MUTANT: A Multi-sentential Code-mixed Hinglish Dataset

Rahul Gupta
IIT Gandhinagar
Gandhinagar, Gujarat, India
gupta.rahul@iitgn.ac.in

Vivek Srivastava
TCS Research
Pune, Maharashtra, India
srivastava.vivek2@tcs.com

Mayank Singh
IIT Gandhinagar
Gandhinagar, Gujarat, India
singh.mayank@iitgn.ac.in

Abstract

The multi-sentential long sequence textual data unfolds several interesting research directions pertaining to natural language processing and generation. Though we observe several high-quality long-sequence datasets for English and other monolingual languages, there is no significant effort in building such resources for code-mixed languages such as Hinglish (code-mixing of Hindi-English). In this paper, we propose a novel task of identifying multi-sentential code-mixed text (MCT) from multilingual articles. As a use case, we leverage multilingual articles from two different data sources and build a first-of-its-kind multi-sentential code-mixed Hinglish dataset i.e., MUTANT. We propose a token-level language-aware pipeline and extend the existing metrics measuring the degree of code-mixing to a multi-sentential framework and automatically identify MCT in the multilingual articles. The MUTANT dataset comprises 67k articles with 85k identified Hinglish MCTs. To facilitate future research directions, we will make the dataset and the code publicly available upon publication.

Figure 1: Example MCT and the corresponding article's title form two multilingual data sources: (A) Dainik Jagran news article and (B) Man-ki-baat speech transcript. We color code the tokens as: English, Hindi, and language independent.

1 Introduction

Over the years, we have seen enormous downstream applications of multi-sentential datasets in the areas such as question-answering (Joshi et al., 2017; Tapaswi et al., 2016), summarization (Sharma et al., 2019; Cachola et al., 2020), machine translation (Bao et al., 2021), etc. The existing state-of-the-art methods prove challenging to scale effectively and efficiently on multi-sentential long sequence text (Ainslie et al., 2020), which unplugs several exciting research avenues. Unfortunately, to a large extent, the majority of the research on multi-sentential data is dominated by a few popular monolingual languages such as English, Chinese, and Spanish. Due to this, code-mixed languages (among other low-resource and under-explored languages) suffer from non-existent works in the aforementioned areas of interest.

We posit that due to several inherent challenges, the NLP community hold back on building multi-sentential datasets for the low-resource and code-mixed languages. One of the most significant bottlenecks in building such resources is the unavailability of MCT on traditional and widely popular data sources such as social media platforms where the short-length and noisy code-mixed text is available in abundance. It presents several challenges such as the difficulty in curating a large-scale multi-sentential dataset at ease. Another major challenge is the lack of metrics to measure the degree of code-mixing in the multi-sentential framework. The existing metrics such as code-mixing index (Das and Gambäck, 2014) and multilingual-index (Barnett et al., 2000) already suffers from major limitations (Srivastava and Singh, 2021a) in the short-length text format. In such a scenario, it gets mystifying

744

Dataset	Task(s)	Data Source(s)	# Instances	Avg Tokens	Avg Sentences	Retrieval
(Srivastava and Singh, 2020)	Machine Translation	Social media posts on Twitter & Facebook	13738	13	1.04	Automatic
(Khanuja et al., 2020)	Natural Language Inference	Hindi Bollywood movie transcripts	2240	87	7.15	Automatic
(Mehnaz et al., 2021)	Dialogue Summarization	Manual translation of dialogues and summaries from (Gliwa et al., 2019)	6830	31	7.85	-
(Srivastava and Singh, 2021b)	Generation & Evaluation	IIT-B En-Hi parallel corpus (Kunchukuttan et al., 2018)	1974	20	1.05	-
MUTANT	Summarization	Speech transcripts, press releases, and news articles	84937	159	10.23	Manual + Automatic

Table 1: Comparison of the MUTANT dataset with the currently available datasets in the Hinglish language.

to build a retrieval pipeline to identify MCT and we need to depend heavily on the expertise of human annotators which is a time and cost-demanding exercise. In this work, we address both of these challenges. As a representative use case, we base our work on Hinglish, a popular code-mixed language in the Indian subcontinent. But the insights from our exploration could be extended to other code-mixed language pairs.

To address the first challenge, we identify two non-traditional multilingual data sources[1] i.e., political speeches and press releases along with Hindi daily news articles (discussed in detail in Section 3). Figure 1 shows example Hinglish MCTs from two multilingual data sources. To address the second challenge, we propose a token-level language-aware pipeline and extend a widely popular metric (i.e., code-mixing index) measuring the degree of code-mixing in a multi-sentential framework. We demonstrate the effectiveness of the proposed pipeline with a minimal task-specific annotation which significantly reduces the overall human effort (discussed in detail in Section 4).

