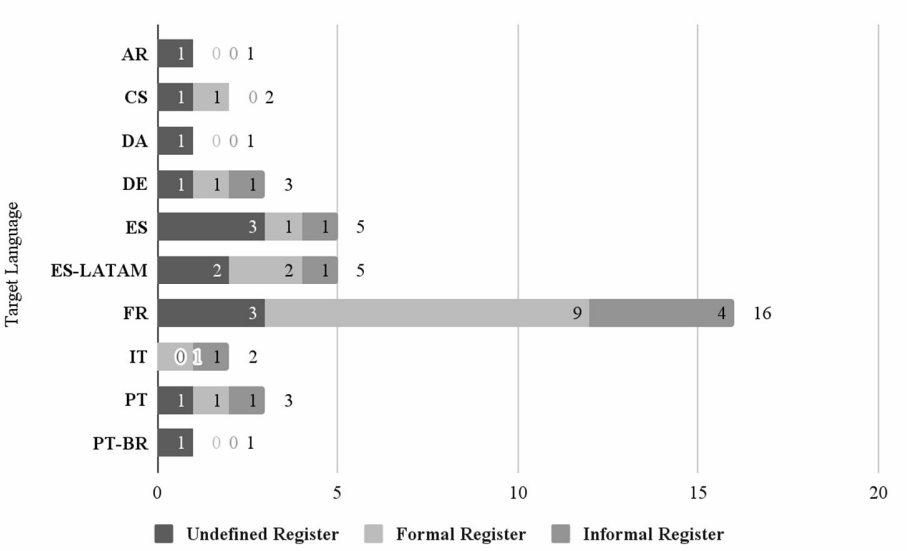
[1] A proprietary programming language that provides a simplified and intuitive interface, enabling linguists to overcome the complexities typically associated with more advanced programming languages.

3.1. **Baseline analysis**

Our research question and motivation stemmed from our initial evaluation, *i.e.* the baseline analysis conducted on custom language rules. This pivotal step involved utilizing quality metrics to assess a total of **39** language-specific rules depicted in Figure 1, all of which were enabled for production by November 5, 2021. This evaluation process prompted us to delve deeper into understanding the effectiveness and impact of these rules, leading us to formulate our research question and pursue further investigation in this area.

Figure 1 provides an overview of the language rules examined during the current baseline analysis. It is important to note that while certain rules were designed for general translations, others were specifically tailored to address formal or informal registers. Furthermore, it is worth highlighting that throughout our analysis, the source language remained consistent as English (EN) across all language pairs examined.

Figure 1. Smartcheck rules: Number of rules per target language



Note. Generic language rules that are not to be applied in any specific register contexts are considered to be in the "Undefined Register" category. Conversely, the "Formal Register" and "Informal Register" categories comprise rules that must only be applied when there is a requirement to use these registers.

The following examples show the difference between each type of rule mentioned above:

(1) **Undefined Register**
Target language: DA; Category: Punctuation
Rule: No punctuation mark must be used after the greeting.

(2) **Formal Register**
Target language: IT; Category: Lexical Register - Formal
Rule: Must use "Cordiali saluti" or "Distinti saluti" in closings.

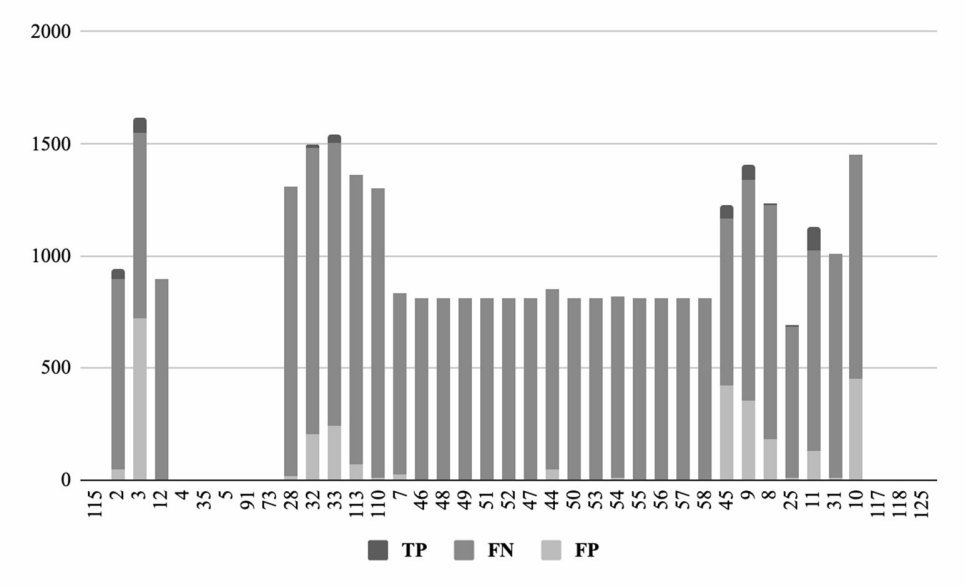
(3) **Informal Register**
Target language: IT; Category: Lexical Register - Informal
Rule: Must use "Ciao", "Saluti" or "Arrivederci" in closings.

It is important to note that our focus in this analysis was on evaluating the individual performance of each rule, rather than considering a set of translated sentences as a whole. Thus, each rule was evaluated with the same set of metrics and the outcome of the evaluation provided us with valuable insights into the frequency with which rules were triggered:

- Cases of True Positives (TPs) – when a rule is correctly triggered due to an existing error in the translation;
- Cases of False Positives (FPs) – when there are no errors in the translation, but a rule is incorrectly triggered nonetheless;
- Cases of False Negatives (FNs) – instances where errors were annotated, but no rule was able to detect and address them.

