Universal Dependencies Treebank for Tatar: Incorporating Intra-Word Code-Switching Information

[ə, ɣ]Chihiro Taguchi, [ə]Sei Iwata, [ə]Taro Watanabe
[ə]Nara Institute of Science and Technology, Japan
[ɣ]University of Edinburgh, United Kingdom
{taguchi.chihiro.td0, iwata.sei.is6, taro}@is.naist.jp

Abstract

This paper introduces a new Universal Dependencies treebank for the Tatar language named NMCTT. A significant feature of the corpus is that it includes code-switching (CS) information at a morpheme level, given the fact that Tatar texts contain intra-word CS between Tatar and Russian. We first outline NMCTT with a focus on differences from other treebanks of Turkic languages. Then, to evaluate the merit of the CS annotation, this study concisely reports the results of a language identification task implemented with Conditional Random Fields that considers POS tag information, which is readily available in treebanks in the CoNLL-U format. Experimenting on NMCTT and the Turkish-German CS treebank (SAGT), we demonstrate that the proposed annotation scheme introduced in NMCTT can improve the performance of the subword-level language identification. This annotation scheme for CS is not only universally applicable to languages with CS, but also shows a possibility to employ morphosyntactic information for CS-related downstream tasks.

Keywords: Tatar, treebank, Universal Dependencies, code-switching, low-resource languages, language identification

1. Introduction

Globalization and the digital revolution affect the world's languages in a two-fold manner. On one side, except for a handful of languages with a prominent international status, no languages are immune to the multilingualism, diglossia, and language shift to a majority language. In such a linguistic community, it is common to find these languages mixed within a single discourse. This linguistic phenomenon is called code-switching (CS). On the other hand, the information society enables us to access data of low-resource languages more easily. This situation coincides with the recent trend of multilingual and low-resource natural language processing (NLP) and their applications. Universal Dependencies (UD) (Nivre et al., 2020) is one of such projects that aims to create multilingual annotated corpora with universal rules and labels.

Following this momentum, this paper provides two main contributions: (1) it introduces the NAIST Multilingual Corpus Tatar (NMCTT[1]), and (2) validates the benefits of NMCTT's CS segmentation annotation. NMCTT is the first annotated corpus for the Tatar language. The innovative characteristic of the corpus is that language code information is explicitly annotated for each word, and CS segments and corresponding language codes are added if CS occurs within a word, which we call intra-word CS in this paper. For the evaluation of the usefulness of incorporating intra-word CS in UD, we conduct simple experiments of character-level tagging for both span prediction and language identification on Tatar–Russian and Turkish–German data. Leveraging the part-of-speech (POS) tag information which is readily available in the CoNLL-U format, we show that combining UD's linguistic information and CS annotation has the potential to improve the performance of segment-level language classification. In doing so, we encourage the annotation of language tags in treebanks of languages with CS.

[1]TT is from `tt`, the ISO 639-1 language code for Tatar.

1.1. Tatar: Linguistic Background

The Tatar language, a language categorized in the Kipchak (Northwestern) language group of the Turkic language family, is chiefly spoken in the Republic of Tatarstan, Russia. Kazakh, Kyrgyz, and Bashkir are other notable languages that fall into the same language group. Tatar is reported to have more than 5 million speakers (Eberhard et al., 2021), most of which are bilingual with Russian. However, the bilingualism is asymmetric; that is, while the Tatars communicate in Tatar and Russian, the Russians typically speak only in Russian (Safina, 2020). This asymmetry leads to frequent CS with, and gradual language shift to, Russian, leaving Tatar less resourced.

The canonical word order of Tatar is Subject-Object-Verb, and adjectival modifiers precede the modified nouns, i.e., head-final. It is a typical agglutinative language, and nominal case and verbal inflection are marked by suffixes.

Most modern Tatar texts are written in the Cyrillic script with some extensions to express phonemes unique to Tatar. The language can also be written in the Latin script, and the Latin orthography is mainly used among diaspora communities in Turkey and Finland. The linguistic examples from Tatar in this paper employ the Latin alphabet for convenience.

ID	FORM	LEMMA	UPOS	FEATS	HEAD	DEPREL	MISC
1	Татарстанда	Татарстан	PROPN	Case=Loc|Number=Sing	5	obl	LangID=TT
2	коронавирустан	коронавирус	NOUN	Case=Abl|Number=Sing	4	nmod	CSPoint=коронавирус§тан|LangID=MIXED[RU§TT]
3	беренче	беренче	ADJ	_	4	amod	LangID=TT
4	прививканы	прививка	NOUN	Case=Acc|Number=Sing	5	obj	CSPoint=прививка§ны|LangID=MIXED[RU§TT]
5	ясатырга	яса	VERB	VerbForm=Inf|Voice=Cau	0	root	LangID=TT
6	мөмкин	мөмкин	AUX	_	5	aux	LangID=TT|SpaceAfter=No
7	.	.	PUNCT	_	5	punct	LangID=OTHER

Table 1: An example of annotation with CS information. Note that the optional columns XPOS and DEPS are omitted as they are left blank in NMCTT. The free translation of the original sentence is "The first dose for coronavirus will be available in Tatarstan."