Eventually, we build a novel multi-sentential dataset for the Hinglish language with 85k MCTs identified from 67k articles. In Table 1, we compare MUTANT with four other Hinglish datasets (Srivastava and Singh, 2020; Khanuja et al., 2020; Mehnaz et al., 2021; Srivastava and Singh, 2021b) proposed for a variety of tasks such as machine translation, natural language inference, generation, and evaluation. The MUTANT dataset has a significantly higher average number of sentences along with longer MCT (high average number of tokens). Alongside, the dataset notably consists of a higher number of data instances which is a rarity for the code-mixed datasets (Srivastava and Singh, 2021a).

[1] these data sources have not been actively employed in building datasets for the code-mixed languages

2 Multi-sentential Code-mixed Text Span (MCT)

Due to the absence of a formal definition of MCT in the literature, we propose and use the following definition of MCT throughout this work:

MCT: Consider a multilingual article $A = \{s_1, s_2, ..., s_n\}$ consisting of n sentences denoted by s_i where $i \in [1, n]$. A unique non-overlapping MCT M_p in A is a chunk of $m > 1$ consecutive sentences i.e. $M_p = \{s_k, s_{k+1}, ..., s_{k+m-1}\}$. M_p should satisfy the following two properties:

1. $P1$: At least one s_{k+j} in M_p should be code-mixed. Trivially, at most m-1 s_{k+j} in M_p could be monolingual. Here, $j \in [0, m\text{-}1]$.
2. $P2$: s_k in M_p is either the first sentence of the article or preceded by a line break. Likewise, s_{k+m-1} is either the last sentence of the article or succeeded by a line break.

It should be noted that an article A can have multiple non-overlapping unique MCTs i.e. $A = \{M_1, M_2, ..., M_q\}$ where $q \geq 0$.

3 Multilingual and Multi-sentential Data Sources

Over the years, we observe several interesting and diverse code-mixed data sources such as Twitter, Facebook, movie transcripts, etc. Social media sites have acted as the cornerstone of the code-mixed data collection pipelines due to the ease of availability of large-scale data. Nonetheless, they present several challenges such as noisy data, short text, abusive, and multimodal data. Given the requirements of *MUTANT* (i.e. multi-sentential and high-quality data), we refrain from using social media sites in this work. Here, we focus on two major data sources:

3.1 Political speeches and press releases

Here, we scrape data from five different web sources. Collectively, we denote this data source as D_{speech}.

Aam Aadmi Party press releases (AAP): We scrape the press releases from the official website of Aam Aadmi Party[2]. We have scraped 320 Hindi press releases from their website. The website contains all the press releases in the last five years starting from June 2017.

Indian National Congress speeches (INC): The official website of the INC stores some of the speeches by major INC political leaders. We have extracted 112 of these speeches from their official website[3]. The timeline for the scraped speeches is between August 2018 to March 2022.

Man-ki-baat (MKB): Man-ki-baat is a radio program hosted by the Indian prime minister Narendra Modi where he periodically addresses the people of the nation. The MKB website[4] stores the official transcripts in Hindi and English languages. We have extracted the transcripts of 67 of these programs between December 2015 to December 2021.

Press Information Bureau (PIB): The Press Information Bureau houses the official press releases from all Indian government ministries including President's office, the Prime Minister's office, Election Commission, etc. We have extracted 30283 articles from the PIB website[5]. The timeline for these articles is from June 2017 to March 2022.

PM speech (PMS): Majority of the Indian Prime Minister speeches (different from MKB speeches) are stored digitally on the PM India website[6]. We have extracted 694 of these speeches that are recorded between November 2016 to October 2021.

3.2 Hindi news articles

Here, we scrape data from two major Hindi news daily websites. Collectively, we denote this data source as D_{news}.

Dainik Bhaskar (DB): Dainik Bhaskar is one of the most popular Hindi newspapers in India. It is ranked 4th in the world by circulation according to

World Press Trends 2016[7]. They have digitized the daily newspapers on their website[8]. Articles on DB website have been divided into many categories such as '*Entertainment*' and '*Sports*'. We have extracted 115324 articles uploaded on the website between February 2019 to May 2022. In Table 2, we present the category-wise distribution of the articles scraped from the DB website.

Category	DB	DJ
Business	16012	4203
Entertainment	18498	52173
Featured	5536	19373
Lifestyle	12189	-
Miscellaneous	20221	-
National	18615	160005
Politics	-	33604
Sports	9950	-
World	14303	42478
Total	115324	311836

Table 2: Number of articles in various news categories in the DB and DJ datasets.

Dainik Jagran (DJ): Dainik Jagran is another popular Indian Hindi newspaper. According to World Press Trends 2016, DJ is ranked 5th in the world by circulation. Similar to the DB website, they have also created a repository of articles on their official website[9]. Here, we extract 311836 of these articles from the website that were uploaded between April 2013 to May 2022. In Table 2, we present the category-wise distribution of the articles scraped from the DJ website.