In order for a rule to achieve high values of accuracy and precision, it is expected that the number of FPs and FNs is minimized in comparison to the number of TPs. However, our analysis revealed a notable disparity, as there were significantly fewer instances of TPs than any other category. This discrepancy is visually depicted in Figure 2, where the y-axis represents the occurrences of rule firings or missed opportunities for firing (corresponding to specific rules identified along the x-axis). This insight highlights the need for further investigation and improvement in order to address the imbalances observed in the performance of the evaluated rules.

Figure 2. Smartcheck rules – TPs, FNs and FPs per rule



The outcomes depicted in Figure 2 provided the necessary data to calculate performance metrics, such as Precision, Recall, and the F1-score, as shown in Figure 3. These metrics are commonly used to assess the effectiveness and overall performance of the rules by offering a holistic assessment of its performance and are calculated in the following manner (Makhoul et al., 1999):

$$Precision = \frac{True\ Positives}{True\ Positives + False\ Positives}$$

$$Recall = \frac{True\ Positives}{True\ Positives + False\ Negatives}$$

$$F1 - score = \frac{2 \times Precision \times Recall}{Precision + Recall}$$

On one hand, Precision must be used in order to quantify the accuracy of an automated system such as Smartcheck. It provides insights into how well Smartcheck performs in terms of identifying and flagging actual categories of errors in translations, as opposed to making false identifications. As such, Precision measures the proportion of positive identifications made by Smartcheck that were genuinely correct. A higher Precision score indicates that the system is more accurate in its error detection, as it is capturing a larger proportion of TP errors among the flagged instances. In summary, Precision is concerned with how accurate Smartcheck positive identifications are, emphasizing the avoidance of FP cases.

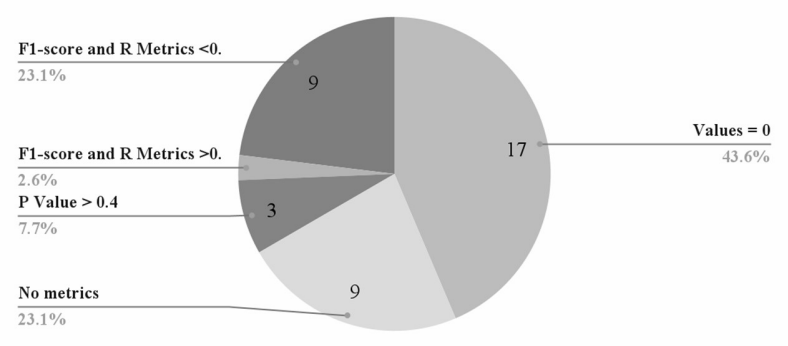
On the other hand, Recall measures the proportion of actual errors within the text that were correctly identified by Smartcheck. It provides insight into how well the system performs in terms of capturing the true errors present in the reference translations. A higher recall score indicates that Smartcheck is more effective at identifying a greater proportion of actual errors within the text, demonstrating its ability to comprehensively detect errors and minimize FP cases. In other words, Recall is concerned with how well Smartcheck captures the actual errors, emphasizing the avoidance of FN cases.

Lastly, the F1-score metric, which is calculated as the harmonic mean between precision (P) and recall (R) as described earlier, represents a balance between these two performance measures.

Regarding Figure 3, it is noteworthy that seventeen rules obtained a score of 0 for all the metrics, including Precision, Recall, and the F1-score. This indicates that these rules did not demonstrate any positive impact or effectiveness in terms of their predictive performance. Additionally, nine other rules experienced a production time out, which prevented the calculation of any metric values. Therefore, the evaluation did not yield any information regarding the performance of these specific rules. Among the evaluated rules, the remaining thirteen rules displayed extremely low values for each metric. None of these rules achieved a score higher than 0.1 for both the F1-score and Recall, except for rule 11, which performed slightly better. Furthermore, only three rules achieved a score higher than 0.4 for Precision.

These results highlight the overall poor performance of the evaluated rules, as evidenced by their consistently low scores across all metrics. The lack of significant positive impact suggests that these rules might not effectively contribute to the desired outcomes or meet the desired standards for overall performance.

Figure 3. Smartcheck rules: Baseline results summary



It is important to note that the observed low results cannot be attributed to the syntax of the rules, as the metrics employed were able to evaluate the rules successfully. The baseline analysis yielded significantly low metric values, indicating a notable performance deficiency. To understand the underlying cause behind Smartcheck rules either failing to flag existing errors in MT outputs or incorrectly flagging correct tokens, a preliminary analysis was conducted to identify potential issues.

The subsequent step involved performing a root-cause analysis to address the results obtained from the previous evaluation. The prior EDF, which serves as the gold standard, should encompass representative examples of the translated content produced by the MT systems and is specifically designed for assessing Error Detection Systems, such as Smartcheck. Smartcheck relies on annotations to provide suggested error corrections to post-editors. As a result, the prior EDF must consist of revised "gold annotations" that require ongoing quality evaluations and frequent updates to ensure their relevance.

To determine the starting point for addressing the issue, instead of individually examining and attempting to enhance the performance of each rule, the focus shifted towards analyzing the data employed for evaluating the rules themselves. Consequently, the evaluation process began with an in-depth analysis of the prior EDF.

3.2. **Root-cause analysis**

The root-cause analysis began with a compilation of data, in which it was required to filter out irrelevant content, and organize the data into distinct files based on specific criteria, namely formal, informal, and generic language-specific data. Due to time constraints only seven language pairs were considered for the analysis.

The data preparation process was conducted in a systematic manner, consisting of two distinct steps. In the first step, the prior EDF was prepared for annotation revision, without any corrections being made at this stage. The second step involved the actual annotation revision process, guided by specific criteria.