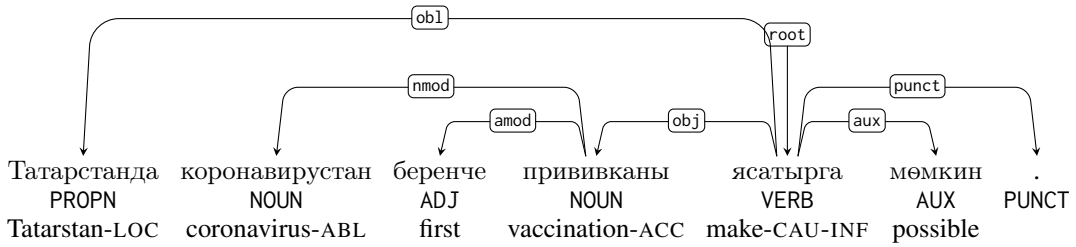

Figure 1: Dependency tree of Table 1 with morphological information.

1.2. Intra-Word Code-Switching

CS is broadly defined as "the alternative use by bilinguals of two or more languages in the same conversation" (Alvanoudi, 2017). When a minority language's morphology more or less has grammatical declension and inflection, CS can occur inside a word. Although intra-word CS is commonly found in minority language varieties, it is not easy to collect their text data, because intra-word CS is often a colloquial phenomenon and is not written down. The word in (1) is an example of intra-word CS in Tatar.[2] The noun *privivka* is a Russian word meaning "vaccination", and it takes an accusative case-marking suffix *-nı*.

(1) *privivka -nı*
RU -TT
vaccination-ACC

One may well see this sort of CS as mere loanwords, but in NMCTT we categorize them as a subset of CS for the following three reasons. First, these Russian words mixed in Tatar are typically pronounced with the Russian phonology (Burbiel, 2018), whereas loanwords are more or less incorporated in the phonology of the receiving language (Kang, 2011). Second, the use of Russian words depends on the speaker's preference and knowledge of and attitude toward the two languages as well as on other social factors (Burbiel, 2018). This tendency conforms with the characteristics of CS, which constitutes "a contact-induced speech behavior that occurs extensively in the talk of bilinguals" in contrast to borrowing that constitutes "a completed contact-induced change" (Alvanoudi, 2017). Third, handling these words as CS has benefits in other applications of text processing such as a transliteration task. Tatar and Russian are transliterated into Latin characters differently. For this reason, it is practically more convenient to annotate them as Russian that is code-switched from Tatar rather than as loanwords integrated in Tatar. Therefore, this study treats them as (intra-word) CS.

[2]Following the convention of linguistics, the text is transliterated into Latin Tatar.

2. Related Work

Code-switching and language resources. Recent spotlight on CS has brought several annotated corpora of various CS language pairs. SEAME (Lyu et al., 2010) and the Mandarin–English code-switching corpus (Li et al., 2012) are some of the first CS resources for computational linguistics.

UD Turkish-German SAGT (Çetinoğlu, 2016) is another treebank that handles CS texts and explicitly annotates a language tag and CS segmentation. UD Hindi-English HIENCS (Bhat et al., 2018) is a corpus of tweets involving Hindi-English CS, but CS occurs on the word level (i.e., not within a word). Tagalog is also a language known to have CS with English especially in its colloquial variety, but the two Tagalog treebanks available on UD, TRG (Samson et al., 2020) and Ugnayan (Aquino, 2020), do not contain any explicit annotation marking CS.

There are several language resources of Tatar, such as the Corpus of Written Tatar (Saykhunov et al., 2021) with ∼356 million tokens and the Tatar National Corpus (Suleimanov et al., 2013) with ∼180 million tokens. Although these corpora contain POS and morphological information linked with the text, it is automatically generated through a rule-based tagging. Therefore, at the time of writing, there is no manually annotated treebank of Tatar, let alone on UD.

Available Turkic UD treebanks. From the Turkic language family excluding Tatar and high-resource

Turkish, the present UD v2.9 contains Kazakh KTB (Makazhanov et al., 2015; Tyers and Washington, 2015) with 10,383 tokens, Old Turkish Tonqq (Derin and Harada, 2021) with 221 tokens, Uyghur UDT (Eli et al., 2016) with 40,236 tokens, and Yakut YKTDT (Merzhevich and Gerardi, 2021) with 271 tokens. Except for Turkish UD treebanks that contain 733K tokens in total, Turkic languages in UD are overall low-resourced.

Colloquial Kazakh also has CS similar to Tatar, but the Kazakh KTB treebank is based on formal written texts that do not contain CS, and therefore does not consider language tagging.

Language processing for CS. Though not much work has been done on computational approaches to CS relative to how common CS is in the world, one of the earliest studies on the topic is Joshi (1982) which investigated CS between Marathi and English. Early work on identifying segmental points where languages are switched is Solorio and Liu (2008), where the model was trained to learn to predict natural CS points. Anastasopoulos et al. (2018) conducted research on a POS tagging task for Griko, a language with token-level CS to Italian. Exploiting additional grammatical information for a tagging task is discussed in Silfverberg et al. (2014).

More recent work includes intra-word CS where language codes may switch at a morpheme level, particularly found in morphologically rich languages. Intra-word CS language identification by Mager et al. (2019) employs Segmentation Recurrent Neural Network (SegRNN) (Lu et al., 2016) to test on CS texts in the language pairs of German–Turkish and Spanish–Wixarika, a Uto-Aztecan language indigenous to Mexico. Sabty et al. (2021) also uses SegRNN for the language identification task of Arabic–English CS texts. Taguchi et al. (2021) is a work on transliteration from Cyrillic Tatar to Latin Tatar combining subword-level language identification; however, the subword tokenization is fully done by Byte-Pair Encoding (BPE).

