Cross-Lingual Text Reuse Detection at Document Level for English-Urdu Language Pair

MUHAMMAD SHARJEEL, Department of Computer Science, COMSATS University Islamabad, Lahore Campus, Pakistan., Pakistan
IQRA MUNEER, University of Engineering & Technology Lahore, Narowal Campus, Pakistan, Pakistan
SUMAIRA NOSHEEN, Bahria University, Lahore Campus, Lahore, Pakistan, Pakistan
RAO MUHAMMAD ADEEL NAWAB, Department of Computer Science, COMSATS University Islamabad, Lahore Campus, Lahore, Pakistan
PAUL RAYSON, Lancaster University, Lancaster, United Kingdom, United Kingdom

In recent years, the problem of Cross-Lingual Text Reuse Detection (CLTRD) has gained the interest of the research community due to the availability of large digital repositories and automatic Machine Translation (MT) systems. These systems are readily available and openly accessible, which makes it easier to reuse text across languages but hard to detect. In previous studies, different corpora and methods have been developed for CLTRD at the sentence/passage level for the English-Urdu language pair. However, there is a lack of large standard corpora and methods for CLTRD for the English-Urdu language pair at the document level. To overcome this limitation, the significant contribution of this study is the development of a large benchmark cross-lingual (English-Urdu) text reuse corpus, called the TREU (Text Reuse for English-Urdu) corpus. It contains English to Urdu real cases of text reuse at the document level. The corpus is manually labelled into three categories (Wholly Derived = 672, Partially Derived = 888, and Non Derived = 697) with the source text in English and the derived text in the Urdu language. Another contribution of this study is the evaluation of the TREU corpus using a diversified range of methods to show its usefulness and how it can be utilized in the development of automatic methods for measuring cross-lingual (English-Urdu) text reuse at the document level. The best evaluation results, for both binary (F_1 = 0.78) and ternary (F_1 = 0.66) classification tasks, are obtained using a combination of all Translation plus Mono-lingual Analysis (T+MA) based methods. The TREU corpus is publicly available to promote CLTRD research in an under-resourced language, i.e. Urdu.

CCS Concepts: • **Computing methodologies** → **Language resources**;

Additional Key Words and Phrases: Cross-Lingual Text Reuse, Cross-Lingual Text Reuse Detection, English-Urdu Language Pair, Cross-Lingual Sentence Embedding, Translation plus Mono-lingual Analysis

Authors' addresses: Muhammad Sharjeel, muhammadsharjeel@cuilahore.edu.pk, Department of Computer Science, COMSATS University Islamabad, Lahore Campus, Pakistan., Comsats University Islamabad, Lahore Campus, Pakistan, Lahore, Pakistan, 54000; Iqra Muneer, iqramuneer@uet.edu.pk, University of Engineering & Technology Lahore, Narowal Campus, Pakistan, University of Engineering & Technology Lahore, Narowal Campus, Pakistan, Lahore, Pakistan, 54000; Sumaira Nosheen, summaira.bulc@bahria.edu.pk, Bahria University, Lahore Campus, Lahore, Pakistan, Bahria University, Lahore Campus, Lahore, Pakistan, Lahore, Pakistan, 54000; Rao Muhammad Adeel Nawab, adeelnawab@cuilahore.edu.pk, Department of Computer Science, COMSATS University Islamabad, Lahore Campus, Lahore, Comsats University Islamabad, Lahore Campus, Lahore, Pakistan, 54000; Paul Rayson, p.rayson@lancaster.ac.uk, Lancaster University, Lancaster, United Kingdom, Lancaster University, Lancaster, United Kingdom.

2375-4699/2023/5-ART $15.00
https://doi.org/10.1145/3592761

1 INTRODUCTION

Cross-Lingual Text Reuse (CLTR) is the process of creating new text by borrowing text(s) from a different language. The amount of borrowed text varies from small phrases, sentences, and paragraphs, to entire documents. Moreover, CLTR often implies different levels of rewriting as it starts from verbatim (simple translation), stretches to paraphrasing (after translation, contents are rephrased using different text editing operations), to cases where the re-written text is produced completely independent of its source text.

Recent studies suggest that CLTR is on the rise for a number of reasons, the transformation of the Web into a social and multi-lingual hub, the expansion of Wikipedia in multiple languages with readily available electronic documents, and the widely adopted use of Machine Translation (MT) systems [20, 27]. Consequently, the computational study and thorough analysis of CLTR is becoming a hot research topic. Besides, developing reliable systems for the detection of CLTR has become an interesting intellectual problem and one whose solution promises practical benefits to both individuals and organizations. Additionally, Cross-Lingual Text Reuse Detection (CLTRD) has numerous applications in other fields, e.g., cross-lingual information retrieval, cross-lingual plagiarism detection, and cross-lingual question answering [19].

Although many studies have targeted CLTRD, the majority of the previous efforts were inclined towards English-Arabic, English-Persian, or English-European language pairs. However, there is a large population of the world that speak Indo-Aryan languages (approximately one billion) and there is a clear shortage of corpora and methods proposed for the CLTRD research on these languages. Urdu, belonging to the Indo-Aryan language family, is the official language of Pakistan and is predominantly spoken in the country. Moreover, it is one of the most popular languages spoken by around 175 million people around the globe. In contrast to English, Urdu is conventionally written right-to-left in Nastaliq style and relies heavily on Arabic and Persian sources for literary and technical vocabulary. However, it is a low-resource language concerning even core processing tasks such as tokenization, Part-of-Speech (PoS) tagging, or morphological analysis. Moreover, there is a clear shortage of corpora and methods available for text reuse and extrinsic plagiarism detection research in the Urdu language.

CLTR can occur at different levels of granularity including lexical, syntactical, phrasal, sentence, passage, and document levels. CLTRD methods show different performances at different levels of rewrite [33] including lexical, syntactical, and phrasal levels. Similarly, as can also be seen from the literature, CLEU-Sen (English-Urdu) corpus at the sentence level [32], and CLEU (English-Urdu) corpus at the sentence/passage level [34] show different performances. E.g. a T+MA baseline method shows F_1 score (0.711, 0.486) and (0.890, 0.724) with N-gram-comb for CLEU [34], and CLEU-Sen [32] for both binary, and ternary classification task respectively. Documents are a composite of sentences and passages and performance effects when varying granularity will be different. Consequently, it becomes more challenging to capture the semantic similarities when the length of the text increases especially at the document level. Similarly, USTRD [38] corpus at sentence/passage level, and COUNTER [40] at document level have been developed for the task of text reuse detection for the Urdu language. In addition, MRPC [17] corpus for sentence level, and METER [12] corpus at document level have been developed for the task of text reuse detection for the English language. To summarize, different benchmark corpora including lexical, syntactical, and phrasal [33] sentences/passages [32, 34] have been developed for the task of CLTRD for English-Urdu language pair. However, there is no single gold standard benchmark at the document level. To overcome this limitation, there is a dire need to develop a CLTRD corpus at the document level for the English-Urdu language pair.

To address this shortcoming, we present efforts on developing a benchmark evaluation corpus and methods to detect CLTR in the English-Urdu language pair. We believe that this is the first work that thoroughly explores this problem in the cross-lingual context for the English-Urdu language pair. The contributions of this study are as follows.

1) We present a large-scale CLTR corpus for the English-Urdu Language pair. The TREU (Text Reuse English Urdu) Corpus is developed in the footsteps of COUNTER [40] and METER [12] corpora, i.e., compiling data from journalism. The corpus is comprised of cross-lingual English-Urdu real cases of text reuse at the document level. The source text documents in the corpus are in the English language while the derived text documents are in the Urdu language. For source text documents, the English news reports released by the news agencies are used. The derived text documents, on the other hand, are Urdu newspaper stories published in the popular Urdu newspapers of Pakistan. Each of the news agency reports (English text) has a one-to-one mapping with the newspaper story (Urdu text), but as practised in journalism, the newspaper story may or may not contain text from the news agency report. The TREU Corpus contains a total of 2,257 source-derived text document pairs (a total of 4,514 text documents). These pairs are divided into three categories, i.e. (1) Wholly Derived (WD), when the derived text document is the mere translation (with small changes due to language structure) of the source text document (verbatim copy), (2) Partially Derived (PD), when the derived text document is the paraphrased version of the translated source text document, and, (3) Non Derived (ND) when the text document is independently written without referring to the source text document.

2) To the best of our knowledge, this is the first study that has applied a diverse range of CLTRD methods on an English-Urdu cross-lingual text reuse corpus at the document level. The applied methods provide in-depth analysis and set a strong baseline for the CLTRD task in a low-resource language pair, i.e., English-Urdu. Furthermore, these methods could easily be extended to other similar language pairs (e.g., English-Arabic, English-Persian, etc.) for the CLTRD task. The applied methods are based on (T+MA) and broadly categorized into five types, i.e., lexical overlap, string matching, structural similarity, mono-lingual word embedding, and mono-lingual sentence embedding.

This study holds both theoretical and practical significance. As far as we are aware, the TREU corpus is the first of its kind cross-script cross-lingual standard evaluation resource developed for CLTRD research for the English-Urdu language pair. We believe that the corpus will serve as a benchmark for the evaluation of the state-of-the-art CLTRD methods in general, and more specifically, for the English-Urdu (or similar) language pair. Moreover, it can also facilitate the development of algorithms that can detect cross-script cross-lingual text reuse at the document level.

The rest of this paper is organized as follows: Section 2 discusses existing corpora and methods for CLTRD. Section 3 presents the corpus generation process used to create the cross-lingual corpus. Section 4 describes the proposed techniques for CLTRD. Section 5 describes the experimental setup. Section 6 presents results and their analysis. Finally, Section 7 concludes the paper with future research directions.

2 RELATED WORK

In the previous literature, the majority of efforts have been to create various methods and corpora for estimating CLPD and CLTRD.

Recently, in another study, three large gold-standard cross-lingual text reuse detection corpora have been developed for the task of CLTRD along with cross-lingual methods for the English-Urdu language pair by Muneer et al. [33]. The proposed cross-lingual corpora include CLEU-Lex, CLEU-Syn, and CLEU-Phr at the lexical, syntactical, and phrasal levels for the English-Urdu Language Pair.