The following considerations were taken into account during the data preparation process:

1. Translation step

The focus was exclusively on the MT output, and any post-edited translations were excluded from the analysis. This ensured a clear assessment of the performance of the MT system;

2. Segment duplication
To maintain data integrity and avoid redundant information, duplicated segments were removed from the analysis. This step aimed to streamline the evaluation process and avoid any potential bias resulting from repeated data;

3. Sorting language pair and register
The data was carefully organized based on language pair and register, distinguishing between formal and informal language usage. This allowed for a more targeted evaluation and analysis of the specific linguistic characteristics associated with each language pair and register.

The subsequent step focused on the actual annotation revision process, guided by specific criteria. Annotations were revised according to the MQM-compliant typology for error identification, ensuring a standardized and consistent approach. Additionally, it was crucial to detect incorrect annotations (cases where no errors were present but were erroneously considered as such) as well as missing annotations (instances where errors were present but were not annotated). The severity levels assigned to annotations were also reviewed to determine if they were correctly or incorrectly attributed. Furthermore, the proprietary language guidelines were employed to revise the annotations, aligning them with the desired linguistic standards defined by the language framework.

By following this structured data preparation approach and adhering to the defined criteria, a thorough and reliable assessment of the annotations and their associated linguistic characteristics was achieved and the results are compiled in Table 1. The process ensured data integrity, eliminated duplication, and organized the data based on language pair and register.

Table 1. Prior EDF's analysis: Totals (*Note*: "Ann." stands for "annotations")

Target Language	Total Annotations	Correct Ann.	Incorrect Ann.	Missing Ann.	Correct Severities	Incorrect Severities	Duplicates
DE	1979	1868	111	188	1757	111	312
ES	90	80	10	5	77	3	0
ES-LATAM	1902	1181	721	249	1123	58	189
FR	777	709	68	352	492	217	132
IT	1674	1175	499	244	629	546	458
PT	1303	1183	120	134	553	630	197
PT-BR	930	730	200	326	679	51	534
TOTAL	**8655**	**6926**	**1729**	**1498**	**5310**	**1616**	**1822**

Rather than examining annotations within a dataframe containing duplicated and unverified data, a deliberate choice was made to prioritize the review of more current annotations based on authentic and representative translated content generated by the MT systems. Furthermore, the revision of annotations was deemed imperative, regardless of the selected data. Consequently, it can be deduced that the original dataframe proved inadequate for its intended purpose, highlighting the necessity for an improved evaluation standard.

3.3. **Creation of the New EDF**

A novel and enhanced dataframe was constructed, along with the development of reliable Test Suites. The creation process of the Test Suites followed a similar methodology as presented in Avramidis et al. (2019), Stewart et al. (2022), and Cabeça et al. (2023), encompassing the identification and categorization of errors through

annotations. Contrary to Avramidis et al.'s work, no errors were detected through regular expressions, and there was no requirement for data augmentation. In our work, the errors were meticulously identified through manual annotation, obviating the necessity for regular expressions. The data utilized for this purpose was obtained from our proprietary MT systems, ensuring its inherent representativeness. It is important to emphasize that the new EDF employed in this study does not incorporate gold translations (also known as reference translations), as this aspect falls beyond the scope of our research.

3.3.1. **Data curation**

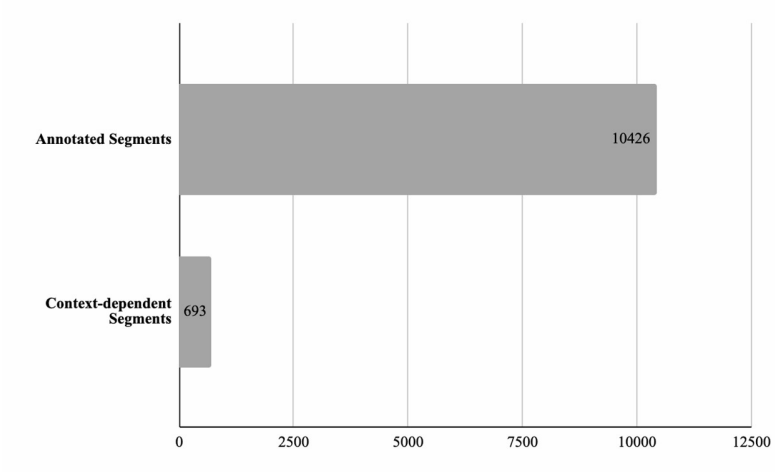
The creation of the new EDF followed a meticulous process, which encompassed the curation of data and the extraction of recently annotated segments. The data utilized for constructing the Test Suites spanned from January 1, 2021, to February 28, 2022, and was meticulously filtered using a methodology akin to the aforementioned filtering step. This data curation step served a practical purpose, as the manual review of annotations necessitates a higher level of effort and data control.

Consistent with all language pairs, duplicates were identified and subsequently removed to ensure the integrity of the dataset. Notably, this data curation step addressed a significant limitation present in the previous dataframe. By scrutinizing the data at a more granular level, it became possible to identify context-dependent annotations that require additional information beyond the individual segment. While the current SURF rules do not explicitly account for context-dependent errors, incorporating this supplementary curation step was imperative. It will enable the new Test Suites to encompass such context-dependent annotations once the SURF rules are updated to accommodate them in future iterations.

As such, during the analysis conducted, three distinct types of context-dependent annotations were identified. The first type involved inconsistencies (translation errors annotated as *Inconsistency*), which are susceptible to nearby segments within the same text. The second type pertained to *Capitalization*, exclusively observed in greetings and closings, dictated by the language specifications for tickets, which are composed in a manner akin to traditional letters. Tickets require distinct guidelines pertaining to capitalization and punctuation, particularly in greetings and closings, highlighting language-specific variations. The third type comprised agreement-related annotations (annotated as *Agreement*), with the referent located outside the segment.