3. Overview of the Tatar Universal Dependencies

This section outlines the feature of NMCTT with an emphasis on the comparison with other treebanks of Turkic languages. The policies of the annotation by and large follow the guideline proposed in Tyers et al. (2017). An exemplary annotation is shown in Figure 1 as well as its dependency tree in Figure 1.

3.1. Text

The raw text is obtained from the Tatar language version of Tatar-Inform,[3] an online news media actively posting articles in Tatar and Russian.

Note that, upon the use of the news text, it is necessary to attach a hyperlink to the original news article, as stipulated in the Russian federal law. In the treebank, the source link of each sentence is explicitly shown in the metadata row starting from `# link =`.

3.2. Tokenization and Word Segmentation

We obtained tokens by splitting at spaces and punctuation. UD Turkish-German SAGT employs a slightly more fine-grained approach to tokenizing sentences. For example, the Turkish locative adjectivizer suffix *-ki* is attached to the preceding element directly in the Turkish orthography, and SAGT further tokenizes them as different tokens. An example of the usage of *-ki* and the corresponding morpheme in Tatar *-ğı/ge* (*-qı/ke*) are illustrated in phrases (2) and (3).[4] The contrast is apparent in Tables 2 and 3. While NMCTT treats *Berlin-da-ğı* ("in Berlin") as one token, SAGT detaches *-ki* of *Berlin'-de-ki* and treats *ki* as an adposition.

(2) *Berlin'-de-ki* *ev* (Turkish)
Berlin-LOC-ADJVZ house
"A house in Berlin"

(3) *Berlin-da-ğı* *öy* (Tatar)
Berlin-LOC-ADJVZ house
"A house in Berlin"

ID	FORM	LEMMA	UPOS
1-2	Berlin'deki	_	_
1	Berlin'de	Berlin	PROPN
2	ki	ki	ADP

Table 2: Tokenization and tags in SAGT.

ID	FORM	LEMMA	UPOS
1	Берлиндагы	Берлин	PROPN

Table 3: Tokenization and tags in NMCTT (transliterated).

The first motivation to tokenize text simply by spaces and punctuation is that it will ensure more accurate automatic tokenization than splitting inside a word. The second motivation, at least in Tatar, is that the morpheme *-ğı/ge* (*-qı/ke*) corresponding to Turkish *-ki* is often treated as a derivational suffix to form a relational adjective (Burbiel, 2018) rather than an independent word or a clitic. Therefore, it is unnatural to tokenize it as a separate word that bears a POS tag.

3.3. Parts-of-speech

The statistics of the Universal POS (UPOS) tags are summarized in Table 4. Of all the UPOS tags, INTJ (interjection), PART (particle), SYM (symbol), and X (other)

[3] https://tatar-inform.tatar.

[4] See Appendix for glossing abbreviations. Note that *-ki* in Turkish and *-ğı/ge* (*-qı/ke*) in Tatar have several morphosyntactic properties, and ADJVZ "adjectivizer suffix" is a tentative glossing that by and large covers their properties.

Class	UPOS	Total	Russian	Mixed
Open	NOUN	413	21	62
	PROPN	79	34	8
	VERB	169	0	1
	ADJ	117	8	0
Closed	AUX	18	0	0
	DET	9	0	0
	ADV	40	0	0
	SCONJ	8	0	0
	ADP	35	0	0
	CCONJ	26	0	0
	PRON	26	0	0
	NUM	12	0	0
Other	PUNCT	167	0	0

Table 4: The distribution of UPOS tags in the treebank with respect to language code. The first column specifies whether the UPOS tag is an open class or a closed class.

do not appear in the present NMCTT. The use of PART is explicitly avoided as the UD guideline notes that "the PART tag should be used restrictively and only when no other tag is possible".[5] Other unattested UPOS tags might appear in additional texts in the future. Table 4 also demonstrates the disproportional distribution of CS in each POS tag. While open class words, such as NOUN and PROPN, contain several cases of CS to Russian, closed class words only appear in Tatar. This sort of distributional tendency of CS has often been empirically reported such as in Joshi (1982). We will return to this point in Section 4.

3.4. Morphology

The morphological features (e.g., in the FEATS column in Figure 1) in NMCTT are designed to be correspondent with morphological inflection as uniquely as possible. An example that reflects this policy well is the treatment of converbs. A converb is a non-finite verb form whose main function is to mark adverbial subordination (Haspelmath, 1995). In UD, converb is loosely defined as "a non-finite verb form that shares properties of verbs and adverbs."[6] Turkic languages are commonly known to have several converbs (Johanson, 2021). In Tatar, for instance, boldfaced converbs in sentences (4)–(7) are contrasted in the aspect. The suffix *-(ï/e)*p exemplified in (4) is a generic kind of converb used to denote consecutive or simultaneous actions and states. The suffix *-a/ä* (*-ıy/i*) in (5) composes a converb of simultaneous action or state. (6) shows a case of converb suffix *-ğaç/gäç* that means the action precedes the action expressed by the main predicate. The fourth suffix *-ğançı/-gänçe*, in contrast, expresses an action or state that happens after the event of the main predicate. To distinguish these functionally different converbs, NMCTT is designed to have different morphological annotations in the FEATS column for each of these converb suffixes.