4 Experimental Setup

Problem definition: Given a multilingual article A comprising of q multi-sentential text spans (MST) i.e. $A = \{M_1, M_2, ..., M_q\}$, we predict a binary outcome L_{CM} for each MST M_i i.e. $L(A) = \{L_{CM}^{M_1}, L_{CM}^{M_2}, ..., L_{CM}^{M_q},\}$. $L_{CM}^{M_i} = 1$, if M_i is code-mixed, otherwise 0. In a nutshell, a code-mixed MST M_i is a MCT and it satisfies the properties $P1$ and $P2$ (ref. §2).

Figure 2 shows the architecture of the MCT identification pipeline. Next, we discuss the various components of this pipeline in detail.

4.1 Token-level language annotation (TLA)

We exploit the token-level language information to identify MCT given a multilingual article A.
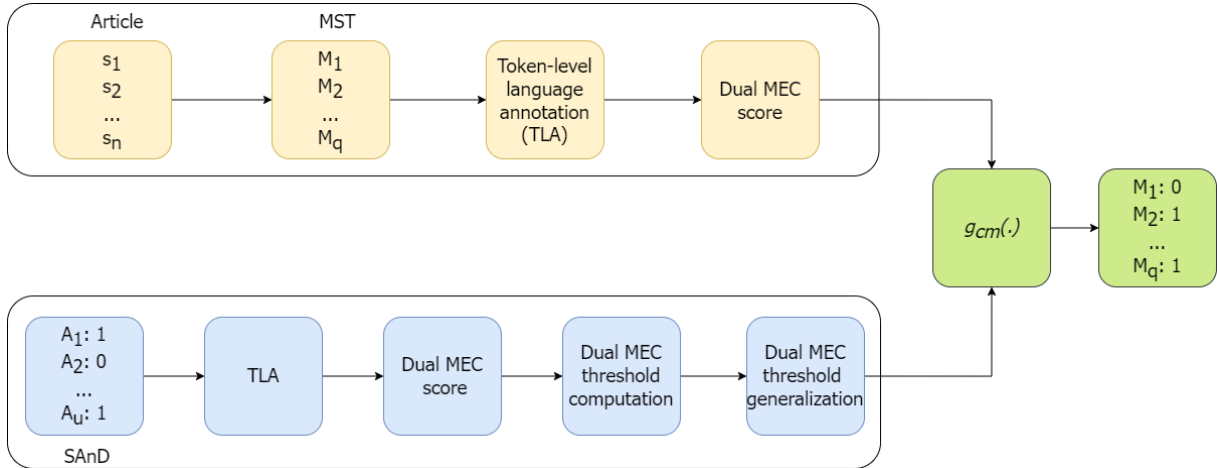
Figure 2: Architecture of MCT identification pipeline.

We annotate the words in A using a code-mixed language identification tool. Specifically, we use L3Cube-HingLID (Nayak and Joshi, 2022) for this task. A word $w_i \in A$ can take either of the three language tags from the set $\{English, Hindi, Other\}$. Given that L3Cube-HingLID works only on the Roman script text, we use a Devanagari to Roman script transliteration tool[10] for the tokens written in Devanagari script. In Table 3, we report the percentage of $Hindi$ and $English$ tokens. With an exception of the AAP dataset, $Hindi$ is the predominant language in all the data sources.

	Articles	AW	AC	%H	%E
AAP	320	1129	6033	53.97	45.09
INC	112	2312	10691	63.83	33.12
MKB	67	4151	20706	77.17	22.41
PIB	30283	525	3015	80.96	17.59
PMS	694	2591	13400	79.02	20.45
DB	115324	382	1977	80.22	18.25
DJ	311836	391	2037	79.28	19.60
D_{speech}	31476	590	3339	79.97	18.65
D_{news}	427160	388	2020	80.18	18.51
D_{speech} $+ D_{news}$	458636	401	589	80.05	18.54

Table 3: Distribution of the scraped articles from various data sources. AW: average number of words. AC: average number of characters. %E: percentage of English tokens. %H: percentage of Hindi tokens.

4.2 Code-Mixing Index (CMI)

In the literature, we observe several metrics that has been proposed to measure the degree of code-mixing in text such as code-mixing index (CMI,

(Das and Gambäck, 2014)), multilingual-index (M-index, (Barnett et al., 2000)) and integration-index (I-index, (Guzmán et al., 2017)). Each of these metrics has its own merits and limitations (Srivastava and Singh, 2021a). In this work, we use the most widely used CMI metric due to the ease of interpretation and the suitability for the task. CMI, by definition, measures the degree of code-mixing in a text as:

$$CMI = \begin{cases} 100 * [1 - \frac{max(w_i)}{n-u}] & n > u \\ 0 & n = u \end{cases} \quad (1)$$

Here, w_i is the number of words of the language i, $max\{w_i\}$ represents the number of words of the most prominent language, n is the total number of tokens, u represents the number of language-independent tokens (such as named entities, abbreviations, mentions, and hashtags). The CMI score ranges from 0 to 100. A low CMI score suggests the prevalence of only one language in the text whereas a high CMI score indicates a high degree of code-mixing.

4.3 Small annotated dataset (SAnD)

We create a small manually annotated dataset comprising all seven data sources. The objective of the annotation is to assign a binary label to each MST such that we can identify if the MST is code-mixed or not from the assigned label.