As depicted in Figure 4, among the 10,426 annotations that were thoroughly examined, a mere 6.6% of them, corresponding to a total of 693 segments, were determined to exhibit context-dependent characteristics. This relatively small subset of annotations demonstrated a dependency on surrounding segments or required additional contextual information to accurately evaluate and interpret the linguistic errors present.

Figure 4. New EDF's content: Number of annotated segments and context-dependent segments



3.3.2. **Annotation curation**

The annotation curation process involved a meticulous revision of the previously annotated segments, guided by proprietary language and annotation guidelines. A specific focus was placed on addressing certain types of errors, with priority given to *Orthography*, *Punctuation*, *Register*, and localization challenges such as *Date/time format*, *Currency*, and numeral-related errors. Additionally, particular attention was devoted to selected language pairs, namely DE, ES, ES-LATAM, FR, IT, JA, KO, PT, PT-BR, ZH-CN, and ZH-TW, where linguists and translators carried out more comprehensive reviews. This approach aimed to establish a consensus among the experts involved and ensure consistency throughout the revision process.

The annotation curation step consisted of two distinct assessments, progressively delving into finer details:

- **General Assessment**: This initial phase involved examining the comprehension of the translated text and adhering to language-specific requirements. It entailed scrutinizing aspects related to accuracy, fluency, typography, style, and localization. Errors such as *Additions*, *Omissions*, *Untranslated* content, *MT hallucinations*, *Mistranslations*, *Duplications*, grammar-related issues, and formatting inconsistencies were carefully evaluated within the context of the translated text.
- **Assessment of Customer Support Rules**: This subsequent phase focused specifically on Customer-Service related content. The assessment concentrated on issues pertaining to *Lexical Register*, *Punctuation*, and *Capitalization*. Understanding and implementing appropriate lexical choices, adhering to specific punctuation conventions in greetings and closings, and ensuring correct capitalization based on the sentence structure and part of speech category were key considerations in this evaluation.

3.3.3. **New EDF content**

Through these comprehensive assessments, the new EDF underwent a thorough curation process, resulting in a completed and refined dataset that met the standards set forth for further analysis and evaluation. Thus, the new

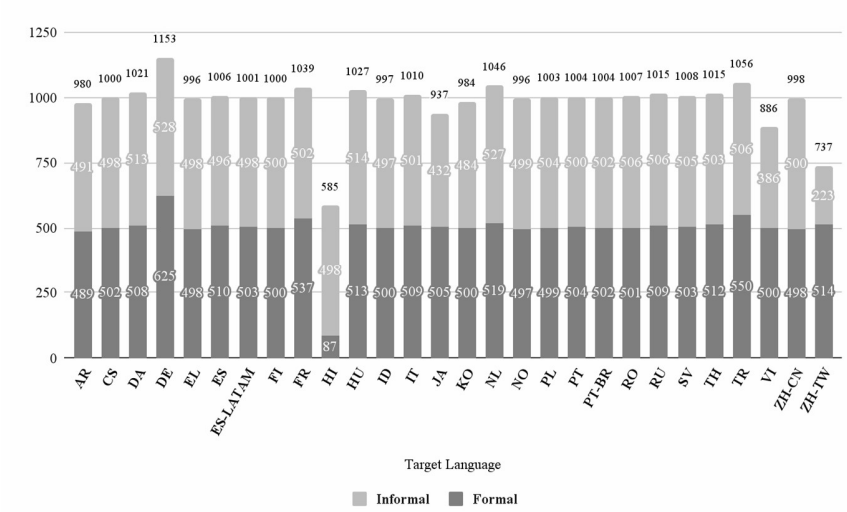
EDF emerged as a comprehensive Test Suite suitable for evaluating Smartcheck rules. The new EDF comprised a total of nearly thirty thousand segments, as depicted in Table 2.

Table 2. New EDF's content: Grand total of formal and informal segments

	Nº of segments
Formal Segments	13894
Informal Segments	13617
TOTAL	27511

These segments encompassed translations for twenty-eight distinct target languages, as illustrated in Figure 5. While it is worth noting that the data for the formal register in HI exhibits a considerably lower number of segments compared to the informal register, the vast majority of languages in the evaluation exhibit a well-balanced distribution of segments. This ensures that the Test Suite consists in a representative sample for most supported languages, enabling a robust evaluation. The focus on maintaining balance in segment counts across languages strengthens the reliability and generalizability of the evaluation results.

Figure 5. EDF's content: Total of formal and informal segments per target language



3.4. **Smartcheck's Performance Optimization**

Each successive stage outlined in this methodology has been dedicated to establishing an appropriate evaluation process to gauge the efficacy of Smartcheck and its error detection rules. Consequently, it is now feasible to conduct a comprehensive assessment of this tool's accuracy and confidently rely on the evaluation results to identify its limitations.

Upon the conclusion of the aforementioned evaluation process, the annotation typology underwent an update to a new version, requiring the adjustment of not only the evaluation standard - the new EDF - but also the rules

themselves. This requirement arises from the fact that Smartcheck relies on annotations to provide suggestions to post-editors. During the revision and updating of the rules, the following was observed: not only were certain rules overly intricate, comprising multiple sub-rules; but additionally, new content-specific rules and rules tailored to specific client requirements needed to be incorporated. In light of this, a thorough analysis was conducted to identify the rules that were most frequently disregarded by the post-editors. Subsequently, the rules that were deemed unhelpful in terms of their limited accuracy in detecting errors were deliberately disabled from Smartcheck.