(4) *ul* ***aşa-p*** *utır-a.*
he eat-CVB sit-PRS.3
"S/he is sitting and eating." (manner)
VerbForm=Conv

(5) ***aşı-y*** ***aşı-y*** *öy-gä* *qayt-tı.*
eat-CVB.PROG = house-DAT return-PST.3
"(S/he) went home while eating." (simultaneity)
Aspect=Prog|VerbForm=Conv

(6) ***aşa-ğaç*** *öy-gä* *qayt-tı.*
eat-CVB.PF house-DAT return-PST.3
"(S/he) went home after eating." (prior event)
Aspect=Perf|VerbForm=Conv

(7) ***tuy-ğançı*** *aşa-dı.*
become.full-CVB.IMPF eat-PST.3
"He ate till he became full." (posterior event)
Aspect=Imp|VerbForm=Conv

The annotation of converbs differs to a great extent among the treebanks of the Turkic languages, in particular of Turkish, as shown in Table 5. The treebank that shares the similar spirit to ours is Uyghur UDT (Eli et al., 2016).

3.5. Syntactic Dependency

Syntactic trees often differ in shape and branching among modern linguistic theories. However, following the gist of UD that pursues the unified format and rules for describing dependency, we tried to avoid innovative usage of dependency tags in NMCTT, and conformed to the guidelines for Turkic languages proposed by Tyers et al. (2017) as well as conventions in other existing UD treebanks.[7]

3.5.1. Nominal Arguments: nsubj, obj, obl

In UD, there are three grammatical relations of nominal arguments to a predicate: nsubj for a nominal subject, obj for a direct object, and obl for other non-core arguments. These notions are compatible with Lexical Functional Grammar (LFG) (Dalrymple, 2001).
For example, non-core arguments in sentence (8) are parsed with obl dependency relation as in Figure 2.

(8) *bala uram-da at-qa alma bir-ä*
child street-LOC horse-DAT apple give-PRS.3
"A child gives an apple to the horse on the street."

[5] https://universaldependencies.org/u/pos/PART.html

[6] https://universaldependencies.org/u/feat/VerbForm.html

[7] The detailed list of tags used in NMCTT and their statistics are summarized on the UD website: https://universaldependencies.org/treebanks/tt_nmctt/index.html.

Language	Treebank	POS	VerbForm=Conv	Conv distinction
Tatar	NMCTT	VERB	correct	yes
Turkish	FrameNet	ADV	NA	no
	GB	VERB	correct	no
	Kenet	ADV	incorrect	no
	Penn	ADV	incorrect	no
	Tourism	ADV	incorrect	no
	Atis	ADV	incorrect	no
	BOUN	VERB	incorrect	yes
	PUD	ADV	incorrect	yes
	IMST	VERB	correct	no
Turkish German	SAGT	VERB	correct	no
Kazakh	KTB	VERB	correct	yes
Uyghur	UDT	VERB	correct	yes

Table 5: Comparison of the annotation for converbs in Turkic treebanks. The values in the column "VerbForm=Conv" summarizes whether the corpus annotates converbs as VerbForm=Conv correctly; if the feature is not used at all, the value is NA. The column "Conv distinction" shows whether functionally different converbs are distinguished in morphological features. Yakut and Old Turkish are not included because the converb is not attested or left unannotated in the corpora.

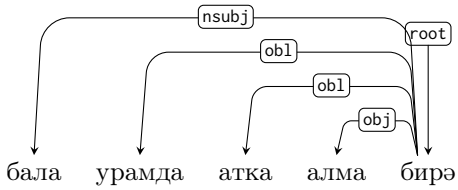

Figure 2: Dependency parsing of (8).

3.5.2. Copula: cop

Though nominal predication does not require a copula in the present tense in Tatar, it employs an overt copula in the past and future tenses. In UD, it is conventional to treat a predicate noun as a head of a copula unlike approaches of generative syntax that often puts a copula higher than the predicate noun. In this respect, too, UD shares the formalism in common with LFG.

(9) *min student ide-m*
I student COP.PST-1SG
"I was a student"

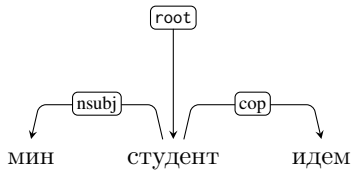

Figure 3: Dependency parsing of (9).

3.5.3. Light Verb Construction: compound:lvc

A light verb is a verb that has little meaning by itself but forms a complex predicate with a noun which serves as the semantic content. This complex predicate construction is labeled as compound:lvc in UD's dependency annotation. In light verb constructions, the verb is conventionally treated as the head of the noun in UD. In Tatar, there are a number of light verb constructions, typically with a light verb *it-*, as exemplified in sentence (10). The corresponding dependency is illustrated in Figure 4.

(10) *däres däwam it-te*
class continuation do-PST.3
"The class continued"

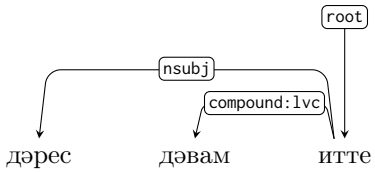

Figure 4: Dependency parsing of (10).

3.5.4. Grammaticalized Auxiliaries: aux

In Tatar, certain verbs following a converb are grammaticalized to lose their original lexical meaning and gain a new functional role. As shown in the example (11), the finite verb *çıq-* no longer retains its generic meaning of going out, but denotes aspectual semantics that implies the completion of the action expressed by the preceding converb. In such a case, the dependency relation between the converb and the finite verb is marked as aux (auxiliary), where the head is the converb. Therefore, the dependency tree of sentence (11) should be as in Figure 5.