More formally, $SAnD = \{A_1: l_1, A_2: l_2, ..., A_u: l_u\}$, represents u manually annotated MST[11] where $l_i \in \{0,1\} \ \forall i \in [1,u]$. Here, $l_i=1$, if A_i is code-mixed, otherwise 0.

---

[10]https://github.com/ritwikmishra/devanagari-to-roman-script-transliteration

[11]For distinctive representation, we denote MST in $SAnD$ with A instead of M.

	Articles	MST Total	MST Hing	E/H
AAP	5	6	2	4
INC	3	69	5	64
MKB	3	66	25	41
PIB	47	62	27	35
PMS	2	36	13	23
DB	30	207	48	159
DJ	30	122	28	94
D_{speech}	60	239	72	167
D_{news}	60	329	76	253
D_{speech} + D_{news}	120	568	148	420

Table 4: $SAnD$ dataset statistics. Hing: Hinglish, E/H: English/Hindi.

For this annotation task, we have selected a small number of articles (60 each from D_{speech} and D_{news}) randomly from the scraped articles. We leave it to the judgment of the annotator to decide if a sentence (and subsequently the MST) is code-mixed or not. The annotator has expert-level proficiency in Hindi, English, and Hinglish languages. In Table 4, we show the distribution of the annotated articles for each data source. In total, we annotate 120 articles and 568 MST where we identify 121 MST (21.3%) as code-mixed.

4.4 Estimating multilinguality

Though CMI is widely used in numerous previous works, we couldn't find any discussion on the ideal CMI score thresholding criteria to identify a good code-mixed text. The problem becomes even more challenging when we use the CMI metric in a multi-sentential framework along with constraints $P1$ and $P2$ (ref §2). Various works (Khanuja et al., 2020) have used empirically identified CMI thresholds to measure the degree of code-mixing in the text. But, we couldn't find any experimental justification for their findings.

Dual MEC score: Here, we propose a novel adoption of the CMI metric in a constrained multi-sentential framework. For MST M_p with k sentences, we compute the scores for dual multilinguality estimation criteria (MEC) as:

1. Sentence-level CMI (CMI): We compute $CMI(s_i)$ for the sentence $s_i \in M_p$ using the language-information of all the words in s_i and the formulation given in 1.
2. Multilinguality ratio (MR): We compute C_{MR} for the MST M_p as:

$$MR(M_p) = \frac{N_{cm}}{k} \qquad (2)$$

Here, N_{cm} and k are the number of code-mixed and total sentences in M_p respectively.
Figure 3 shows the mean and standard deviation of dual MEC scores on seven different data sources.
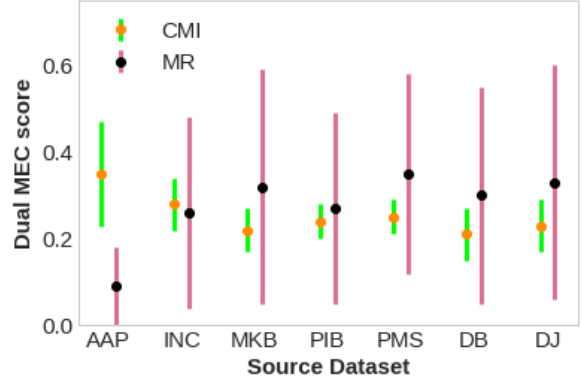
Figure 3: The mean and standard deviation of the dual MEC score for different data sources. The CMI score is scaled between 0 to 1.

Formulation: We identify if the sentence s_i is code-mixed or monolingual using $CMI(s_i)$ score as:

$$f_{cm}(s_i) = \begin{cases} 1, & CMI(s_i) > \alpha \\ 0, & otherwise \end{cases} \qquad (3)$$

Here, $\alpha \in [0, 100]$ is the sentence-level CMI score threshold and $f_{cm}(.)$ estimates the code-mixing status (1 being code-mixed and 0 being monolingual) of the sentence under consideration. Using 3, we compute N_{cm} as:

$$N_{cm} = \Sigma_{i=1}^{k} f_{cm}(s_i) \qquad (4)$$

Using 2 and 4, we compute $MR(M_p)$ as:

$$MR(M_p) = \frac{\Sigma_{i=1}^{k} f_{cm}(s_i)}{k} \qquad (5)$$

We formulate the following function to identify if MST M_p with k sentences is code-mixed:

$$g_{cm}(M_p) = \begin{cases} 1, & MR(M_p) > \beta \\ 0, & otherwise \end{cases} \qquad (6)$$

Here, $\beta \in [0, 1]$ is the multilinguality ratio threshold and $g_{cm}(.)$ estimates the code-mixing status (1 being code-mixed and 0 being monolingual) of the MST under consideration.