Consequently, the subsequent step focused on the evaluation of these revised rules, now utilizing the updated EDF, so as to ensure their precise detection of errors.

4. **Results**

The results obtained from the baseline analysis revealed Smartcheck's low performance. The high number of FNs and FPs indicated that Smartcheck was unable to effectively identify the majority of translation issues, and even worse, it flagged problems that did not exist. This led us to formulate a hypothesis that the quality of the data used to evaluate Smartcheck's rules might be a contributing factor to this problem. To address this concern, we made the decision to replace the previous EDF with a new set of gold annotations. The substitution of the EDF with linguist-curated gold annotations represented a notable improvement in the evaluation methodology. The new set of annotations provided a more robust and reliable benchmark for assessing the performance of Smartcheck's rules across all supported language pairs.

4.1. **Rule evaluation comparison between prior EDF and new EDF**

Upon the implementation of the new EDF, it became essential to verify the status of the 39 previously evaluated rules to ensure that they were still enabled in the production environment. This verification was necessary as the baseline analysis had been conducted several months prior to the introduction of the new EDF.

After completing the verification process, Smartcheck was run through numerous MT outputs, and the corresponding metric values were collected and compared. To illustrate this process, let us consider the following example: during the baseline analysis, an issue was identified concerning a punctuation mark in greetings for a certain target language. To automatically detect this language-specific translation problem, Smartcheck referred to Rule A, which specifically addresses punctuation requirements in greetings for that same target language, triggering a warning. It is important to note that Rule A was among the rules evaluated during the baseline analysis.

Subsequently, with the implementation of the new EDF, Smartcheck was once again applied to new MT outputs, and a similar punctuation issue in greetings for the same target language was encountered. Consequently, both sets of MT outputs exhibited the same issues, leading to the activation of the same rule (Rule A) in both instances. If such conditions were met for other existing rules, it would allow for a direct comparison of results between them. In Table 3, we present **7** rules out of the 39 previously evaluated that fulfilled this criterion. Hence, for an unambiguous comparison of rule evaluations, only these seven rules will be considered.

Furthermore, it is important to emphasize that the baseline comparison will focus exclusively on cases of TPs, FNs, and FPs.

Table 3. Rule's comparison between baseline analysis and the new EDF

		Prior EDF			New EDF		
Target Language	**Rule ID**	**TPs**	**FNs**	**FPs**	**TPs**	**FNs**	**FPs**
AR	117		No metrics		4	8	2
CS	125		No metrics		0	40	2
DA	115		No metrics		2	12	1
DE	12	0	0	899	0	21	1
ES-LATAM	28	0	16	1297	3	2	6
	110	3	9	1294	45	8	0
FR	45	59	420	750	123	67	1

Upon comparing the results from both evaluations, two significant inferences were drawn. Firstly, a disparity was observed in terms of FPs between the two evaluations. Secondly, it became possible to evaluate rules that were previously not assessable.

In the baseline analysis, there was a considerable disparity in the number of TPs, FNs, and FPs between the two evaluations. The high number of FP cases in the baseline analysis had a detrimental effect on the system's performance. However, the implementation of the new EDF resulted in a significant reduction in the number of FPs. FN cases also decreased, except for rule 12 in the case of English to German. Additionally, the number of TPs substantially increased for rules 110 and 45. Therefore, it can be concluded that the Smartcheck rules do not possess the low quality that was previously assumed. In other words, solely by modifying the evaluation standard - specifically, the EDF - without altering the rules themselves, the rules used by Smartcheck for error detection exhibit improved performance, albeit not perfect.

On the other hand, for the initial three rules listed in Table 3, no results were available to ascertain their precision, coverage, or overall value for a tool like Smartcheck, as no metrics were obtained. However, due to the introduction of the new EDF, every rule in Smartcheck can now undergo evaluation. This step was crucial in the quality assurance process, as improvements in machine translation systems can only be achieved when existing issues are accurately identified and the necessary adjustments are implemented. Understanding whether a rule fails to trigger when it should or if it incorrectly identifies high-quality translations as problematic is of utmost importance, as it provides guidance for rule revision during the review process.

Nonetheless, drawing conclusions from the evaluation results calls for a certain level of trustworthiness. Since the current EDF comprises non-duplicated data, representative examples that align with the content translated by proprietary MT systems, and annotations reviewed by linguists, it can be inferred that the new evaluation standard and its corresponding results are substantially more reliable and accurate compared to the previous approach.

4.2. **Smartcheck's performance evaluation with new EDF**

The following step in the methodology involved an examination of a correlation pertaining to Smartcheck predictions. Smartcheck's annotations were evaluated in comparison to those of the EDF, while keeping the Smartcheck rules unchanged. Fundamentally, one error detection system would be supplied with the pre-existing data from Smartcheck, while a second error detection system would make use of the reviewed data from the EDF. Subsequently, each system would generate predictions for the same translated segments, identifying tokens deemed erroneous by the system and annotating them accordingly.

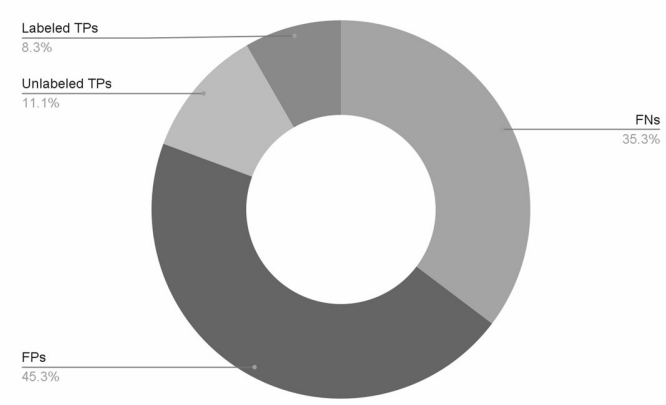
Hence, our objective was to ascertain the disparity between the annotations produced by this grammar-checking tool and a novel, meticulously curated standard. To achieve this, both Smartcheck and the EDF underwent

analysis on various translated segments, and the outcomes were combined and categorized into three distinct sections: performance assessment involving instances of TPs, FPs, and FNs; a comparison of predicted annotations between Smartcheck and the EDF; and a recalculation of metric values associated with Precision, Recall, and the F1-score.