(11) *ul kitap-nı uqı-p çıq-tı*
he book-ACC read-CVB go.out-PST.3
"he read the book (finished reading the whole)"

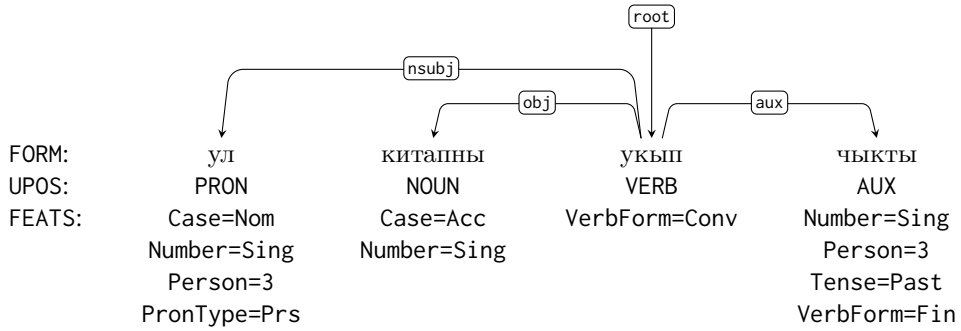

Figure 5: Dependency parsing of (11).

Note that the canonical usage of converbs described in Section 3.4 is represented by the dependency relation advcl (adverbial clause). In this case, the converb is the dependent of the main inflected verb, as illustrated in sentence (12) and its dependency tree in Figure 6.

(12) *ul kitap-nı uqı-p yoqla-dı*
he book-ACC read-CVB sleep-PST.3
"He read the book and slept."

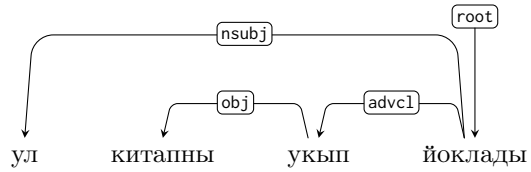

Figure 6: Dependency parsing of (12).

3.6. Language Tags (LangID=)

The most innovative characteristic of NMCTT is that it explicitly annotates language tags at a segment level for each token in the MISC column of the CoNLL-U format table. The idea of incorporating CS information into UD has already been carried out in UD Hindi-English HIENCS (Bhat et al., 2018) and UD Turkish-German SAGT (Çetinoğlu, 2016). However, HIENCS does not consider intra-word CS, and SAGT simply tags the intra-word CS with the MIXED tag, agnostic of what language codes are inside the token. UD Komi-Zyrian Lattice (Partanen et al., 2018), a UD treebank of another minority language of Russia, also explicitly annotates Russian words by specifying as OrigLang=ru, but their CS segments are unclear. NMCTT differs from these corpora by tagging each intra-word CS segment with a language code, allowing for more flexibility and expressiveness in the language tagging.

An example of segment-level language tagging in NMCTT is shown in (13). Following SAGT, the segmentation point is marked by the character § in the element starting with CSPoint=. The breakdown of the mixed languages is described in the brackets after MIXED. The same character § is used to show the segments where the languages are switched, which corresponds to the segment described in CSPoint=. Гыйбәтдинов (*translit*. *Ğibätdinov*) is a Tatar male surname that consists of a Tatar-origin morpheme *Ğibätdin* and a Russian-origin suffix *-ov* that derives a Russified surname from a non-Russian surname ending with a consonant. In the example, a dative case suffix *-qa*, a Tatar morpheme, is added.

Language	Count
Tatar (TT)	819
Russian (RU)	63
Mixed (MIXED)	71
Other (OTHER)	166

Table 6: Distribution of language tags in NMCTT for each token.

(13) CSPoint=Гыйбәтдин§ов§ка
LangID=MIXED[TT§RU§TT]
"To Gibatdinov"

The criteria for determining if a segment is considered CS or is a loanword are outlined in Section 1.2. For example, the Tatar word *mömkin* etymologically comes from Arabic مُمْكِن (*mumkin*), but the word is fossilized in the Tatar vocabulary and also is pronounced in accordance with the Tatar phonological paradigm, and thus it is classified as a loanword and not a CS word.

The statistics of tokens for each language ID in NMCTT is summarized in Table 6. Note that NMCTT does not use the language label LANG3 used in SAGT.

4. Experiment: Language Identification and Segmentation

To evaluate the usefulness of the proposed CS annotation, we implement a simple character-level tagger that jointly predicts language tags and span boundaries taking into account the corresponding POS tag. We test it not only on the UD Tatar treebank but also on UD Turkish-German SAGT (Çetinoğlu, 2016).

The Tatar training and test data contain 888 and 231 tokens, respectively. For the Turkish-German dataset, the training data contains 10,005 tokens; we concatenated the dev and test files to use them as the test data, comprising 26,929 tokens. Since the dataset of NMCTT is

too small to demonstrate the effects of POS tags statistically, we employ SAGT to verify the results.

Note that the objective of this experiment is to verify the effects of adding POS features in an explainable manner, and not to pursue the state-of-the-art performance of language identification and span prediction.