The CLEU-Lex contains 66,485 pairs with the source text in English and reused text in Urdu based on simulated cases at the lexical level. The pairs were manually labeled into three classes (Wholly Derived = 22,236, Partially Derived = 20,315, Non Derived = 23,934) [33]. Three different methods including baseline (Bi-lingual Dictionary), and proposed (Cross-lingual Semantic Tagger, CL-WE, and CL-ST). The best results were obtained with F_1 score of (0.69, 0.80) for CLEU-Lex [33] for the ternary and binary classification tasks respectively.

The other proposed gold standard bench-mark ’CLEU-Syn’ corpus contains 60,267 pairs with the source text in English and reused text in Urdu based on simulated cases at the syntactical level. The pairs were manually labeled into three classes (Wholly Derived = 20,007, Partially Derived = 16,979, Non Derived = 23,281) [33]. Three different methods including baseline (Bi-lingual Dictionary), and proposed (Cross-lingual Semantic Tagger, CL-WE, and CL-ST). The best results were obtained with F_1 score of (0.82, 0.92) for CLEU-Syn [33] for the ternary and binary classification tasks respectively.

The third gold standard bench-mark corpus named ’CLEU-Phr’ contains 60,106 cross-lingual pairs with the source text in English and reused text in Urdu based on simulated cases at the phrasal level. The CLTR pairs were again manually labeled into three classes (Wholly Derived = 23,862, Partially Derived = 15,878, Non Derived = 20,366) [33]. Three different methods including baseline (Bi-lingual Dictionary), and proposed (Cross-lingual Semantic Tagger, CL-WE, and CL-ST). The best results were obtained with F_1 score of (0.78, 0.94) for CLEU-Phr [33] for the ternary and binary classification tasks respectively.

Recently, Muneer et al [32] presented a corpus along with a variety of approaches for Cross-lingual text reuse detection. The proposed cross-lingual corpus consists of 21,669 English-Urdu pairs at the sentence level based on simulated data. The corpus is manually annotated into three categories as (Wholly Derived = 7,655, Partially Derived = 6,461, Non Derived = 7,553) with source in English and derived in Urdu languages. The authors applied Translation + Monolingual analysis (T+MA) approaches, Cross-lingual sentence transformers (CLST) approaches, and combinations of these approaches were also applied for Cross-lingual Text Reuse Detection (CLTRD) for binary and ternary classification. The best results obtained were F_1 of 0.94 for binary with a combination of all CLST, T+MA approaches along with all combined T+MA approaches. Furthermore, the best results obtained were F_1 of 0.84 for ternary classification using a combination of all CLST and T+MA approaches.

Muneer et al. proposed a sentence/passage level benchmark for English-Urdu language pair for measuring CLTRD (called the CLEU corpus) [34]. There is a total of 3,235 CLTR document pairs based on real cases. The benchmark is manually labeled into three categories as (Near Copy = 751, Paraphrased Copy = 1751, Independently Written = 733) with source in English and derived in Urdu languages. To develop and evaluate CLTRD systems for the English-Urdu language pair, three sets of methods (N-gram Overlap, Greedy String Tiling (GST), and Longest Common Sub-sequence) using T+MA were applied on their proposed CLEU sentence/passage corpus. The best performance was obtained (F_1 = 0.732) using N-gram Overlap (unigram) and (F_1 = 0.552) using Greedy String Tiling (GST-mml1) for binary and ternary classification tasks respectively.

In addition, Muneer et. al. [31] have proposed new methods for the CLTRD for the English-Urdu language pair at the sentence/passage level. The authors have proposed and compared T+MA-based methods using the probabilistic, word embedding, semantic, and deep learning methods. The best performance was reported using ‘Comb-All’ method with (F_1 = 0.77), and (F_1 = 0.61) for the binary, and ternary classification tasks respectively.

Recently, Haneef et al. proposed a document-level benchmark for the English-Urdu language pair for measuring CLPD [22]. There is a total of 2,395 CLPD document pairs with the source (in English) - derived (in Urdu), based on simulated cases of CLPD. The benchmark is comprised of 540 automatic translations, 539 artificially paraphrased, 508 manually paraphrased, and 808 Non plagiarized. The authors compared N-gram overlap and the longest common sub-sequence for the development of the CLPD system. The best results were obtained using N-gram Overlap (unigrams) with mean similarity scores of 1.00, 0.68, 0.52, and 0.22 for automatic translation, artificially paraphrased, manually paraphrased, and Non plagiarized documents, respectively.

Table 1 shows the summarized literature as well as research gaps and highlights the specific contributions of the proposed work. It can be seen from the table, different benchmark corpora including lexical, syntactical, phrasal [33], sentence [32], and sentence/passage [34] have been developed for the task of CLTRD for English-Urdu language pair. The existing corpora contain artificial, simulated, and real cases of CLTR and CLP at the lexical, syntactical, phrasal, sentence, and passage levels. However, the problem of CLTR has not been explored for real cases at the document level for the English-Urdu language pair. To overcome this limitation, this study proposes

Table 1. Summary of the literature review

Corpus	Reuse Type	No of source documents	No of suspicious Documents	Obfuscation level	Granularity Level	Language Pair	Applied Methods	Best Results
CLEU-Sen [32]	Simulated	21,699	21,699	Wholly Derived = 7,655, Partially Derived = 6,461, Non Derived = 7,553	Sentence level	English-Urdu	1. Translation Plus Mono-lingual Analysis 2. Cross-lingual word Embedding 3. Cross-lingual Sentence Transformer	F_1 = 0.94 for binary F_1 = 0.84 and ternary classification tasks using combination of all
CLEU-Lex [33]	Simulated	66,485	66,485	Wholly Derived = 22,236, Partially Derived = 20,315, Non Derived = 23,934	Lexical level	English-Urdu	1. Bi-lingual Dictionary 2. Cross-Lingual Sentence Transformers 3. Cross-lingual Word Embedding 4. Cross-Lingual Semantic Tagger	F_1 = 0.80, and F_1 = 0.69 for the binary and ternary classification tasks using Cross-lingual Sentence Transformers
CLEU-Syn [33]	Simulated	60,267	60,267	Wholly Derived = 20,007, Partially Derived = 16,979, Non Derived = 23,281	Syntactical level	English-Urdu	1. Bi-lingual Dictionary 2. Cross-Lingual Sentence Transformers 3. Cross-lingual Word Embedding 4. Cross-Lingual Semantic Tagger	F_1 = 0.92, and F_1 = 0.82 for the binary and ternary classification tasks using Cross-lingual Sentence Transformers, Cross-lingual Word Embedding, and Cross-lingual Semantic Tagger
CLEU-Phr [33]	Simulated	60,106	60,106	Wholly Derived = 23,862, Partially Derived = 15,878, Non Derived = 20,366	Phrasal level	English-Urdu	1. Bi-lingual Dictionary 2. Cross-Lingual Sentence Transformers 3. Cross-lingual Word Embedding 4. Cross-Lingual Semantic Tagger	F_1 = 0.94, and F_1 = 0.78 for the binary and ternary classification tasks using Cross-lingual Sentence Transformers, Cross-lingual Word Embedding, and Cross-lingual Semantic Tagger
CLEU [34]	Real	3,235	3,235	Near Copy = 751, Paraphrased Copy = 1751, Independently Written = 733	Sentence/Passage level	English-Urdu	Translation Plus Mono-lingual Analysis	F_1 = 0.77, and F_1 = 0.61 for the binary, and ternary classification all T+MA
EU-CLPD [22]	Artificial, Simulated	2,395	2,395	540 automatic translation, 539 artificially paraphrased, 508 manually paraphrased, and 808 Non plagiarized	Document level	English-Urdu	Translation Plus Mono-lingual Analysis	1.00, 0.68, 0.52, and 0.22 using N-gram
					Proposed Work			
TREU	Real	2,257	2,257	Derived = 672, Partially Derived = 888, Non Derived = 697	Document level	English-Urdu	Translation Plus Mono-lingual Analysis	F_1 = 0.66 and F_1 = 0.78 for the binary, and ternary classification all T+MA

a gold-standard benchmark corpus containing real cases of CLTR for English-Urdu pair at document. In addition, we applied various T+MA-based methods including lexical overlap, string matching, structural similarity, mono-lingual word embedding, and mono-lingual sentence embedding. To our knowledge, the proposed gold-standard benchmark corpus based on real cases is the first corpus for CLTRD for the English-Urdu language pair.

3 CORPUS CREATION

In this section, we discuss the corpus creation process in detail which includes data collection, annotations, corpus characteristics, and examples from the TREU corpus.

3.1 Data Collection

The TREU (Text Reuse English Urdu) Corpus is created in the footsteps of COUNTER [40] and METER [12] corpora, i.e., compiling data from journalism. The idea was motivated by the fact that a large amount of journalistic text is freely available and a lot easier to extract in electronic form, especially for the Urdu language. Moreover, borrowing text from the news agency to compose newspaper stories is a well-known practice in journalism. It is a routine task for journalists to formulate a news story by using the press report released by the news agency either directly (verbatim) or by rephrasing (paraphrasing) it [11, 43]. In addition, it is important to investigate the behavior of state-of-the-art cross-lingual text reuse detection methods on these real examples of reuse. COUNTER [40] is a document-level corpus for the task of monolingual text reuse detection for the Urdu language only. METER [12] is a document-level corpus for task monolingual text reuse detection for the English language only.

Table 2. Distribution of documents by news agencies, newspapers and domains in the TREU corpus

News Agencies		News Papers		Domains	
APP	2,015	Nawa-e-Waqt	1,525	National	1,127
INN	242	Daily Express	663	Foreign	538
		Daily Jang	57	Domestic	339
		Daily Pakistan	12	Sports	225
			55	Business	28

It can be seen that both corpora are available for monolingual text reuse detection, highlighting the fact that they cannot be used for the task of CLTRD for the English-Urdu language pair.