4.3. **Performance assessment**

After running Smartcheck and the EDF on translated segments, the corresponding data was collected. The evaluation focused on cases of FPs, FNs, and TPs. Regarding TPs, they were categorized into two groups: labeled TPs, wherein both the error span and category were correctly identified according to the gold standard, and unlabeled TPs, wherein the error span was correctly identified, yet Smartcheck attributed an error category that did not align with the "gold category." Examining the aggregate count of TPs, FNs, and FPs in Figure 6 allows for the conclusion that Smartcheck considered numerous correct tokens as incorrect, resulting in FP cases. A high prevalence of FPs is detrimental to any error detection system and should be minimized, as it leads to the erroneous perception of statistical evidence that is non-existent. Furthermore, over a third of all annotations were FN cases, indicating that the system overlooked a considerable number of existing translation errors. Nevertheless, FN cases do not have as severe an impact on system performance as FPs, since we have the ability to create new grammar checking rules or review existing ones to reduce the occurrence of FN cases.

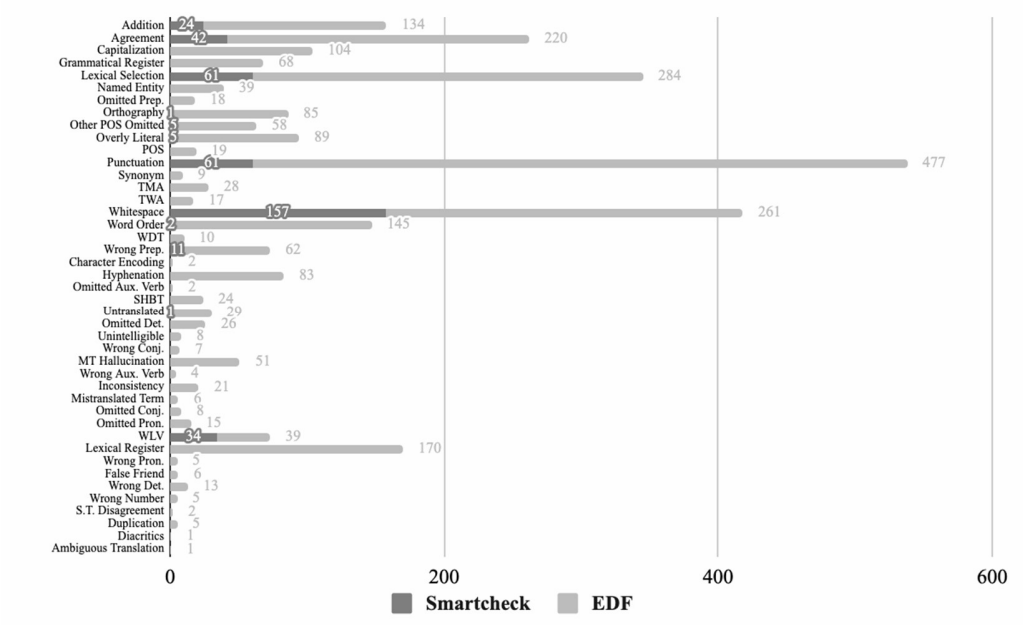
Figure 6. Grand Total of FNs, FPs and TPs when evaluating Smartcheck with the new EDF



4.3.1. **Predicted annotation comparison**

During the comparative analysis, annotations generated by Smartcheck were contrasted with those produced by the EDF using the same dataset from the preceding section. It is important to note that specific target languages and registers were not taken into consideration for the current analysis. Figure 7 presents an overview of the EDF's predicted ideal annotations in comparison to Smartcheck's correctly detected annotations, specifically labeled TP cases. As depicted in Figure 7, Smartcheck failed to achieve the expected objective for all forty-three detected error types in the EDF. The annotations for *Wrong Language Variety*, for instance, came closest to the target. However, for the remaining error types, Smartcheck's annotations significantly deviated from the target count.

Figure 7. Smartcheck labeled TP cases in comparison to gold annotations from the EDF



An analysis of this nature proves highly valuable when evaluating the performance of an error detection system, as it offers a comprehensive understanding of the most problematic error types and allows for the identification of concealed issues that would otherwise go unnoticed.

4.3.2. **Smartcheck evaluation**

The concluding phase of the Smartcheck testing process entailed the computation of Precision, Recall, and F1-score measures, based on the outcomes of the preceding section, specifically the TP, FN, and FP cases. Accordingly, these metrics were individually calculated for each supported target language, with a particular emphasis on discerning the disparity between formal and informal registers.

In order to determine the accuracy of positive identifications made by Smartcheck, the Precision measure was employed. Once computed, the average Precision value for the formal register was slightly higher compared to the average value for the informal register, as evidenced in Table 4.

Table 4. P Average Total per register

Register	Precision Average
Formal	0.3035
Informal	0.3024

To assess the effectiveness of Smartcheck in correctly identifying actual positives based on the new EDF, the Recall measure was employed. The average Recall value for the formal register significantly exceeded that of the informal register, as demonstrated in Table 5.