4.5 Dual MEC threshold computation

The dual MEC formulation helps us to identify the MCT in a constrained setting by jointly modeling the sentence-level and MST-level multilinguality information. As discussed in Section 4.4, the ideal thresholds α and β are a conundrum that needs further exploration. Here, we propose to use the $SAnD$ dataset to identify the dual MEC thresholds (α and β). Algorithm 1 shows the procedure to compute the thresholds. The algorithm takes $SAnD$ dataset D with u labeled MST. We represent the parameter search space for α and β with α_{cand} and β_{cand} respectively. α_{cand} ranges from α_{low} to α_{high} with a step-size of α_{step} whereas β_{cand} ranges from β_{low} to β_{high} with a step-size of β_{step}. Based on our empirical observation, we set (α_{low}, α_{high}, α_{step}) with (0, 50, 1) and (β_{low}, β_{high}, β_{step}) with (0, 0.5, 0.025).

We perform the grid search on each threshold combination of (α_i, β_j) to identify the best combination. For each threshold combination, we identify the accuracy of identifying the MCT in D leveraging $f_{cm}(.)$ and $g_{cm}(.)$ formulations. We select the threshold combination with the highest accuracy as the final threshold (α and β). Table 5 shows the best-identified thresholds on various data sources of the $SAnD$ dataset. Figure 4 shows the mean and standard deviation of the accuracy on various dual MEC threshold combinations for different data sources.

	α	β	Accuracy(%)
AAP	25	0.35	100
INC	28	0.30	89
MKB	22	0.35	64
PIB	26	0.15	68
PMS	21	0.45	89
DB	18	0.40	72
DJ	28	0.40	79
D_{speech}	24	0.35	72
D_{news}	29	0.475	78
D_{speech} + D_{news}	29	0.45	75

Table 5: Best identified thresholds (α and β) along with the accuracy of identifying MCT on various data sources in the $SAnD$ dataset.
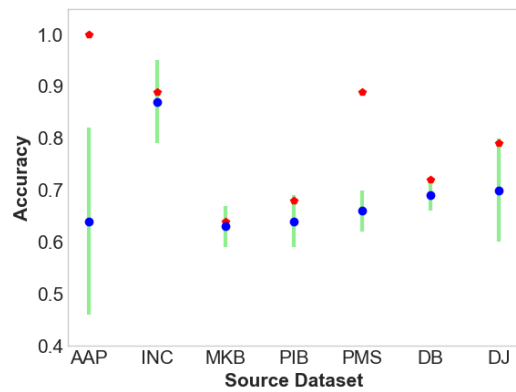
Figure 4: The mean and standard deviation of the accuracy on various dual MEC threshold combinations. The red dot corresponding to each data source indicates the accuracy against the best-identified thresholds.

4.6 Dual MEC threshold generalization

As evident from Table 5, the thresholds α and β vary across the data sources. So, it is important to identify which of these identified thresholds will result in a robust and stable performance across datasets. Here, we experiment with five dual MEC threshold generalisation techniques:

1. **Local Average (LA)**: For the data source D_i, we take the mean sentence-level CMI score and mean MR score as the dual MEC thresholds.

2. **Global Average (GA)**: For the data source D_i, we take the mean sentence-level CMI score and mean MR score of the corresponding category data-source (D_{speech} or D_{news}) as the dual MEC thresholds.

3. **Average of LA and GA (ALG)**: For the data source D_i, we take the average of LA and GA identified thresholds as the dual MEC thresholds.

4. **Single data source generalization (SDG)**: In

Algorithm 1 $compute_{\alpha,\beta}(D)$

Require: $D = \{A_1: l_1, A_2: l_2, ..., A_u: l_u\}$ where $A_i = \{s_1, s_2, ..., s_k\}$
Require: $\alpha_{cand} = [\alpha_{low}, \alpha_{low}+\alpha_{step}, ..., \alpha_{high}]$
Require: $\beta_{cand} = [\beta_{low}, \beta_{low}+\beta_{step}, ..., \beta_{high}]$
Require: $Accuracy = \{\}$
 1: **for** α_i in α_{cand} **do**
 2:     **for** β_j in β_{cand} **do**
 3:         $hits = 0$
 4:         **for** $A_p \in D$ **do**
 5:             $F_{cm} = f_{cm}(s_q) \; \forall \; s_q \in A_p$
 6:             Compute $g_{cm}(A_p)$ using F_{cm}
 7:             **if** $g_{cm}(A_p) == l_p$ **then**
 8:                 $hits = hits + 1$
 9:             **end if**
10:         **end for**
11:         $Accuracy[(\alpha_i, \beta_j)] = 100 * (hits/u)$
12:     **end for**
13: **end for**
14: $\alpha = max_{value}(Accuracy).key()[0]$
15: $\beta = max_{value}(Accuracy).key()[1]$
16: **return** α, β

---

this approach, we generalize the dual MEC thresholds identified locally on a single data source D_i (using Algorithm 1) to identify MCT globally on other data sources.