The TREU corpus has two types of text documents: source text documents in the English language and derived text documents in the Urdu language. To create source text documents, the press reports released by two well-known news agencies of Pakistan, i.e., Associated Press of Pakistan (APP) and Independent News Pakistan (INP) are used. A subscription was established with both news agencies to receive English news reports daily by email. On the other hand, the derived text documents were hand-picked from the Urdu news stories published in the top four large circulation national dailies of Pakistan, i.e., Nawa-i-Waqt, Daily Express, Daily Pakistan, and Daily Jang. The newspaper stories were collected manually over a period of 12 months (from July 2015 to June 2016). The news text collection was carried out throughout each month excluding the public holidays on which either the newspaper was not published, or the news agency did not provide the service. To have variation in the data, the news data was collected across National, Foreign, Domestic, Sports, and Business domains.

Table 2 shows the distribution of text documents in the TREU corpus. In the table, 'News Agencies' refer the distribution of source document (English Document), and 'News Papers' refer the distribution of reused (Urdu) text taken from different news papers. Whereas 'domain' repents the subject wise distribution of the corpus including National, Foreign, Domestic, Sports, and Business.

3.1.1 *Annotation Guidelines.*

As a first step, an annotation scheme was prepared under the guidance of a linguist. The following are the key points of the annotation scheme used to tag a text document pair in one of the three classes, i.e., Wholly Derived, Partially Derived, or Non Derived.

Wholly Derived A text pair was tagged as 'WD' if the derived text is almost an exact translation of the source text. However, due to the cross-lingual setting, small changes appearing in the derived text were ignored. Additionally, a small amount of new text may also appear in the derived text due to the structural difference in both languages.

Partially Derived A text pair was tagged as 'PD' if contents in both texts were semantically the same, i.e., describing the same story (or information). However, the derived text was not the mere translation of the source text. Rather, the source text was paraphrased using different text editing operations including (but not limited to) word or sentence re-ordering, merging or splitting of sentences, insertions or deletions of new text, replacing words or phrases with appropriate synonyms, expansion or compression of text, etc.

Non Derived A text pair was tagged as 'ND' if the context of the news story was the same in both texts or if they both were describing the same event. However, the derived text was not borrowed from the news agency text (although there may be individual words that co-occur). Moreover, possibly a lot more new information may be present in the derived text with completely different facts and figures (this shows that the journalist who formulated the news story has not used the news agency's report as a source).

3.1.2 *Annotations.*

Two human annotators performed the annotations of the TREU corpus with the help of a linguist. Both the annotators were postgraduate NLP students, native speakers of the Urdu language, who studied

Table 3. Statistics of the TREU corpus

Corpus Statistics	Source	Derived
Total number of documents	2,257	2,257
Total number of words	486,264	522,805
Average number of words per document	215	231
Total number of types (unique words)	24,105	17,736
Smallest document (by words)	25	26
Largest document (by words)	1,799	2,404
Number of documents 1000 words	9	33
Number of documents 500 but 1000 words	124	139
Number of documents 100 but 500 words	1,623	1,564
Number of documents 100 words	486	512

English as a foreign language and as the language of instruction throughout their academic careers. Furthermore, they were provided with training about the journalistic text reuse phenomena and with tutorials on different text rewriting operations by the linguist.

Annotations were performed in multiple phases. In the first phase, based on the annotation scheme, a random subset of 50 text document pairs was annotated by the two annotators and the linguist. The results of each annotator were compared with the linguist and conflicting pairs were discussed with them individually. Moreover, the annotation scheme was re-examined after the discussion to make a few changes. In the second phase, another subset of 250 text document pairs was now annotated by the two human annotators according to the revised scheme. The results were reviewed by the linguist again and it was observed that the rate of conflicts had dropped. During the third phase, the two annotators manually tagged the remaining 1,957 text document pairs and the results were saved. Both annotators agreed on 1,919 and disagreed on 338 text document pairs.

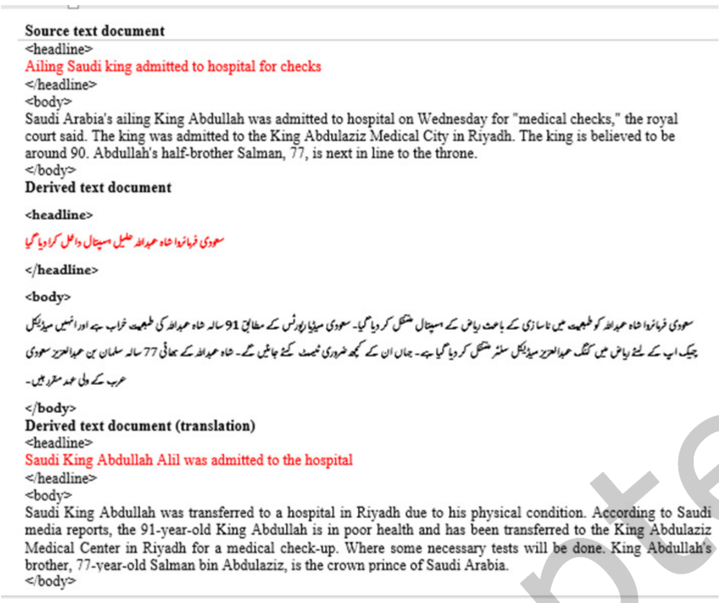
3.1.3 Inter-Annotator Agreement. The final Inter-Annotator Agreement (IAA) score on the entire corpus is 85.02%, and the Cohen's Kappa score was computed to be 0.77% (Unweighted), 0.82% (Linear weighting), 0.87% (Quadratic weighting) [13]. As can be noted, these scores are of a substantial level considering the difficulty of the annotation task. Besides, this draws attention to the fact that annotation guidelines were well defined which assisted annotators to recognize between various levels of CLTR in the TREU Corpus. In addition, this also shows that annotators were well-trained and had expertise in the field of CLTR.

In the last phase, the 338 conflicting text document pairs were given to the journalist for conflict resolution. The decisions of the third annotator were considered final. The final gold standard corpus contains 2,257 text document pairs, out of which 672 are WD, 888 are PD, and 697 are ND.

3.2 Corpus Statistics

Table 3 shows the detailed statistics of the TREU corpus. It contains a total of 4,514 text documents (2,257 source and 2,257 derived text documents). It is substantially large in size and contains in total 1,009,069 (approx. one million) words (tokens), out of which 486,264 are English and 522,805 are Urdu words. The average length of an English source text document is 215 words while for an Urdu-derived text document it is 231 words. The corpus is saved in a standard XML format and available as a free download resource[1].

[1] The sample corpus (100 text documents) can be accessed from https://drive.google.com/drive/folders/1G9gvDrc0ULWJe82CwAnaH0UTLoju9FV2?usp=sharing for the reviewers. We will upload the full corpus once the paper is accepted.



Source text document
<headline>
Ailing Saudi king admitted to hospital for checks
</headline>
<body>
Saudi Arabia's ailing King Abdullah was admitted to hospital on Wednesday for "medical checks," the royal court said. The king was admitted to the King Abdulaziz Medical City in Riyadh. The king is believed to be around 90. Abdullah's half-brother Salman, 77, is next in line to the throne.
</body>
Derived text document

<headline>

سعودی فرمانروا شاہ عبداللہ علیل ہسپتال داخل کرادیا گیا

</headline>

<body>

سعودی فرمانروا شاہ عبداللہ کو طبیعت میں ناسازی کے باعث ریاض کے ہسپتال منتقل کر دیا گیا۔ سعودی میڈیا رپورٹس کے مطابق 91 سالہ شاہ عبداللہ کی طبیعت خراب ہے اور انہیں میڈیکل
چیک اپ کے لئے ریاض میں کنگ عبدالعزیز میڈیکل سنٹر منتقل کر دیا گیا ہے۔ جہاں ان کے کچھ ضروری ٹیسٹ کئے جائیں گے۔ شاہ عبداللہ کے بھائی 77 سالہ سلمان بن عبدالعزیز سعودی
عرب کے ولی عہد مقرر ہیں۔

</body>
Derived text document (translation)
<headline>
Saudi King Abdullah Alil was admitted to the hospital
</headline>
<body>
Saudi King Abdullah was transferred to a hospital in Riyadh due to his physical condition. According to Saudi media reports, the 91-year-old King Abdullah is in poor health and has been transferred to the King Abdulaziz Medical Center in Riyadh for a medical check-up. Where some necessary tests will be done. King Abdullah's brother, 77-year-old Salman bin Abdulaziz, is the crown prince of Saudi Arabia.
</body>

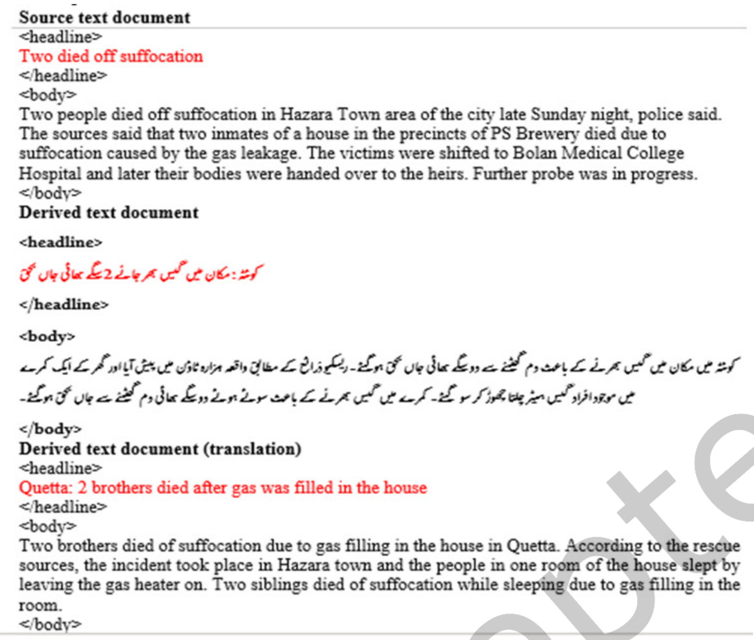
Fig. 1. Example of Wholly Derived

3.3 Examples from proposed corpus

Figure 1 shows a WD text document example pair from the corpus. It can be noted that the derived text is almost the exact translation of the source text. Moreover, the order of information is also preserved. However, a very small amount of information is added or removed in the derived text document due to language structural differences. Furthermore, the source text document has one sentence (Principal Staff Officers and a large number of Airmen attended the ceremony) that is not present (derived) in the derived text document.