Table 5. R Average Total per register

Register	Recall Average
Formal	0.2963
Informal	0.2477

The F1-score metric, which represents the harmonic mean of Precision and Recall, was employed to statistically evaluate the performance of the system. As indicated in Table 6, the formal register exhibited a higher average F1-score compared to the informal register.

Table 6. F1-score Average Total per register

Register	F1-score Average
Formal	0.2814
Informal	0.2636

4.4. **Evaluation of spell checkers with new EDF**

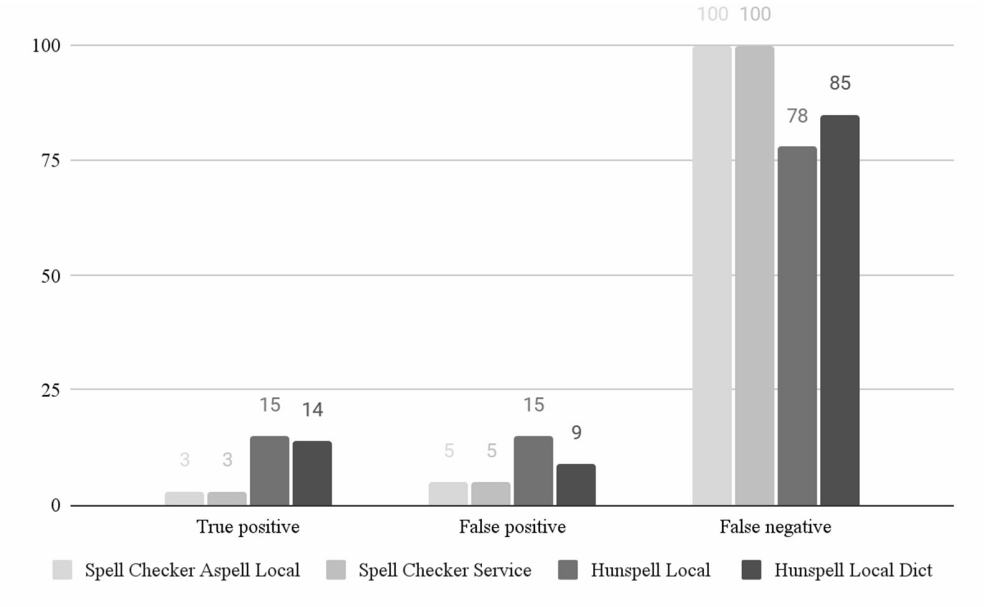
As previously stated, the EDF served as a reliable source of annotated error data in MT outputs, making it the designated gold standard for evaluating error detection systems. In light of this, the EDF's gold data was recently incorporated to assess four different spell checkers used in the production environment with the objective of determining the most suitable candidate for production deployment.

Referring to the data presented in Figure 8, it becomes evident that FN cases exhibited the highest numbers, particularly in relation to the first two spell checkers. This evaluation highlights the significant oversight of existing translation errors (according to the EDF's data) by these spell checkers. Furthermore, the limited number of TP cases indicates a lack of accuracy in error identification. One notable conclusion drawn from the evaluation was that both Hunspell spell checkers outperformed Aspell and the Spell Checker Service. For instance, the Hunspell[2] spell checker exhibited higher TP counts, lower FN totals, and notably higher values for Precision, Recall, and subsequently, F1-score when compared to the others. Based on this assessment, the decision to replace Aspell[3] with

[2] An open source spell checker accessible at http://hunspell.github.io/
[3] An open source spell checker accessible at http://aspell.net/

Hunspell as the spell checker in production was deemed beneficial. Such a decision would not have been possible without the revised and truthful evaluations made possible by the revised EDF.

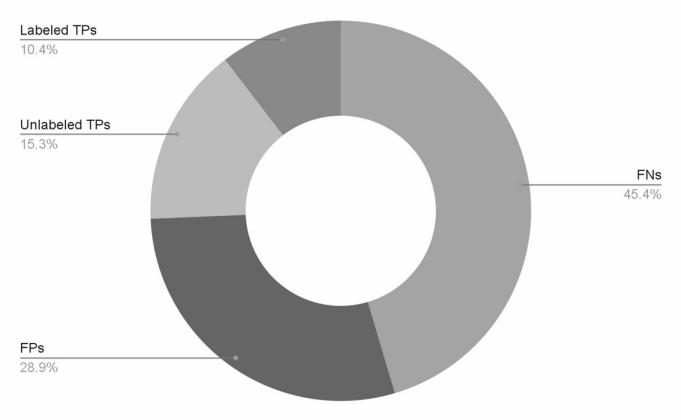
Figure 8. Spell Checkers Evaluation with new EDF as Gold Standard: Cases of TPs, FPs and FNs



4.5. **Smartcheck's Performance Optimization Results**

Upon the requisite modifications implemented in the rules, a subsequent evaluation was conducted to assess their performance, employing the new EDF as the updated evaluation standard once again.

Figure 9. Smartcheck's second evaluation results with the new EDF



Upon careful examination of the final evaluation results, it can be deduced that the outcomes are promising, indicating notable advancements in the functionality of Smartcheck, as depicted in Figure 9. This conclusion is substantiated by the following observations in comparison to the previous evaluation (refer back to Figure 6): Firstly, the total number of TPs has increased from 19.4% to 25.7%, denoting a positive trend in the system's accuracy. Secondly, there has been a significant reduction in the number of FPs, decreasing from 45.3% to 28.9%, representing a notable improvement of 16.4%. Although there has been a rise in the number of FNs from 35.3% to 45.4%, it is important to note that a higher percentage of FNs is preferable over FPs. This preference stems from the fact that it is more desirable for the editor to identify errors independently rather than constantly having to dismiss multiple incorrect suggestions by clicking on the "Ignore" option. Nevertheless, our objective remains to minimize the number of FNs and ultimately eliminate any instances of FPs.