4.1. Task Description

The architecture of the span prediction and language classification task is as follows. Given an input word x that consists of characters $\langle c_1, \cdots, c_{|x|} \rangle$, our objective is to correctly predict a pair of a language tag $y_l\{\text{L1, L2, other}\}$ and a span tag $y_s \in \{\text{B, I, E, S}\}$ for each character. (L1, L2) are the CS language pair, i.e., (Tatar, Russian) or (Turkish, German), and tokens in other languages or punctuation fall in to "other". B, I, E denote the beginning, intermediate, and ending position of a segment respectively, and S a single-character segment. Prediction $\hat{y}$ is taken to be correct only if it matches both the corresponding language tag y_l and span tag y_s. Therefore, there are 12 possible labels to be predicted in the task.

To keep the labels consistent in tests on both NMCTT and SAGT, label `LangID=LANG3` used in SAGT (a third language that is neither Turkish nor German) is converted to "other" during the data formatting.

4.2. Language–Span Tagging with Conditional Random Fields

The tagger is modeled with Conditional Random Fields (CRFs) (Lafferty et al., 2001). To predict correct labels $\boldsymbol{y}$ given a sequence of input $\boldsymbol{x}$, the CRF model is defined as

$$p(\boldsymbol{y}|\boldsymbol{x}) = \frac{1}{Z(\boldsymbol{x})} \exp\left\{\sum_{k=1}^{K} \lambda_k f_k(y_t, y_{t-1}, \boldsymbol{x}_t)\right\},$$

where $Z(\boldsymbol{x}) = \sum_{\boldsymbol{y}} \exp\left\{\sum_{k=1}^{K} \lambda_k f_k(y_t, y_{t-1}, \boldsymbol{x}_t)\right\}$ is a normalization function to ensure the sum of p is 1, λ_k is a parameter vector, and $\{f_k\}_{k=1}^{K}$ is a set of feature functions. The feature function f takes into account bigram features (transition) and unigram features (observation/emission) by applying a function $\mathbf{1}_{\{q\}}$ that returns 1 when the desired condition q is met, namely:

$$\mathbf{1}_{\{q\}} = \begin{cases} 1 & \text{if } q \text{ is true} \\ 0 & \text{otherwise.} \end{cases}$$

K is the total number of features after combining transition features $f_{ij}(y, y', x) = \mathbf{1}_{\{y=i\}}\mathbf{1}_{\{y'=j\}}$ for each transition (i, j) and observation features $f_{io}(y, y', x) = \mathbf{1}_{\{y=i\}}\mathbf{1}_{\{x=o\}}$ for each state-observation pair (i, o); namely,

$$\begin{aligned} &\sum_{k=1}^{K} \lambda_k f_k(y_t, y_{t-1}, \boldsymbol{x}_t) \\ &= \sum_{t} \left\{ \sum_{i,j} \lambda_{ij} f_{ij}(y_t, y_{t-1}, \boldsymbol{x}_t) \right. \\ &\qquad \left. + \sum_{i,o} \lambda_{io} f_{io}(y_t, y_{t-1}, \boldsymbol{x}_t) \right\}. \end{aligned}$$

One of the architectural strengths of CRFs is that we can specify features that we want to include in the feature extraction function. The list of employed features used as the default in the experiment their example values are illustrated in Table 7. For example, assuming that we are looking at the first character "M" of the word "Mars", the extracted features will look like the right column of the table. We used trigram features to take neighboring characters into account as observation features. In doing so, it is possible to approximately model meaningful morphological units such as affixes. In addition, character features such as letter case, digit, and punctuation play a significant role in predicting correct language and span tags. Though the task is done at the character level, it is also possible to include word-level information such as the word form, word length, and its POS tag as a part of the observation features. In the ablation studies to confirm the efficacy of adding POS tag information, the POS and Word features (the last two rows in Table 7) are to be omitted.

An intuitive motivation to include POS tag in the features rather than morphology (`FEAT`) and dependency (`DEPREL`) comes from the following two points. First, since the number of POS tags is limited compared to other grammatical features such as dependency and morphology, we can assume that the effect of POS tags is straightforward and is easier to interpret. Second, intuitively, the distribution of tokens that undergo CS, at least in Tatar, seems to depend on POS tags as illustrated in Table 4.

For the training step, we chose limited-memory BFGS (Liu and Nocedal, 1989) as the parameter optimization algorithm, and set the L1 and L2 regularization parameters to 0.25 and 0.3, respectively, and the max iteration to 100. The evaluation is based on precision, recall, and F1 scores, considering the fact that the class distribution is imbalanced as seen in Table 6.

The architecture for the experiment is implemented on `sklearn-crfsuite`,[8] a wrapper library of CRFsuite (Okazaki, 2007) made to be compatible with `scikit-learn`.

4.3. Results and Discussion

Tables 8 and 9 show the results with ablation studies of the experiment. Note that all values are weighted average scores, and the F1 scores are not derived directly

[8] `https://sklearn-crfsuite.readthedocs.io`

Feature	Example value
Character	"M"
Character +1	"a"
Character +2	"r"
Character -1	False
Character -2	False
Word-initial?	True
Word-final?	False
Word in titlecase?	True
Character in uppercase?	True
Punctuation?	False
Number?	False
Word length	4
POS	"PROPN"
Word	"Mars"

Table 7: An example of a feature table for the character "M" in "Mars".