Figure 2 shows a cross-lingual PD text reuse example from the corpus. It is worth noting that sentences (or phrases) have been reordered to generate the derived text. The information at the start of the source text document is added (after paraphrasing) at the end of the derived text document. Moreover, some extra details have been added in the derived text (which may be based on the journalist's observations), i.e., the name of the person who offered the Namaz-e-Janaza. The source text document has general information (representatives of MQM) whereas the derived text document has more detailed and specific information, i.e., actual names of the representatives. Furthermore, some words have been replaced with appropriate synonyms. These changes highlight the fact that different editing operations have been used by journalists in formulating the newspaper story. However, while creating the derived text, the meanings of the source text have been preserved.

Figure 3 shows an ND text document pair from the corpus. Both source and derived texts are describing the same news event, i.e., proceedings of a Senate meeting and the walkout of members from the meeting. However, the explanation of the event and the way of expressing it are entirely different in the source and derived text documents. In the source text document, two members (Haji Adeel and Zahid Khan) are requesting the Deputy Chairman to adjourn the proceedings whereas the derived text states it was requested collectively by the opposition members. Furthermore, in the source text, it is mentioned that the meeting was adjourned for half an hour while the derived text details that it was restarted after half an hour but postponed again until Friday. In addition to this, the information is very compressed in the source text whereas the event has been reported in



Source text document

<headline>
Two died off suffocation
</headline>
<body>
Two people died off suffocation in Hazara Town area of the city late Sunday night, police said. The sources said that two inmates of a house in the precincts of PS Brewery died due to suffocation caused by the gas leakage. The victims were shifted to Bolan Medical College Hospital and later their bodies were handed over to the heirs. Further probe was in progress.
</body>

Derived text document

<headline>

کوئٹہ : مکان میں گیس بھر جانے 2سگے بھائی جاں بحق

</headline>

<body>

کوئٹہ میں مکان میں گیس بھرنے کے باعث دم گھٹنے سے دو سگے بھائی جاں بحق ہوگئے۔ ریسکیو ذرائع کے مطابق واقعہ ہزارہ ٹاؤن میں پیش آیا اور گھر کے ایک کمرے
میں موجود افراد گیس ہیٹر چلتا چھوڑ کر سو گئے۔ کمرے میں گیس بھرنے کے باعث سوئے ہوئے دو سگے بھائی دم گھٹنے سے جاں بحق ہوگئے۔

</body>

Derived text document (translation)

<headline>
Quetta: 2 brothers died after gas was filled in the house
</headline>
<body>
Two brothers died of suffocation due to gas filling in the house in Quetta. According to the rescue sources, the incident took place in Hazara town and the people in one room of the house slept by leaving the gas heater on. Two siblings died of suffocation while sleeping due to gas filling in the room.
</body>

Fig. 2. Example of Partially Derived

greater depth in the derived text document. This shows that derived text is generated independently of the source text and any overlap of words (phrases) is very low (mainly stop-words are common) between the text pair.

4 CROSS-LINGUAL TEXT REUSE DETECTION METHODS

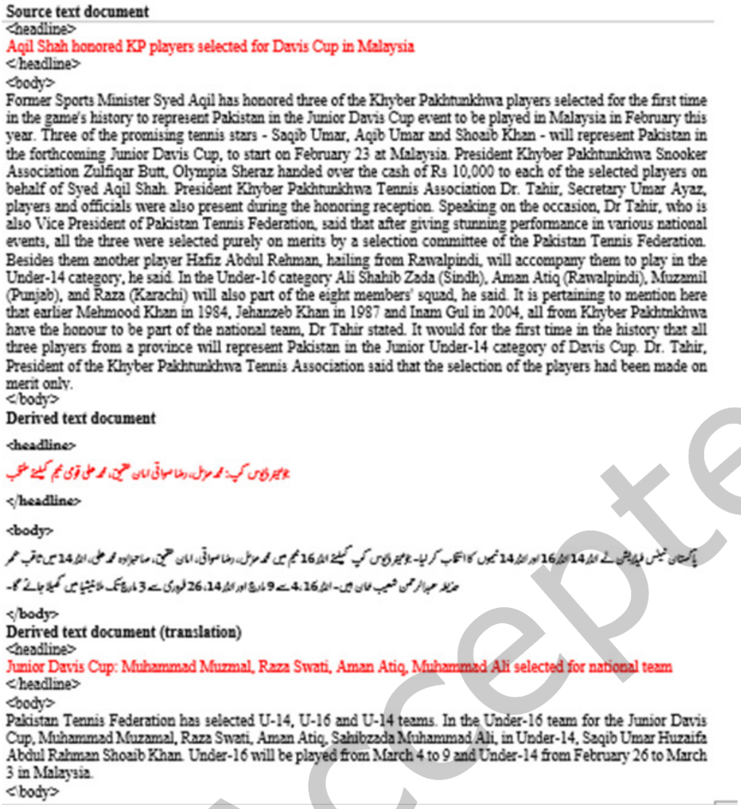
A range of Cross-lingual Text Reuse Detection methods are applied on the TREU corpus to show its usefulness and how it could be utilized in the development and evaluation of cross-lingual (English-Urdu) text reuse detection systems. The CLTRD methods using T+MA for the proposed corpus include lexical overlap, string matching, structural similarity, mono-lingual word embedding, and mono-lingual sentence embedding. To the best of our knowledge, this is the first study that has applied these diverse methods on an English-Urdu cross-lingual text reuse corpus at the document level for real cases[2]. Now, we will discuss these methods in detail.

4.1 Translation plus mono-lingual analysis

The Translation + Monolingual Analysis (T+MA) method is based on Machine Translation for the task of cross-lingual text reuse detection and has been very popular and widely used because of its simplicity [5]. The method first translates the source or derived text documents into one language and then addresses the task as mono-lingual text reuse detection. The translation is usually performed using an automatic MT system Google Translator[3]. For the cross-lingual (English-Urdu) text reuse detection experiments performed on the TREU corpus using the T+MA method, the derived text documents are translated from Urdu to English using Google Translate. In the next step, the English text is first pre-processed to remove the punctuation marks, extra white spaces, newline characters, foreign characters, numbers, and single alphabet tokens. It is then lemmatized using the Stanford Lemmatiser

[2] All experiments are performed using Python v3.6, Scikit-learn v0.19.0, SciPy v1.6.0, and Gensim v3.7.0.

[3] https://translate.google.com: LAST VISITED: 20-Sep-2021)



Source text document

<headline>

Aqil Shah honored KP players selected for Davis Cup in Malaysia

</headline>

<body>

Former Sports Minister Syed Aqil has honored three of the Khyber Pakhtunkhwa players selected for the first time in the game's history to represent Pakistan in the Junior Davis Cup event to be played in Malaysia in February this year. Three of the promising tennis stars - Saqib Umar, Aqib Umar and Shoaib Khan - will represent Pakistan in the forthcoming Junior Davis Cup, to start on February 23 at Malaysia. President Khyber Pakhtunkhwa Snooker Association Zulfiqar Butt, Olympia Sheraz handed over the cash of Rs 10,000 to each of the selected players on behalf of Syed Aqil Shah. President Khyber Pakhtunkhwa Tennis Association Dr. Tahir, Secretary Umar Ayaz, players and officials were also present during the honoring reception. Speaking on the occasion, Dr Tahir, who is also Vice President of Pakistan Tennis Federation, said that after giving stunning performance in various national events, all the three were selected purely on merits by a selection committee of the Pakistan Tennis Federation. Besides them another player Hafiz Abdul Rehman, hailing from Rawalpindi, will accompany them to play in the Under-14 category, he said. In the Under-16 category Ali Shahib Zada (Sindh), Aman Atiq (Rawalpindi), Muzamil (Punjab), and Raza (Karachi) will also part of the eight members' squad, he said. It is pertaining to mention here that earlier Mehmood Khan in 1984, Jehanzeb Khan in 1987 and Inam Gul in 2004, all from Khyber Pakhtnkhwa have the honour to be part of the national team, Dr Tahir stated. It would for the first time in the history that all three players from a province will represent Pakistan in the Junior Under-14 category of Davis Cup. Dr. Tahir, President of the Khyber Pakhtunkhwa Tennis Association said that the selection of the players had been made on merit only.

</body>

Derived text document

<headline>

جونیئر ڈیوس کپ: محمد مزمل، رضا سواتی، امان عتیق، محمد علی قومی ٹیم کیلئے منتخب

</headline>

<body>

پاکستان ٹینس فیڈریشن نے انڈر 14، انڈر 16 اور انڈر 14 ٹیموں کا انتخاب کر لیا۔ جونیئر ڈیوس کپ کیلئے انڈر 16 ٹیم میں محمد مزمل، رضا سواتی، امان عتیق، صاحبزادہ محمد علی، انڈر 14 میں ثاقب عمر
حذیفہ عبدالرحمن شعیب خان ہیں۔ انڈر 16، 4 سے 9 مارچ اور انڈر 14، 26 فروری سے 3 مارچ تک ملائیشیا میں کھیلا جائے گا۔

</body>

Derived text document (translation)

<headline>

Junior Davis Cup: Muhammad Muzmal, Raza Swati, Aman Atiq, Muhammad Ali selected for national team

</headline>

<body>

Pakistan Tennis Federation has selected U-14, U-16 and U-14 teams. In the Under-16 team for the Junior Davis Cup, Muhammad Muzamal, Raza Swati, Aman Atiq, Sahibzada Muhammad Ali, in Under-14, Saqib Umar Huzaifa Abdul Rahman Shoaib Khan. Under-16 will be played from March 4 to 9 and Under-14 from February 26 to March 3 in Malaysia.

<\body>

Fig. 3. Example of Non Derived

[28]. NLTK is used for word tokenization and stop-word removal from the text [6]. Lastly, case-folding is applied to convert the text to lowercase.