These findings underscore the positive impact of the ongoing efforts to optimize Smartcheck, leading to a more refined and effective tool.

5. **Conclusions**

This paper presents a comprehensive approach to improve the evaluation methodology for error detection systems, with a focus on Smartcheck. We address the limitations of the previous evaluation data set, which failed to capture core errors and led to unreliable results and conclusions. By implementing new Test Suites, we demonstrate the increased trustworthiness of the decision-making process and the ability to evaluate Smartcheck.

Our work contributes to the replicability and visibility of the methodological process involved in creating and curating Test Suites. The main objective was to answer the research question of whether the dataframe used for evaluating Smartcheck rules was suitable for its intended purpose. We propose and implement a robust methodology for creating reliable Test Suites specifically tailored for testing Smartcheck and evaluating custom language rules.

Through this process, we provide valuable insights that contribute to improving translation quality by suggesting edits to post-editors with Smartcheck.

Furthermore, by utilizing the newly developed EDF, we extend the application of the Test Suite beyond its original scope to include the evaluation of spell checkers. This expansion demonstrates the importance of linguistically motivated and scalable Test Suites that can accommodate diverse evaluation objectives.

Additionally, our methodology enables an error-specific evaluation of Smartcheck, striking a good balance between manual and automated metrics. This approach provides a historical perspective on the performance of models, enabling the identification of features that may have been overshadowed by minor improvements when evaluating overall quality. Through these insights, we highlight the significance of small features that can have a substantial impact on the performance and effectiveness of error detection systems.

In conclusion, our research contributes to the advancement of evaluation methodology, ensuring more reliable and valid results for error detection systems. The implementation of our approach demonstrates the importance of carefully curated Test Suites and their role in facilitating informed decision-making processes and improving the overall quality.

References

Avramidis, Eleftherios, Vivien Macketanz, Ursula Strohriegel & Hans Uszkoreit. (2019) Linguistic evaluation of German-English machine translation using a test suite. In *Proceedings of the Fourth Conference on Machine Translation* (Vol. 2). Association for Computational Linguistics, pp. 445–454. https://doi.org/10.18653/v1/W19-5351

Avramidis, Eleftherios, Vivien Macketanz, Arle Lommel & Hans Uszkoreit (2018) Fine-grained evaluation of quality estimation for machine translation based on a linguistically motivated test suite. In *Proceedings of the AMTA 2018 Workshop on Translation Quality Estimation and Automatic Post-Editing*. Association for Machine Translation in the Americas, pp. 243–248. Available at https://aclanthology.org/W18-2107

Balkan, Lorna (1994). Test Suites: some issues in their use and design. In *Proceedings of the Second International Conference on Machine Translation: Ten years on*. Available at https://aclanthology.org/1994.bcs-1.24

Balkan, Lorna, Doug Arnold & Siety Meije (1994) Test suites for natural language processing. In *Proceedings of Translating and the Computer 16*. Aslib, pp. 51–58. Available at https://aclanthology.org/1994.tc-1.5

Cabeça, Mariana, Marianna Buchicchio, Madalena Gonçalves, Christine Maroti, João Godinho, Pedro Coelho, Helena Moniz & Alon Lavie (2023) Quality fit for purpose: Building business critical errors test suites. In *Proceedings of the 24th Annual Conference of the European Association for Machine Translation*. European Association for Machine Translation, pp. 451–460. Available at https://aclanthology.org/2023.eamt-1.44

Dale, Robert, Ilya Anisimoff & George Narroway (2012) HOO 2012: A report on the preposition and determiner error correction shared task. In *Proceedings of the Seventh Workshop on Building Educational Applications Using NLP*. Association for Computational Linguistics, pp. 54–62. Available at https://aclanthology.org/W12-2006

King, Margaret & Kirsten Falkedal (1990) Using test suites in evaluation of machine translation systems. In *COLING 1990: Papers presented to the 13th International Conference on Computational Linguistics* (Vol. 2). pp. 212–216. Available at https://aclanthology.org/C90-2037

Lavie, Alon, & Michael J. Denkowski (2009) The Meteor metric for automatic evaluation of machine translation. *Machine Translation* 23, pp. 105–115. https://doi.org/10.1007/s10590-009-9059-4

Lommel, Arle, Hans Uszkoreit & Aljoscha Burchardt (2014) Multidimensional Quality Metrics (MQM): A framework for declaring and describing translation quality metrics. *Tradumàtica: Tecnologies de La Traducció* (12), pp. 455–463. https://doi.org/10.5565/rev/tradumatica.77

Makhoul, John, Francis Kubala, Richard Schwartz & Ralph Weischedel (1999) Performance measures for information extraction. In *Proceedings of DARPA Broadcast News Workshop*. DARPA, pp. 249–252

Papineni, Kishore, Salim Roukos, Todd Ward & Wei-Jing Zhu (2002) BLEU: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting of the Association for Computational Linguistics*. Association for Computational Linguistics, pp. 311–318. https://doi.org/10.3115/1073083.1073135

Rei, Ricardo, Craig Stewart, Ana C Farinha & Alon Lavie (2020) COMET: A Neural Framework for MT Evaluation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Association for Computational Linguistics, pp. 2685–2702. https://doi.org/10.18653/v1/2020.emnlp-main.213

Stewart, Craig, Madalena Gonçalves, Marianna Buchicchio & Alon Lavie (2022) Business critical errors: A framework for adaptive quality feedback. In *Proceedings of the 15th Biennial Conference of the Association for Machine Translation in the Americas* (Vol. 2). Association for Machine Translation in the Americas, pp. 231–256. Available at https://aclanthology.org/2022.amta-upg.17