5. **Multi data source generalization (MDG)**: In this approach, we use the dual MEC threshold information from multiple sources and use the majority voting to identify the best thresholds. For the data source D_i, we use the thresholds identified on three data sources (using Algorithm 1), namely D_i, D_{speech} (if $D_i \in D_{speech}$, else ăD_{news}), and $D_{speech} + D_{news}$. We then make an independent prediction on each of the three thresholds and take majority voting for the final classification of M_p.

5 MUTANT: A Multi-sentential Code-mixed Hinglish Dataset

We evaluate the performance of MCT identification pipeline and the five dual MEC threshold generalization techniques using the three subsets of the $SAnD$ dataset: D_{speech}, D_{news}, and $D_{speech} + D_{news}$. We report the following metric scores on each of the seven data sources:

1. Accuracy: We compute accuracy as the ratio of the total correct prediction of MCT and non-MCT to the total number of MST. We multiply this ratio by 100 and report the accuracy percentage. A high accuracy % is preferred.

2. False MCT Rate (FMR): We define FMR as the ratio of incorrectly identified MCT to the total number of actual monolingual MST. We report the FMR% and a low FMR% is preferred.

3. Diversity@10 (D@10): We define D@10 as the percentage of articles in data source D_i having more than 10% correctly identified MCT. A high D@10 score is preferred.

We report the results in Tables 6, 7, 8. The mean-based threshold generalization techniques (LA, GA, and ALG) consistently show poor performance on all the metrics. Given the nature of the problem, we prefer a low rate of misidentification of mono-

	Accuracy					FMR					D@10				
	L	G	A	S	M	L	G	A	S	M	L	G	A	S	M
AAP	62	66	64	72	74	15	21	20	17	17	49	46	51	60	62
INC	63	66	64	73	74	17	21	20	16	12	49	46	51	59	59
MKB	61	66	62	69	72	28	21	26	22	18	51	46	48	68	70
PIB	62	66	64	67	72	24	21	24	30	17	53	46	55	73	74
PMS	67	66	64	71	74	17	21	23	20	16	51	46	53	67	69
DB	66	63	62	67	78	29	26	28	30	5	57	56	57	78	78
DJ	62	63	64	75	78	26	26	26	6	5	48	56	49	73	74

Table 6: Results on D_{speech} dataset. L: LA, G: GA, A: ALG, S: SDG, M: MDG.

	Accuracy					FMR					D@10				
	L	G	A	S	M	L	G	A	S	M	L	G	A	S	M
AAP	72	70	71	72	73	17	15	17	14	14	60	58	62	70	72
INC	69	70	71	73	73	14	15	15	9	7	58	58	58	65	66
MKB	66	70	68	70	72	25	15	21	21	15	73	58	71	79	80
PIB	68	70	68	70	73	23	15	22	29	14	73	58	71	79	80
PMS	61	70	69	74	73	14	15	18	14	12	63	58	63	71	69
DB	66	69	67	68	71	28	22	26	29	3	76	72	74	84	85
DJ	68	69	68	72	71	22	22	22	4	3	70	72	68	77	73

Table 7: Results on D_{news} dataset. L: LA, G: GA, A: ALG, S: SDG, M: MDG.

	Accuracy					FMR					D@10				
	L	G	A	S	M	L	G	A	S	M	L	G	A	S	M
AAP	69	70	69	73	74	12	15	15	13	13	55	60	57	65	66
INC	70	70	69	73	74	11	15	14	10	8	57	60	56	62	63
MKB	67	70	69	70	72	21	15	19	17	14	62	60	65	68	65
PIB	69	70	69	67	73	18	15	18	23	14	63	60	64	75	74
PMS	62	70	70	72	74	13	15	17	16	12	57	60	59	65	69
DB	67	68	67	67	75	23	19	22	24	4	64	62	62	76	75
DJ	68	68	69	74	75	19	19	19	5	4	57	62	62	71	74

Table 8: Results on $D_{speech}+D_{news}$ dataset. L: LA, G: GA, A: ALG, S: SDG, M: MDG.

lingual MST as the MCT and at the same time a high number of actual MCT should also be identified. MDG threshold generalization technique satisfies both conditions with low FMR and high accuracy on all the datasets. D@10 depicts if the threshold generalization technique is influenced by the presence of a few outliers in the dataset. SDG and MDG both show competitive results on the D@10 metric outperforming the mean-based threshold generalization techniques by a large margin. The constant poor performance of mean-based threshold generalization against SDG and MDG also shows the efficacy of the proposed threshold computation strategy (Algorithm 1).

Finally, to build the MUTANT dataset, we use the MCT identification pipeline with the MDG threshold generalization technique. Table 9 shows the statistics of the MUTANT dataset. To facilitate future work on this novel task of MCT identification, we will release the MUTANT dataset along with the initially scraped data from all the data sources and the annotated $SAnD$ dataset. The MUTANT dataset can be used for various tasks including but not limited to question-answering, text summarization and machine translation for Hinglish texts. This dataset could be used as a pre-training dataset to train efficient NLU models for various tasks on Hinglish data.