Afterwards, the similarity score between a text pair is obtained by applying a diverse range of monolingual text reuse detection methods. The applied methods are classified under five categories, (1) lexical overlap, (2) string matching, (3) structural similarity, (4) monolingual word embeddings, and (5) mono-lingual sentence embeddings. For lexical overlap, Word n-grams overlap and Vector Space Model are applied. For string matching, Longest Common Subsequence, and Greedy String Tiling are used. For structural similarity, Stop-word n-grams overlap is chosen. For monolingual word embeddings, averaged embeddings, weighted averaged embedding, and weighted maximum embeddings variants are applied. For the more recent mono-lingual sentence embeddings, Sent2Vec, InferSent, Universal Sentence Encoder, and LASER are used.

4.1.1 *Lexical overlap.*

Word N-grams Overlap. The word N-grams overlap method tries to estimate the number of common N-grams between source and derived text documents. It is one of the simplest methods used in text reuse detection that could easily be applied to a large collection of texts because of its low complexity. For the experiments performed on the TREU corpus, word N-grams are generated from the source and derived text documents by varying the lengths of n from [1˘5]. Moreover, the similarity between the sets of unique N-grams is computed using four different similarity measures, i.e., Containment 4, Jaccard 1, Overlap 2, and Dice 3.

The equation of Jaccard similarity

$$S_{Jaccard} = \frac{|S(st,n) \cap S(dt,n)|}{(|S(st,n)| \cup |S(dt,n)|)} \tag{1}$$

The equation of overlap similarity is

$$S_{overlap} = \frac{|S(st,n) \cap S(dt,n)|}{min(|S(st,n)|, |S(dt,n)|)} \tag{2}$$

The equation of dice similarity

$$S_{Dice} = \frac{|S(st,n) \cap S(dt,n)|}{(|S(st,n)| + |S(dt,n)|)} \tag{3}$$

The equation of containment similarity is :

$$S_{Containment} = \frac{|S(st,n) \cap S(dt,n)|}{(|S(st,n)|)} \tag{4}$$

Vector Space Model. The Vector Space Model is another method used for calculating the degree of similarity between a given text pair. Using this method, the source and derived text documents are represented in a high dimensional vector space and similarity between them is calculated using the cosine similarity. For these experiments, Vector Space Model is applied in two ways i.e., (1) Bag-of-Words (VSM-BoW) and (2) Character N-Grams (VSM-CNG).

For VSM-BoW, each source and derived text document is first converted into its BoW representation. The individual terms (words) are then weighted using the *tf-idf* weighting scheme. After that, the text documents are converted into vectors and similarity between the vectors is calculated using cosine similarity (equation 5).

For VSM-CNG, in the first step, all white space characters in the source and derived text documents are replaced with *hyphen* "-" and then the text is codified into character *n*-grams of size [3−5]. These *n*-grams are then weighted using *tf-idf* and converted into vectors. Subsequently, the similarity score between source and derived text document vectors is estimated using the cosine similarity (equation 5).

$$Sim(S,D) = \frac{\overrightarrow{S}.\overrightarrow{D}}{|\overrightarrow{S}| \times |\overrightarrow{D}|} \tag{5}$$

Where $|(\overrightarrow{S})|$ and $|\overrightarrow{D}|$ represent the length of the source and derived text respectively. The cosine similarity measure allows partial matching, which enables a better estimation of similarity.

4.1.2 String matching.

Longest Common Subsequence. The Longest Common Sub-sequence is a string-matching method that computes the longest group of elements (words) that are common between the two texts and are in the same order in each text. For the experiments conducted on the TREU corpus, the normalized LCS score (LCS_{norm}), between each source and derived text document, is calculated by dividing the length of LCS by the length of the shorter text document 6.

$$LCS_{norm}(s,d) = \frac{|LCS|}{min(|(s|, |(d)|)} \tag{6}$$

In equation 6, $|(s)|$ and $|d|$ represent the length of the source and reused texts, respectively.

Greedy String Tiling. The Greedy String Tiling identifies the longest rewritten sequence of substrings from the source text and returns the sequence (as tiles) paired with the derived text. To avoid very short matching lengths, a minimum match length (mML) value is used. It is a powerful algorithm that may detect matches even if some of the text is deleted or if additional text has been inserted.

For these experiments, the well-known Running Karp-Rabin Matching and Greedy String Tiling implementation is used [44] and the length of MML is varied [1–5]. The normalized GST similarity score (GST$_{norm}$) is calculated by taking the ratio of the length of GST and the length of the shorter text document (equation 7).

$$GST_{norm}(s,d) = \frac{|GST|}{min(|(s|,|(d)|)} \tag{7}$$

In equation 7, $|(s)|$ and $|d|$ represent the length of the source and reused texts, respectively.

4.1.3 Structural similarity.

Stop-word N-grams Overlap. Similar to the Word *n*-grams overlap method, Stop-word *n*-grams overlap is used to measure the degree of stop-word overlap between a text document pair.

For these experiments, the source and derived text documents are first filtered to remove content words. Subsequently, N-grams are generated for the remaining stop-words in the text by varying the length of n [1–5]. Eventually, the similarity between the sets of unique stop word N-grams is computed using four different similarity measures i.e., Containment (equation 4), Jaccard (equation 1), Overlap (equation 2), and Dice (equation 3).

4.1.4 Mono-lingual word embeddings. The main idea of monolingual word embeddings is to represent words as continuous vectors in a multidimensional vector space [30]. This representation enables the capture of the semantic and syntactic properties of the text. The underlying assumption, from the domain of distributional semantics, is that the words which occur close to each other are semantically similar or have similar meanings.

Several monolingual word embeddings models are available, e.g., Word2Vec [30], GloVe [36], fastText [7], etc. that are trained on large corpora using unsupervised methods, i.e., Continuous Bag-of-Words (CBOW) and Skip-gram. The CBOW predicts the word based on the context of its surrounding words whereas Skip-gram predicts the context word(s), surrounding the word itself. These models are capable of capturing some elements of the context of a word, its semantics, and its relation with other words, although their precise properties are still being evaluated [45]. Consequently, they have been shown to benefit performance for a number of NLP tasks including IR [41], text similarity [23], topic modeling [26], sentiment analysis [46], and authorship analysis [39].

The commonly used text reuse detection methods (N-gram overlap, LCS, GST, etc.) rely on the surface form of the text only, whereas word embeddings could be used to estimate the semantic similarity between pair of words (or vectors) [23, 29]. Therefore, in this work, monolingual word embedding-based methods are used to capture the semantic level similarities between source and derived text documents.

For the experiments performed on the TREU corpus using monolingual word embeddings, both pre-trained and custom-trained models are used. The pre-trained models are Google Word2Vec [30], Stanford NLP GloVe [36], and Facebook fastText [7].

Moreover, all three models are also custom trained on English news data. For training, 105k text documents collected during the development of the TREU corpus are used. These are the English news reports, in plain text format, released by the news agencies (henceforth called Pakistan English News (PEN) corpus). The PEN corpus contains 16,120,843 words and 139,634 types. The corpus text is pre-processed and all three models (i.e., Word2Vec, GloVe, and fastText) are trained using Gensim ("Generate Similarly") toolkit [37] with the same parameter settings, i.e., dimension 300, min-count 5, and windows-size 10. Different dimensions (50, 100, 300) were tested and we determined that 300 works the best. To estimate the similarity between the source and derived text documents using monolingual word embeddings, three different methods are used, (1) averaged embeddings, (2) weighted averaged embeddings, and (3) weighted maximum embeddings.

Averaged embeddings. For the averaged embeddings method, a simple average of all the word embedding vectors in a document is calculated to generate the document vector. For instance, a text document d, composed of words $\{w_1, w_2, w_3, ..., w_n\}$, the word embedding vectors for each word are $\{v_{w_1}, v_{w_2}, v_{w_3}, ..., v_{w_n}\}$. The averaged embedding vector V_d for document d is calculated using equation 8.

$$V_d = \frac{1}{n}\sum_{i=1}^{n} v_{w_i} \tag{8}$$

In equation 9, w_i is the ith word of the document d and n is the number of words in the document.

For the experiments performed on the TREU corpus, the source and derived text documents are first converted to their BoW representations, and word embedding vectors are obtained for the set of unique words in both text documents. For each text document, all the word embedding vectors are averaged to obtain the resultant document vectors. Finally, the source and derived averaged embedding document vectors are normalized and the degree of similarity between them is computed using cosine similarity (equation 5).

Weighted averaged embeddings. Taking the simple average of the word embedding vectors of constituent words in a text document tends to give too much weight to words that are semantically irrelevant. This can possibly be addressed, to some extent, by taking a weighted average of the word embedding vectors. The weights to individual words may be assigned using *pos* weights, *idf* weights, etc.

The weighted averaged embedding vector WV_d of a document d is calculated using equation 9.

$$WV_d = \frac{1}{n}\sum_{i=1}^{n}(idf(w_i).v_{w_i}) \tag{9}$$

In equation 9, *idf* is the function that returns the *idf* value of the ith word w_i, v_{w_i} is the word embedding vector of the ith word w_i and $\cdot$ is the scalar product. Once the weighted averaged embedding document vectors for both sources and derived text documents are generated, the process of computing similarity is similar to the averaged embedding method. Moreover, for these experiments, *idf* weights for each word are computed using the PEN corpus.

Weighted maximum embeddings. Averaged embeddings and weighted averaged embeddings are computationally cheap and based on BoW representations. However, one major drawback of the BoW representation is the loss of word order which results in corrupting the semantics of the text. Though weighting schemes give importance to individual words, they also suffer from the same word order issue. Moreover, for large text documents, using an averaging or linear summation of word vectors, the resultant document vectors ultimately start to approximate each other. The weighted maximum embeddings method works as follows.