Features	Precision	Recall	F1
Default	**90.9**	**90.0**	**88.9**
[-POS]	87.3	86.5	84.3
[-word]	86.4	86.5	84.9
[-POS, -word]	86.7	87.0	85.7

Table 8: Ablation study of features on NMCTT. Scores are calculated at a character level.

from the precision and recall scores in the tables. In both NMCTT and SAGT, the default architecture with both POS and word form information resulted in the highest values of precision, recall, and F1. Also, compared to the model without the POS feature, we can see that the model with the default feature set performs better. However, it is notable that, though we expect models with POS features to be more accurate than ones without POS, the [-word] model in NMCTT turned out to work better than the [-POS, -word] model. This may partially come from the scarcity of the available data in NMCTT, as the scores are more susceptible to one error. We aim to enhance the data size of the NMCTT treebank in future releases.

These results imply that leveraging additional grammatical information available in UD potentially improves the performance of the segmentation and language classification task for both high- and low-resource languages. Although the experiment did not involve other features that can be extracted from UD's CoNLL-U format data, UD's flexibility also allows them to be incorporated in the features. This perspective is worth investigating further in future work. In addition, the results also conform with the observation in Table 4 in the previous section that the distribution of CS tokens is related to that of POS tags.

5. Concluding Remarks

This study reported NMCTT's contribution to UD and discussed the treebank from two aspects. First, we outlined the new treebank focusing on the cross-linguistic validity with the comparison to other Turkic UD treebanks. One of its important contributions is that it proposed a way to annotate language labels at the CS segment level. Given the prevalence of CS, especially between low-resource languages and more prominent languages spoken in the same region, the proposed annotation scheme can be further applied to other CS languages. Second, to evaluate quantitatively the benefits of adding CS information at a morpheme level to the UD annotation, we experimented the joint task of CS segmentation and language identification on NMCTT and SAGT using a simple CRF architecture. The results showed that POS tag information is likely to be meaningful to intra-word language classification. This also implies that combining other linguistic information available on UD-format treebanks may contribute to the improvement in performance of downstream tasks related to CS.

Features	Precision	Recall	F1
Default	**95.9**	**96.1**	**95.9**
[-POS]	95.9	95.8	95.6
[-word]	94.6	94.7	94.6
[-POS, -word]	93.7	93.9	93.8

Table 9: Ablation study of features on SAGT. Scores are calculated at a character level.

NMCTT is still small in the UD v2.9 release; therefore, it is necessary to enlarge the data for more reliable and flexible applications. In addition, the evaluation was experimented solely on two corpora due to the limited quantities of available linguistic data. More active corpus building for low-resource CS languages will enable more investigation into the (non-)universality of this paper's finding.

6. Acknowledgments

The Tatar NMCTT Treebank is an outcome of the CICP NAIST Multilingual Corpus Project supported by the Nara Institute of Science and Technology. We thank Dr. Özlem Çetinoğlu for providing insightful advice for the annotation of code-switching texts. We are also grateful to Arturo and Justin from the NAIST NLP laboratory for proofreading and to the anonymous reviewers of LREC2022 for helpful suggestions and comments.

Appendix: Glossing Abbreviations

1, **2**, **3** — first, second, third person; **ABL** — ablative; **ACC** — accusative; **ADJVZ** — adjectivizer; **CAU** — causative; **COP** — copula; **CVB** — converb; **DAT** — dative; **IMPF** — imperfective; **INF** — infinitive; **LOC** — locative; **PF** — perfective; **PST** — past tense; **PROG** — progressive; **PRS** — present tense; **SG** — singular.

7. Bibliographical References

Alvanoudi, A. (2017). Language contact, borrowing and code switching: a case study of Australian Greek. pages 1–42.

Anastasopoulos, A., Lekakou, M., Quer, J., Zimianiti, E., DeBenedetto, J., and Chiang, D. (2018). Part-of-speech tagging on an endangered language: a parallel Griko-Italian resource. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 2529–2539, Santa Fe, New Mexico, USA, August. Association for Computational Linguistics.

Aquino, A. (2020). UD Tagalog Ugnayan. `https://github.com/UniversalDependencies/UD_Tagalog-Ugnayan`.

Bhat, I., Bhat, R. A., Shrivastava, M., and Sharma, D. (2018). Universal Dependency parsing for Hindi-English code-switching. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 987–998, New Orleans, Louisiana, June. Association for Computational Linguistics.

Burbiel, G. (2018). *Tatar Grammar: A Grammar of the Contemporary Tatar Literary Language*. Institute for Bible Translation.

Çetinoğlu, Ö. (2016). A Turkish-German code-switching corpus. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, pages 4215–4220, Portorož, Slovenia, May. European Language Resources Association (ELRA).

Dalrymple, M. (2001). *Lexical Functional Grammar*, volume 34 of *Syntax and Semantics*. Brill, Leiden, the Netherlands.

Derin, M. O. and Harada, T. (2021). Universal Dependencies for Old Turkish. In *Proceedings of the Fifth Workshop on Universal Dependencies (UDW, SyntaxFest 2021)*, pages 129–141, Sofia, Bulgaria, December. Association for Computational Linguistics.

Eberhard, D. M., Simons, G. F., and Fennig, C. D. (2021). Ethnologue: Languages of the world.

Eli, M., Mushajiang, W., Yibulayin, T., Abiderexiti, K., and Liu, Y. (2016). Universal dependencies for Uyghur. In *Proceedings of the Third International Workshop on Worldwide Language Service Infrastructure and Second Workshop on Open Infrastructures and Analysis Frameworks for Human Language Technologies (WLSI/OIAF4HLT2016)*, pages 44–50, Osaka, Japan, December. The COLING 2016 Organizing Committee.