6 Analysis and Discussion

In this section, we qualitatively evaluate the *MUTANT* dataset by employing two human evaluators, different from the one used for the $SAnD$ to avoid any biases in the evaluation. Both evalua-

	A	M	M/A	Avg CMI			Avg Words			Avg Characters		
				A	M	H	A	M	H	A	M	H
AAP	30	32	1.07	33.0	35.2	21.1	1347	1263	16	6993	6556	63
INC	85	306	3.6	28.1	27.5	-	751	208	-	3368	935	-
MKB	58	243	4.19	20.1	22.4	-	1034	246	-	4843	1156	-
PIB	8473	8786	1.04	23.0	23.2	21.0	572	552	15	3139	3028	87
PMS	597	3909	6.55	25.8	24.7	26.4	952	145	13	4585	700	79
DB	12851	15433	1.20	21.0	21.2	20.2	107	89	24	528	440	123
DJ	44913	56228	1.25	22.2	22.3	21.6	146	117	16	734	586	82
D_{speech}	9243	13276	1.44	23.2	23.8	21.3	604	420	15	3258	2268	87
D_{news}	57764	71661	1.24	21.9	22.0	21.2	137	111	18	688	555	91
$D_{speech} + D_{news}$	67007	84937	1.27	22.0	22.3	21.2	201	159	17	1043	822	90

Table 9: MUTANT dataset statistics. A: Articles, M: MCT, and H: Headings. The INC and MKB datasets contain generic and very-low informative headlines and we do not include them in the final dataset.

	A	MST	CA		CKS	Acc	FMR	D@10
			Hing	E/H				
AAP	5	5	2	3	1.0	100	0	100
INC	5	82	10	67	0.76	88	10	80
MKB	5	119	23	80	0.67	75	25	80
PIB	5	5	2	3	1.0	80	0	50
PMS	5	141	13	110	0.52	84	12	100
DB	5	49	3	43	0.63	78	20	50
DJ	5	18	2	15	0.77	88	13	100
D_{speech}	25	352	50	263	0.65	82	14	71
D_{news}	10	67	5	58	0.69	80	18	75
$D_{speech} + D_{news}$	35	419	55	321	0.65	82	15	74

Table 10: Qualitative evaluation of the MUTANT dataset. A: Articles, CA: complete agreement between the annotators, Hing: Hinglish MST. E/H: English/Hindi MST, CKS: Cohen's kappa score.

tors are proficient in English, Hindi, and Hinglish languages. We randomly sample five articles from each of the seven source datasets and share the originally scraped articles containing both identified MCT and monolingual MST with both evaluators. During the evaluation, we do not disclose which of the MSTs is identified as MCT and share the following guidelines:

1. Any MST containing only Hindi words or only English words is monolingual.
2. Any named entity, date, number, or word common in both English and Hindi languages should be considered a language-independent word.

In Table 10, we report our findings from the qualitative evaluation study. Out of a total of 419 MST, we observe the complete agreement on 321 monolingual MST and 55 code-mixed MST resulting in ≈90% complete agreement. A complete agreement means that both annotators agree that any particular MST is code-mixed or not. On MST with CA, we further compute the three metric scores using MDG. The results strengthen our earlier findings from Section 5. In Figure 5, we report two example MCT incorrectly identified by our MCT identifica-
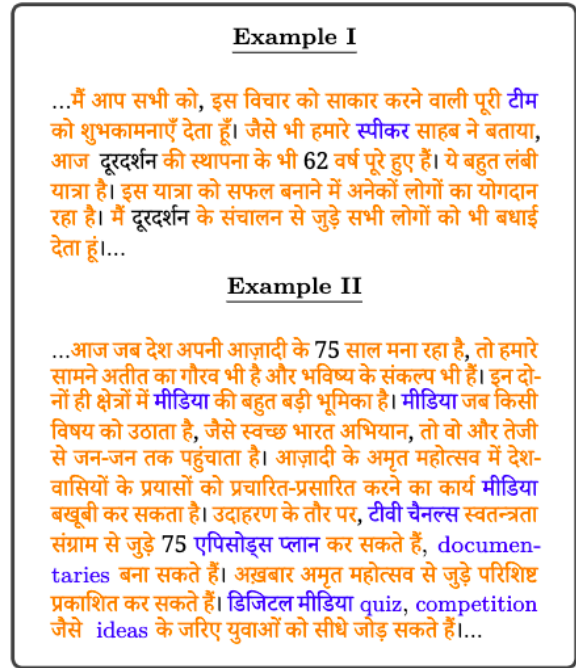

Figure 5: False positive MCT. We color code the tokens as: Hindi, English, and language independent.

tion pipeline. In the first example, both evaluators show complete agreement whereas in the second example there is a disagreement between the evaluators. We attribute this behavior to the poor state of the current code-mixed LID systems (Srivastava and Singh, 2021a) and since the CMI metric and our dual MEC formulation depend heavily on the code-mixed LID tools, the final results get affected. This limitation further provides an opportunity for future works to explore the problem from different perspectives such as a token-level language-independent MCT identification pipeline. It will also be interesting to see how this pipeline performs with other code-mixed languages, especially in a low-resource setting.