Consider a source text document s containing words $\{w_1, w_2, w_3, ..., w_n\}$, and a derived text document d containing words $\{w'_1, w'_2, w'_3, ..., w'_m\}$. In the first step, sets of unique words from both text documents are converted to their respective word vectors, i.e., $\{v_{w_1}, v_{w_2}, v_{w_3}, ..., v_{w_n}\}$ and $\{v_{w'_1}, v_{w'_2}, v_{w'_3}, ..., v_{w'_m}\}$, respectively. After that, cosine similarity (equation 5) is computed for each normalized word vector from the derived text document paired with every normalised word vector in the source text document {cos-sim($v_{w'_1} \leftrightarrow v_{w_1}$), cos-sim($v_{w'_1} \leftrightarrow v_{w_2}$), ..., cos-sim($v_{w'_1} \leftrightarrow v_{w_n}$), and so on}. However, only the maximum similarity is recorded for each source-derived word pair (vector). The resultant maximum cosine similarity scores are multiplied with the *idf* weights of the words from the derived text document. The final similarity between a source and derived text document pair is computed using equation 10 by taking the ratio of sum of all weighted maximum cosine similarity scores and sum of all derived text document word *idf* weights.

$$sim(s,d) = \frac{\sum_{w'_i \in d} idf(w'_i) \times max_{w_j \in s, w'_i \in d} cosine(v_{w'_i}.v_{w_j})}{\sum_{w'_i \in d} idf(w'_i)} \tag{10}$$

In equation 10, w, w′, v_w, and $\mathrm{v}_{w'}$ are the sets of words and their respective vectors from the source and derived text documents, respectively. *idf* is the function that returns the *idf* weight, cosine is cosine similarity (equation 5), and · is the scalar product.

For these experiments, sets of unique words from the source and derived text documents are converted to their word embeddings vectors and *idf* weights are calculated using the PEN corpus. The word-level similarity is measured using cosine similarity (equation 5) and the final similarity score is computed using equation 10.

4.1.5 Mono-lingual sentence embeddings. The unsupervised word embeddings are best suited for word-level similarity. However, to better estimate semantic relatedness (meaning of words) between pairs of sentences or documents, contextual information and word order are important. Besides, supervised learning, presumably, can be more effective in learning the actual meaning of a word in a given sentence (or document).

For this purpose, pre-trained supervised and unsupervised sentence embedding models are available which are similar to word embeddings but for sentences. These models are pre-trained (some of them have an option to fine-tune or custom train) on large corpora to capture as much semantic and syntactic information of lexical units (words) as possible.

To capture the similarity between the source and derived text documents from the TREU corpus, this study uses four sentence embedding models, (1) Sent2Vec, (2) InferSent, (3) Universal Sentence Encoder, and (4) LASER. Each of these models outputs a fixed-length sentence embedding vector on a given input sentence of any length. Using these models, the degree of similarity between a source and derived text document is computed as follows: All the sentences [4] from each source and derived text document are converted to their respective sentence vectors using one of the sentence embedding models [28]. These sentence embedding vectors are then summed to produce the document-level vector representation. Each vector is normalized and the closeness between the source and derived document vectors is estimated using cosine similarly (equation 5).

In the next sections, each of the sentence embedding models and their work is described.

Sent2Vec. Sent2Vec is an unsupervised sentence embedding model that learns the distributed representations of sentences (or short texts) using the CBOW approach [35]. The model simply combines (by averaging) word embeddings with *n*-grams embeddings of each word in a sentence. The method has proven to be beneficial in many NLP tasks such as sentiment analysis [25], IR [2], word similarity [21], and text classification [1].

For these experiments, the pre-trained as well as, custom-trained mono-lingual Sent2Vec models are used. The pre-trained model[5] used is based on the Toronto Books corpus [47] with bi-grams and 700-dimensions. For custom training, all the sentences from the PEN corpus are used to train the model with exactly the same parameters i.e., bi-grams and 700-dimensions.

InferSent. InferSent is a pre-trained supervised model developed by Facebook [15]. It is a neural network-based model, trained on 570k human-written English sentence pairs from the Stanford Natural Language Inference (SNLI) corpus [8]. The model has recently found success in sentiment analysis [4], text summarisation [16], and question-answering systems [10] tasks.

For the experiments performed on the TREU corpus, InferSent mono-lingual model pre-trained[6] [8] on the SNLI corpus [8] is used. The model is trained with the BiLSTM encoder with max pooling, batch-size 64, and word embeddings dimension 300. It outputs a 2,048-dimension sentence embedding vector for an input sentence

[4] Stanford sentence tokenizer is used for sentence boundary detection.

[5] https://drive.google.com/open?id=0B6VhzidiLvjSdENLSEhrdWprQ0k

[6] Only pre-trained model is used as custom training is not possible because of the nature of the training corpus, i.e., SNLI

of any length. Moreover, two variations of the input word embeddings are used, (1) Glove [7] [36] and (2) fastText [8] [7]. Both word embedding models are custom trained on the PEN corpus with dimension 300, min-count 5, and windows-size 10.

Universal Sentence Encoder. The Universal Sentence Encoder, developed by Google, is a supervised sentence embedding model that takes a sentence of any length as input and converts it into a 512-dimension fixed-length vector [9]. Two versions of the model are available, both mono-lingual, trained on a variety of data sources, i.e., news websites, discussion groups, Wikipedia, and the SNLI corpus [8]. First is the advanced transformer-based architecture that uses attention to calculate context-aware embeddings of words in a sentence. These embeddings are then averaged to obtain sentence embeddings. The attention architecture takes care of the ordering and identity of words in the text. The second variant, called Deep Averaging Network (DAN), averages the uni-gram and bi-gram embeddings of all words together. The embeddings are then passed through a deep neural network to generate sentence embeddings. The transformer model has outperformed the DAN model on a number of tasks on the SentEval [14] and GLUE [42] benchmarks.

For these experiments, both transformer[9] and DAN[10] pre-trained models [11] are used.

LASER. LASER (language-agnostic SEntence Representations) is an encoder-decoder architecture, released by Facebook, that converts multi-lingual sentences to fixed-length vector representations [3]. It is pre-trained on 223M parallel texts of 90+ languages. The multi-lingual model follows a sequence-to-sequence design where the output of the encoder is used as input to the decoder. The encoder is an enhanced version of InferSent, language independent, pre-trained [12] on multi-lingual text, and the one responsible for constructing sentence embeddings. It uses a 5-layer BiLSTM (each 512-dimension) with max-pooling over the final states of the last layer. It takes a sentence as input and outputs a 1,024-dimension fixed-length vector. For these experiments, the pre-trained [13] encoder module is used.

5 EXPERIMENTAL SETUP

This section describes the corpus, evaluation methodology, and evaluation measures used for the CLTRD experiments applied on cross-lingual sentence corpus.

5.1 Corpus

The entire TREU corpus is used for the set of experiments carried out in this study. There is a total of 4,514 text documents (2,257 sources, 2,257 derived) in the corpus with three levels of text reuse. The text documents tagged as “Wholly Derived” are 672, “Partially Derived” are 888, and “Non-Derived” are 697.

5.2 Methods

The “Method” columns list the name of the methods which produced the highest result. “lo-wno-d-cmb” refers to the Word N-grams overlap method with Dice similarity measure and by combining N-grams of length [1˘5] (5 features). Similarly, “lo-wno-j-cmb” refers to the Word n-grams overlap method with Jaccard similarity measure and by combining N-grams of length [1–5] (5 features) for the classification tasks. “lo-vsm-bow”, “lo-vsm-c4g”, and “lo-vsm-c5g” refers to the Vector Space Model method applied with Bag of Words, Character 4-grams and

[7] https://dl.fbaipublicfiles.com/infersent/infersent1.pkl

[8] https://dl.fbaipublicfiles.com/infersent/infersent2.pkl

[9] https://tfhub.dev/google/universal-sentence-encoder-large/3

[10] https://tfhub.dev/google/universal-sentence-encoder/2

[11] There is no option to custom train the models.

[12] There is no option to custom train the encoder.

[13] https://dl.fbaipublicfiles.com/laser/models

Character 5-grams, respectively. “sm-lcs” refers to the Longest Common Subsequence while “sm-gst-cmb” refers to the Greedy String Tiling method applied by combining the MML length [1-5] (5 features). “ss-sno-j-cmb” and “ss-sno-d-cmb” refers to the Stop-word *n*-grams overlap method with N-grams of length [1-5] (5 features) and similarity measures Jaccard and Dice, respectively. “we-w2v-ct-ae” refers to the custom trained mono-lingual Word2Vec model with averaged embeddings method. Likewise, “we-w2v-ct-wae” and “we-w2v-ct-wme” refers to the custom-trained mono-lingual Word2Vec model with weighted average embeddings and weighted maximum embeddings methods. “se-laser-pt” refers to the results reported by pre-trained mono-lingual LASER sentence embeddings method. “Lo-sm-ss-cmb” refers to the combinations of lexical overlap, string matching, and Structural similarity methods. Similarly, “we-se-cmb” refers to the combination of mono-lingual word and sentence embeddings methods, and lastly, “all-methods-comb” refers to the experiments performed by combining all variants of all methods used in the study.

5.3 Evaluation methodology

To differentiate between multiple levels of text reuse, the problem is approached as a supervised text document classification task. The prime objective of the task is to see whether it is possible to automatically differentiate between the source and derived text at the document level and further understand which method(s) performs best.

Two variations of the task are used: (1) binary classification and (2) ternary classification. In the first case, the “Wholly Derived” (672 instances) and “Partially Derived” (888 instances) text documents are combined to make the “Derived” class (1,560 instances) and the “Non Derived” text documents remain part of the “Non Derived” class (697 instances). In the second case, the target is to distinguish between three levels of text reuse i.e., “Wholly Derived”, “Partially Derived”, and “Non Derived” classes.

For the set of experiments, the performance of a number of ML classifiers is investigated i.e., (1) Naïve Bayes, (2) Random Forest, (3) J48, (4) Support Vector Machine, (5) Multilayer Perceptron, and (6) Logistic Regression. All of these classifiers take numeric features as inputs and therefore are suitable for the experiments performed on the TREU corpus. Similarity scores generated by applying various methods (Section 5.1) are used as input feature(s) for the classifiers. Python’s Scikit-learn 0.2321 [Pedregosa et al., 2011] based implementation of all the classifiers, with their default parameter settings, is used. 10-fold cross-validation is applied to better estimate the performance of the methods used in the study. The evaluation results are computed for both binary and ternary classes and reported using the weighted average F_1 (equation 2.21) score.