Haspelmath, M. (1995). The converb as a cross-linguistically valid category. *Converbs in Cross-linguistic Perspective: Structure and Meaning of Adverbial Verb Forms — Adverbial Paritıcples, Gerunds —*, pages 1–55.

Johanson, L. (2021). *Turkic*. Cambridge Language Surveys. Cambridge University Press.

Joshi, A. K. (1982). Processing of sentences with intra-sentential code-switching. In *Coling 1982: Proceedings of the Ninth International Conference on Computational Linguistics*.

Kang, Y. (2011). Loanword phonology. In *The Blackwell Companion to Phonology*, chapter 95, pages 1–25. John Wiley Sons, Ltd.

Lafferty, J. D., McCallum, A., and Pereira, F. C. N. (2001). Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *Proceedings of the Eighteenth International Conference on Machine Learning*, ICML '01, page 282–289, San Francisco, CA, USA. Morgan Kaufmann Publishers Inc.

Li, Y., Yu, Y., and Fung, P. (2012). A Mandarin-English code-switching corpus. In *Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC'12)*, pages 2515–2519, Istanbul, Turkey, May. European Language Resources Association (ELRA).

Liu, D. C. and Nocedal, J. (1989). On the limited memory BFGS method for large scale optimization. 45:503–528.

Lu, L., Kong, L., Dyer, C., Smith, N. A., and Renals, S. (2016). Segmental recurrent neural networks for end-to-end speech recognition. *CoRR*, abs/1603.00223.

Lyu, D.-C., Tan, T. P., Siong, C. E., and Li, H. (2010). SEAME: a Mandarin-English code-switching speech corpus in South-East Asia. In *INTERSPEECH*.

Mager, M., Çetinoğlu, Ö., and Kann, K. (2019). Subword-level language identification for intra-word code-switching. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 2005–2011, Minneapolis, Minnesota, June. Association for Computational Linguistics.

Makazhanov, A., Sultangazina, A., Makhambetov, O., and Yessenbayev, Z. (2015). Syntactic annotation of kazakh: Following the universal dependencies guidelines. a report. In *3rd International Conference on Turkic Languages Processing, (TurkLang 2015)*, pages 338–350.

Merzhevich, T. and Gerardi, F. F. (2021). UD Yakut YKTDT. `https://github.com/UniversalDependencies/UD_Yakut-YKTDT`.

Nivre, J., de Marneffe, M.-C., Ginter, F., Hajič, J., Manning, C. D., Pyysalo, S., Schuster, S., Tyers, F., and Zeman, D. (2020). Universal Dependencies v2: An evergrowing multilingual treebank collection. In *Proceedings of the 12th Language Resources and Evaluation Conference*, pages 4034–4043, Marseille, France, May. European Language Resources Association.

Okazaki, N. (2007). CRFsuite: a fast implementation of Conditional Random Fields (CRFs).

Partanen, N., Blokland, R., Lim, K., Poibeau, T., and Rießler, M. (2018). The first Komi-Zyrian Universal Dependencies treebanks. In *Proceedings of the Second Workshop on Universal Dependencies (UDW 2018)*, pages 126–132, Brussels, Belgium, November. Association for Computational Linguistics.

Sabty, C., Mesabah, I., Çetinoğlu, Ö., and Abdennadher, S. (2021). Language identification of intra-word code-switching for Arabic–English. *Array*, 12:100104.

Safina, K. (2020). Bilingualism in the Republic of Tatarstan: language policy and attitudes towards Tatar language education.

Samson, S., Zeman, D., and Tan, M. A. C. (2020). UD Tagalog TRG. `https://github.com/UniversalDependencies/UD_Tagalog-TRG`.

Saykhunov, M. R., Khusainov, R. R., Ibragimov, T. I., Luutonen, J., Salimzyanov, I. F., Shaydullina, G. Y., and Khusainova, A. M. (2021). Corpus of written tatar.

Silfverberg, M., Ruokolainen, T., Lindén, K., and Kurimo, M. (2014). Part-of-speech tagging using Conditional Random Fields: Exploiting sub-label dependencies for improved accuracy. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 259–264, Baltimore, Maryland, June. Association for Computational Linguistics.

Solorio, T. and Liu, Y. (2008). Learning to predict code-switching points. In *Proceedings of the 2008 Conference on Empirical Methods in Natural Language Processing*, pages 973–981, Honolulu, Hawaii, October. Association for Computational Linguistics.

Suleimanov, D., Nevzorova, O., Gatiatullin, A., Gilmullin, R., and Khakimov, B. (2013). National Corpus of the Tatar Language "Tugan Tel": grammatical annotation and implementation. 95:68–74.

Taguchi, C., Sakai, Y., and Watanabe, T. (2021). Transliteration for low-resource code-switching texts: Building an automatic Cyrillic-to-Latin converter for Tatar. In *Proceedings of the Fifth Workshop on Computational Approaches to Linguistic Code-Switching*, pages 133–140, Online, June. Association for Computational Linguistics.

Tyers, F. M. and Washington, J. N. (2015). Towards a free/open-source Universal-Dependency treebank for Kazakh. In *3rd International Conference on Turkic Languages Processing, (TurkLang 2015)*, pages 276–289.

Tyers, F., Washington, J., Çöltekin, , and Makazhanov, A. (2017). An assessment of Universal Dependency annotation guidelines for Turkic languages. 10.