7 Conclusion

In this paper, we present a novel task of identifying MCT from multilingual documents. We propose an MCT identification pipeline by extending CMI to the multi-sentential framework and leveraging the pipeline we build a dataset for the Hinglish language. We highlight several challenges in building such resources and our insights will be useful to future works in code-mixed and low-resource languages.

8 Limitations

The limitations with the *MUTANT* dataset include but are not limited to:

- Contrary to the previous works, all the data sources comprises the non social media sites. This could potentially limit the diversity in the code-mixed text as observed on social media platforms.
- In the current form, the dataset is limited to only one code-mixed language. We believe the proposed technique to extract MCT could be expanded to other code-mixed languages in the future.
- The data sources could potentially have their own biases (topical, style of writing, etc). We expect future works to be cautious while generalizing the results obtained on this dataset.

References

Joshua Ainslie, Santiago Ontanon, Chris Alberti, Vaclav Cvicek, Zachary Fisher, Philip Pham, Anirudh Ravula, Sumit Sanghai, Qifan Wang, and Li Yang. 2020. Etc: Encoding long and structured inputs in transformers. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 268–284.

Guangsheng Bao, Yue Zhang, Zhiyang Teng, Boxing Chen, and Weihua Luo. 2021. G-transformer for document-level machine translation. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 3442–3455.

Ruthanna Barnett, Eva Codó, Eva Eppler, Montse Forcadell, Penelope Gardner-Chloros, Roeland Van Hout, Melissa Moyer, Maria Carme Torras, Maria Teresa Turell, Mark Sebba, et al. 2000. The lides coding manual: A document for preparing and analyzing language interaction data version 1.1–july 1999. *International Journal of Bilingualism*, 4(2):131–271.

Isabel Cachola, Kyle Lo, Arman Cohan, and Daniel S Weld. 2020. Tldr: Extreme summarization of scientific documents. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 4766–4777.

Amitava Das and Björn Gambäck. 2014. Identifying languages at the word level in code-mixed indian social media text. In *Proceedings of the 11th International Conference on Natural Language Processing*, pages 378–387.

Bogdan Gliwa, Iwona Mochol, Maciej Biesek, and Aleksander Wawer. 2019. Samsum corpus: A human-annotated dialogue dataset for abstractive summarization. In *Proceedings of the 2nd Workshop on New Frontiers in Summarization*, pages 70–79.

Gualberto Guzmán, Joseph Ricard, Jacqueline Serigos, Barbara E Bullock, and Almeida Jacqueline Toribio. 2017. Metrics for modeling code-switching across corpora. *Proc. Interspeech 2017*, pages 67–71.

Mandar Joshi, Eunsol Choi, Daniel S Weld, and Luke Zettlemoyer. 2017. Triviaqa: A large scale distantly supervised challenge dataset for reading comprehension. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1601–1611.

Simran Khanuja, Sandipan Dandapat, Sunayana Sitaram, and Monojit Choudhury. 2020. A new dataset for natural language inference from code-mixed conversations. In *Proceedings of the The 4th Workshop on Computational Approaches to Code Switching*, pages 9–16.

Anoop Kunchukuttan, Pratik Mehta, and Pushpak Bhattacharyya. 2018. The iit bombay english-hindi parallel corpus. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*.

Laiba Mehnaz, Debanjan Mahata, Rakesh Gosangi, Uma Sushmitha Gunturi, Riya Jain, Gauri Gupta, Amardeep Kumar, Isabelle G Lee, Anish Acharya, and Rajiv Shah. 2021. Gupshup: Summarizing open-domain code-switched conversations. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 6177–6192.

Ravindra Nayak and Raviraj Joshi. 2022. L3cube-hingcorpus and hingbert: A code mixed hindi-english dataset and bert language models. *arXiv preprint arXiv:2204.08398*.

Eva Sharma, Chen Li, and Lu Wang. 2019. Bigpatent: A large-scale dataset for abstractive and coherent summarization. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 2204–2213.

Vivek Srivastava and Mayank Singh. 2020. Phinc: A parallel hinglish social media code-mixed corpus for machine translation. In *Proceedings of the Sixth Workshop on Noisy User-generated Text (W-NUT 2020)*, pages 41–49.

Vivek Srivastava and Mayank Singh. 2021a. Challenges and limitations with the metrics measuring the complexity of code-mixed text. In *Proceedings of the Fifth Workshop on Computational Approaches to Linguistic Code-Switching*, pages 6–14.

Vivek Srivastava and Mayank Singh. 2021b. Hinge: A dataset for generation and evaluation of code-mixed hinglish text. In *Proceedings of the 2nd Workshop on Evaluation and Comparison of NLP Systems*, pages 200–208.

Makarand Tapaswi, Yukun Zhu, Rainer Stiefelhagen, Antonio Torralba, Raquel Urtasun, and Sanja Fidler. 2016. Movieqa: Understanding stories in movies through question-answering. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4631–4640. IEEE.