The “Classifier” columns list the Machine Learning (ML) classifiers which produced the highest score among all the classifiers used in this study. “nb” is used as short for Naïve Bayes, “rf” as Random Forest, “mlp” as Multilayer Perceptron, “lr” as Logist Regression.

6 RESULTS AND ANALYSIS

Table 4 shows weighted average F_1 scores obtained using various cross-lingual text reuse detection methods for binary and ternary classification tasks respectively.

Overall, the best results for both classification tasks are obtained using the “all-methodscmb” method (F_1 = 0.66 ternaries, F_1 = 0.78 binary) which shows that combining a range of features from different methods helps discriminate between various levels of cross-lingual text reuse in the TREU corpus. It can be noted that these results are not very high and need further improvement. This highlights the fact that the detection of cross-lingual (English-Urdu) text reuse at the document level is a challenging task.

Table 4 also shows results for binary classification, as expected, all the results of the binary classification task are higher than the ternary classification task. This indicates that cross-lingual (English-Urdu) text reuse detection at the document level is easier between two classes than in three classes. Overall, the results corroborate

Table 4. Weighted average F_1 scores obtained by applying different variants of T+MA method on the TREU Corpus

Method	Ternary Classification	Classifier	Method	Binary Classification	Classifier
Lexical overlap					
lo-wno-d-cmb	0.60	mlp	lo-wno-j-cmb	0.74	lr
lo-vsm-bow	0.50	lr	lo-vsm-bow	0.68	mlp
lo-vsm-c4g	0.51	rf	lo-vsm-c5g	0.69	mlp
String matching					
sm-lcs	0.56	rf	sm-lcs	0.73	nb
sm-gst-cmb	0.57	lr	sm-gst-cmb	0.74	mlp
Structural similarity					
ss-sno-j-cmb	0.43	mlp	ss-sno-d-cmb	0.61	rf
Mono-lingual word embeddings					
we-w2v-ct-ae	0.47	rf	we-w2v-ct-ae	0.64	rf
we-w2v-ct-wae	0.47	mlp	we-w2v-ct-wae	0.64	nb
we-w2v-ct-wme	0.58	lr	we-w2v-ct-wme	0.75	rf
Mono-lingual sentence embeddings					
se-laser-pt	0.47	mlp	se-laser-pt	0.67	j48
Combination of methods					
lo-sm-ss-cmb	0.65	rf	lo-sm-ss-cmb	0.76	rf
we-se-cmb	0.61	lr	we-se-cmb	0.75	lr
all-methods-cmb	**0.66**	rf	all-methods-cmb	**0.78**	rf

with ternary classification task results. However, these results are still low considering the binary classification task is much simpler as it involves distinguishing between two classes that are relatively distinct. There could be several possible reasons for this. In binary classification, the WD (672 instances) and PD (888 instances) classes are combined to make the "Derived" class (total of 1,560 instances). This has resulted in class imbalance (1,560 Derived, 697 Non-Derived) which is one of the reasons for its low result. This shows that the T+MA method used in this study is effective in detecting cross-lingual (English-Urdu) text reuse at the document level to a larger extent. Moreover, these results further support the stance that the T+MA method performs better at longer texts (document level) but its performance declines on short cases of text reuse (at the sentence level) [5, 34].

For both classification tasks, the combination of different methods "lo-sm-sscmd", i.e., lexical overlap, string matching, and structural similarity combined (F_1 = 0.65 ternary, F_1 = 0.76 binary) and "we-se-cmd" mono-lingual word and sentence embedding combined (F_1 = 0.61 ternary, F_1 = 0.75 binary) improves performance. This indicates that using a set of features together has proven to be useful in the cross-lingual (English-Urdu) text reuse detection in the TREU corpus.

In terms of individual methods performance, for the ternary classification task, from lexical overlap, Word n-grams overlap performed better (F_1 = 0.60) than both variants of VSM method (F_1 = 0.50 VSM-BoW, F_1 = 0.51 VSM-CNG). Moreover, the best result is obtained using a combination of features [n = 1-5] (5 features) and the Dice similarity measure. This demonstrates that a simple overlap of word N-grams between source and derived text documents is a good indicator of cross-lingual text reuse and a combination of features has further increased its performance. This also shows that combining various lengths of N-grams together contributes better in identifying the cross-lingual (English-Urdu) reuse of text. Besides, the low result of VSM shows that it is better suited for IR or to find topical relevance between text documents instead of overlap between them.

For string matching, GST (F_1 = 0.57) reported comparatively better results than LCS (F_1 = 0.56). This indicates that both these methods are able to capture the word reordering in derived texts, however, could not beat the simple Word N-grams overlap method. It further shows that during the formulation of newspaper stories (derived text documents), the journalist(s) have not derived longer chunks from the news agency's report (source text document) in the TREU corpus. A possible reason for GST performing better than LCS is that it does not suffer from the block-move problem. Additionally, it produced the best result when the lengths of MML are combined (1-5) (5 features) for the classification task. This again highlights the advantage of using a combination of features over a single feature used in the T+MA experiments performed on the TREU corpus.

Stop-word *n*-grams overlap, the only structural similarity method, performed poorly and reported the lowest score (F_1 = 0.43) in all the methods used. The rationale is that it is more suitable for authorship attribution and intrinsic plagiarism detection tasks rather than text reuse detection.

Among monolingual word embeddings, custom trained Word2Vec model with weighted maximum embeddings performed significantly better (F_1 = 0.58) than the other two variants (F_1 = 0.47 averaged embeddings, F_1 = 0.47 weighted averaged embeddings). This shows the usefulness of the method which takes into account word-level similarities with IDF weighting instead of averaging individual word vectors. Moreover, among the three-word embeddings models used, Word2Vec has outperformed GloVe and fastText. Furthermore, it is worth noting that methods based on custom-trained word embeddings have consistently performed better than the pre-trained ones. The most probable reason is that the pre-trained word embeddings use Google News, Common Crawl, Wikipedia, etc. for training whereas custom word embeddings are trained on domain-specific text, i.e. PEN corpus. Consequently, custom-trained word embeddings are less likely to suffer from Out-Of-Vocabulary (OOV) words. Moreover, using domain-specific data, the models could learn representations of words better and ultimately perform better in the downstream task.

For monolingual sentence embeddings, LASER (F_1 = 0.67) has reported better results than others (Sent2Vec, InferSent, and Universal Sentence Encoder). There seem to be two possible reasons, 1) the model is trained on a large corpus (221M sentences), and 2) it supports biLSTM-based recurrent and deeper architecture. Thereby, it captures the syntactic and semantic properties of a sentence (text) better, which has helped in detecting the similarity between two texts. On the other hand, both Universal Sentence Encoder and Sent2Vec use uni- or bigrams with averaging to produce sentence embeddings, hence, could not produce good results. It is worth mentioning here that the pre-trained LASER model (encoder) is trained on different domain data (Europarl [24], United Nations corpus [18], etc.) than the TREU corpus (journalism), hence its performance could not surpass Word embedding based methods.

Regarding individual method results, for lexical, similar to ternary classification, Word *n*-grams overlap (F_1 = 0.74) outperformed VSM-BoW (F_1 = 0.68) and VSM-CNG (F_1 = 0.69). Once again the best result is obtained using a combination of N-gram features [1-5] (5 features) and the Jaccard similarity measure. This shows that the combination of features is helpful in improving performance even in the binary classification task.

The results of the string matching methods show a similar pattern to that of the ternary classification task. GST (F_1 = 0.74) performed slightly better than the LCS (F_1 = 0.73) method. Again it emphasizes the strength of GST which can detect the transposition of tokens (words) better than LCS. Furthermore, the result is obtained by combining MML length [1-5] which highlights that the classifier is better suited for the combination of features.

As expected, and similar to ternary classification, the structural similarity method Stop-word *n*-grams overlap reported the lowest result (F_1 = 0.61). This indicates that it is not an appropriate method to use for the CLTRD task on the TREU corpus.

The performance of various word embedding methods also shows a similar trend to that of the ternary classification task. The best result is obtained using a custom trained Word2vec model and using the weighted maximum embedding method (F_1 = 0.75). It is noteworthy that, for the binary classification task, the method has performed better than all the other distinct methods used in the study. This shows that the method is able to capture the semantic word-level overlap better between source and derived text documents than averaging of word vectors.

For sentence embedding, once again, due to its recurrent neural network architecture (BiLSTM) and having been trained on a large data set (221M sentences), LASER reported the highest result (F_1 = 0.67) among others. Regarding classifiers, in the majority of the cases, RF performed better than the others. Moreover, the highest results for both classification tasks (F_1 = 0.78 for binary and F_1 = 0.66 for ternary) are also reported using RF. This shows that the RF classifier is more appropriate to use for the cross-lingual experiments performed using various T+MA methods on the TREU corpus.

7 CONCLUSION AND FUTURE WORK

This paper presents a large CLTR corpus at the document level for the English-Urdu language pair. The proposed corpus contains real cases of CLTR which are manually annotated at three levels of rewrite (Wholly Derived = 672, Partially Derived = 888, Non Derived = 697). To demonstrate how our proposed corpus can be used for the development, evaluation, and comparison of CLTRD methods for English-Urdu language pair, we applied various Translation plus Monolingual Analysis-based methods on our proposed corpus. For binary classification, best results are obtained (F_1 = 0.78) and (F_1 = 0.66) using a combination of all T+MA based methods for both binary and ternary classification tasks. In the future, we plan to explore and apply other CLTRD methods to the TREU corpus.

REFERENCES

[1] Asan Agibetov, Kathrin Blagec, Hong Xu, and Matthias Samwald. 2018. Fast and Scalable Neural Embedding Models for Biomedical Sentence Classification. *BMC Bioinformatics* 19, 1 (2018), 541–549.

[2] Alexis Allot, Qingyu Chen, Sun Kim, Roberto Vera Alvarez, Donald C Comeau, W John Wilbur, and Zhiyong Lu. 2019. LitSense: Making Sense of Biomedical Literature at Sentence Level. *Nucleic Acids Research* 47, 1 (2019), 594–599.

[3] Mikel Artetxe and Holger Schwenk. 2018. Massively multilingual sentence embeddings for zero-shot cross-lingual transfer and beyond. *arXiv preprint arXiv:1812.10464* (2018).

[4] Man Bai, Xu Han, Haoran Jia, Cong Wang, and Yawei Sun. 2018. Transfer Pretrained Sentence Encoder to Sentiment Classification. In *IEEE 3rd International Conference on Data Science in Cyberspace*. 423–427.

[5] Alberto Barrón-Cedeño, Parth Gupta, and Paolo Rosso. 2013. Methods for cross-language plagiarism detection. *Knowledge-Based Systems* 50 (2013), 211–217.

[6] Steven Bird, Ewan Klein, and Edward Loper. 2009. *Natural Language Processing with Python: Analyzing Text with the Natural Language Toolkit.* Vol. 1. O'Reilly Media, Inc.

[7] Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. 2017. Enriching Word Vectors with Subword Information. *Transactions of the Association for Computational Linguistics* 5, 1 (2017), 135–146.

[8] Samuel R. Bowman, Gabor Angeli, Christopher Potts, and Christopher D. Manning. 2015. A Large Annotated Corpus for Learning Natural Language Inference. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*.

[9] Daniel Cer, Yinfei Yang, Sheng yi Kong, Nan Hua, Nicole Limtiaco, Rhomni St. John, Noah Constant, Mario Guajardo-Cespedes, Steve Yuan, Chris Tar, Yun-Hsuan Sung, Brian Strope, and Ray Kurzweil. 2018. Universal Sentence Encoder. arXiv:1803.11175 [cs.CL]

[10] Eunsol Choi, He He, Mohit Iyyer, Mark Yatskar, Wen-tau Yih, Yejin Choi, Percy Liang, and Luke Zettlemoyer. 2018. QuAC: Question Answering in Context. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*. 2174–2184.

[11] Paul Clough. 2003. *Measuring Text Reuse.* PhD Dissertation. University of Sheffield, UK.

[12] Paul Clough, Rob Gaizauskas, Scott Piao, and Yorick Wilks. 2002. METER: MEasuring TExt Reuse. In *Proceedings of the 40th Annual Meeting of the Association of Computational Linguistics*. 152–159.

[13] Jacob Cohen. 1960. A Coefficient of Agreement for Nominal Scales. *Educational and Psychological Measurement* 20, 1 (1960), 37–46.

[14] Alexis Conneau and Douwe Kiela. 2018. SentEval: An Evaluation Toolkit for Universal Sentence Representations. In *Proceedings of the 11th International Conference on Language Resources and Evaluation.*

[15] Alexis Conneau, Douwe Kiela, Holger Schwenk, Loïc Barrault, and Antoine Bordes. 2017. Supervised Learning of Universal Sentence Representations from Natural Language Inference Data. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*. 670–680.

[16] Divyanshu Daiya and Anukarsh Singh. 2018. Using Statistical and Semantic Models for Multi-Document Summarization. In *Proceedings of the 30th Conference on Computational Linguistics and Speech Processing*. 169–183.

[17] Bill Dolan and Chris Brockett. 2005. Automatically constructing a corpus of sentential paraphrases. In *Third International Workshop on Paraphrasing (IWP2005).*

[18] Andreas Eisele and Yu Chen. 2010. MultiUN: A Multilingual Corpus from United Nation Documents. In *Proceedings of the 7th International Conference on Language Resources and Evaluation.*

[19] Jeremy Ferrero, Laurent Besacier, Didier Schwab, and Frederic Agnes. 2017. Using word embedding for cross-language plagiarism detection. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers*. Association for Computational Linguistics, Valencia, Spain, 415–421. https://www.aclweb.org/anthology/E17-2066

[20] Bela Gipp, Norman Meuschke, and Joeran Beel. 2011. Comparative Evaluation of Text- and Citation-based Plagiarism Detection Approaches using GuttenPlag. In *Proceedings of the 11th Annual International ACM/IEEE Joint Conference on Digital Libraries*. 255–258.

[21] Prakhar Gupta, Matteo Pagliardini, and Martin Jaggi. 2019. Better Word Embeddings by Disentangling Contextual n-Gram Information. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics*. 933–939.

[22] Israr Haneef, Rao Muhammad Adeel Nawab, Ehsan Ullah Munir, and Imran Sarwar Bajwa. 2019. Design and development of a large cross-lingual plagiarism corpus for Urdu-English language pair. *Scientific Programming* 2019 (2019).

[23] Tom Kenter and Maarten De Rijke. 2015. Short text similarity with word embeddings. In *Proceedings of the 24th ACM International on Conference on Information and Knowledge Management*. 1411–1420.

[24] Philipp Koehn. 2005. Europarl: A Parallel Corpus for Statistical Machine Translation. In *Proceedings of the 10th Machine Translation Summit*. 79–86.

[25] Ji-Ung Lee, Steffen Eger, Johannes Daxenberger, and Iryna Gurevych. 2017. Deep Learning for Aspect Based Sentiment Detection. In *Proceedings of the GSCL GermEval Shared Task on Aspect-based Sentiment in Social Media Customer Feedback*. 22–29.

[26] Chenliang Li, Haoran Wang, Zhiqian Zhang, Aixin Sun, and Zongyang Ma. 2016. Topic Modeling for Short Texts with Auxiliary Word Embeddings. In *Proceedings of the 39th International ACM SIGIR conference on Research and Development in Information Retrieval.* 165–174.

[27] Roger Logue. 2004. Plagiarism: The Internet Makes it Easy. *Nursing Standard* 18, 51 (2004).

[28] Christopher Manning, Mihai Surdeanu, John Bauer, Jenny Finkel, Steven Bethard, and David McClosky. 2014. The Stanford CoreNLP Natural Language Processing Toolkit. In *Proceedings of 52nd Annual Meeting of the Association for Computational Linguistics: System Demonstrations*. 55–60.

[29] Fanqing Meng, Wenpeng Lu, Yuteng Zhang, Jinyong Cheng, Yuehan Du, and Shuwang Han. 2017. Qlut at SemEval-2017 Task 1: Semantic Textual Similarity based on Word Embeddings. In *Proceedings of the 11th International Workshop on Semantic Evaluation*. 150–153.

[30] Tomáš Mikolov, Wen-tau Yih, and Geoffrey Zweig. 2013. Linguistic regularities in continuous space word representations. In *Proceedings of the 2013 conference of the north american chapter of the association for computational linguistics: Human language technologies*. 746–751.

[31] Iqra Muneer and Rao Muhammad Adeel Nawab. 2021. Cross-lingual Text Reuse Detection Using Translation Plus Monolingual Analysis for English-Urdu Language Pair. *Transactions on Asian and Low-Resource Language Information Processing* 21, 2 (2021), 1–18.

[32] Iqra Muneer and Rao Muhammad Adeel Nawab. 2022. Cross-Lingual Text Reuse Detection at sentence level for English-Urdu language pair. *Computer Speech & Language* (2022), 101381. https://doi.org/10.1016/j.csl.2022.101381

[33] Iqra Muneer and Rao Muhammad Adeel Nawab. 2022. Develop corpora and methods for cross-lingual text reuse detection for English Urdu language pair at lexical, syntactical, and phrasal levels. *Language Resources and Evaluation* (2022), 1–28.

[34] Iqra Muneer, Muhammad Sharjeel, Muntaha Iqbal, Rao Muhammad Adeel Nawab, and Paul Rayson. 2019. CLEU-A cross-language English-Urdu corpus and benchmark for text reuse experiments. *Journal of the Association for Information Science and Technology* 70, 7 (2019), 729–741.

[35] Matteo Pagliardini, Prakhar Gupta, and Martin Jaggi. 2018. Unsupervised Learning of Sentence Embeddings Using Compositional n-Gram Features. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics.* 528–540.

[36] Jeffrey Pennington, Richard Socher, and Christopher D Manning. 2014. Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*. 1532–1543.

[37] Radim Řehůřek and Petr Sojka. 2010. Software Framework for Topic Modelling with Large Corpora. In *Proceedings of the LREC 2010 Workshop on New Challenges for NLP Frameworks*. 45–50.

[38] Sara Sameen, Muhammad Sharjeel, Rao Muhammad Adeel Nawab, Paul Rayson, and Iqra Muneer. 2017. Measuring short text reuse for the Urdu language. *IEEE Access* 6 (2017), 7412–7421.

[39] Yunita Sari, Andreas Vlachos, and Mark Stevenson. 2017. Continuous n-gram Representations for Authorship Attribution. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics*. 267–273.

[40] Muhammad Sharjeel, Rao Muhammad Adeel Nawab, and Paul Rayson. 2017. COUNTER: corpus of Urdu news text reuse. *Language resources and evaluation* 51, 3 (2017), 777–803.

[41] Ivan Vulić and Marie-Francine Moens. 2015. Monolingual and Cross-lingual Information Retrieval Models based on (Bilingual) Word Embeddings. In *Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval.* 363–372.

[42] Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel Bowman. 2018. GLUE: A Multi-Task Benchmark and Analysis Platform for Natural Language Understanding. In *Proceedings of the 2018 Empirical Methods in Natural Language Processing.*

353–355.

[43] Yorick Wilks. 2004. On the Ownership of Text. *Computers and the Humanities* 38, 2 (2004), 115–127.

[44] Michael J Wise. 1993. *Running Karp-Rabin Matching and Greedy String Tiling*. Vol. 1. University of Sydney.

[45] Yadollah Yaghoobzadeh, Katharina Kann, Timothy J. Hazen, Eneko Agirre, and Hinrich Schütze. 2019. Probing for Semantic Classes: Diagnosing the Meaning Content of Word Embeddings. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*. 5740–5753.

[46] Liang-Chih Yu, Jin Wang, K Robert Lai, and Xuejie Zhang. 2017. Refining Word Embeddings for Sentiment Analysis. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*. 534–539.

[47] Yukun Zhu, Ryan Kiros, Rich Zemel, Ruslan Salakhutdinov, Raquel Urtasun, Antonio Torralba, and Sanja Fidler. 2015. Aligning Books and Movies: Towards Story-like Visual Explanations by Watching Movies and Reading Books. In *Proceedings of the IEEE International Conference on Computer Vision*. 19–27.