Research Article

Machine-Based Transliterate of Ottoman to Latin-Based Script

Ashti Afasyaw Jaf [iD] [1,2] **and Sema Koç Kayhan**[1]

[1]*Department of Electrical and Electronics, University of Gaziantep, Gaziantep 27310, Turkey*
[2]*Department of Computer Science, College of Science, University of Garmian, Kalar 46021, Iraq*

Correspondence should be addressed to Ashti Afasyaw Jaf; ashti.a@garmian.edu.krd

In this paper, a machine-based transliterate is presented. The automatic transliteration of Ottoman to the modern Latin Turkish script can open a big window for scientists in fields of history and literature while most of the Turkish people are not familiar with Ottoman script, despite the fact that no concrete solution has been proposed yet for this issue. The proposed method includes several steps since the transliteration process of Ottoman alphabet to Latin base consists of many problems; the first is the basic character mapping which covers the regular pronunciation and orthography mapping. On the other hand, covering other irregular and extraordinary cases is based on rules and normalization. The transliteration system achieved 73.9% accuracy in general.

1. Introduction

Transliteration is a process of transforming the script of a language to another script for the same or different language. Unlike translation, however, the meaning does not change. For example, the word data in English is "deita" in Korean and "deeta" in Japanese when it is transliterated [1]. The $X_{input} = a_1, a_2, \ldots, a_n$, is mapped directly into $Y_{output} = c_1, c_2, \ldots c_n$ while X and Y are two different scripts. Although the automatic transliteration looks an easy and straightforward process, there are different exceptions.

In general, transliteration faces different challenges in the forms of use; pronunciation varies according to the morphological differences of the language. In other words, all sounds are not deployed in one script. In addition, the power of the language script is another issue related to transliteration. This forces us to use more than one letter to represent a letter in the opposite script or vice versa. This leads us to loss information if our transliteration method does not have enough efficiency [2].

The Ottoman script had been used in Turkey, as a main script, for more than 700 years, since the 13th century until the beginning of the 20th century. However, during the last century (20th C.), the Turkish language fell in a dramatic update.

Alphabets of expanded Arabic were deployed in the Ottoman script. There are many characteristics of Ottoman script that adopted from Arabic and Persian languages, even the 28 letters. The writing was from right to left. Some grammatical and morphological uses of other regional languages can be found in the Ottoman scripts, but the main characteristics are still like the Modern Turkish language [3, 4]. The complexity of Ottoman orthography, its exceptions, and differences from modern Turkish language make it difficult to be understood or used by the majority of the contemporary Turkish citizens [5]. The other important factor in transliteration is pronunciation. In this direction, the representation of vowels in Ottoman script is embedded sounds [6].

Mainly transliteration is needed for several reasons such as education, the development of the language, and the historical, political, and regional documentations. Moreover, related to NLP, many applications of information retrieval, translators, and talking applications use transliteration. In the case of the Ottoman script, documentation is considered as the main reason for transliteration. Although the problem of transliteration of Ottoman to Latin-Turkish has been studied and analysed in several computational studies, including [5, 7–9] as a core issue, a concrete practical method has not been implemented yet.

On the other hand, dealing with Arabic or Perso-Arabic scripts in any NLP application contains many challenges and issues; in our transliteration system from Ottoman Turkish (OTr.) to Latin Turkish (LTr.) scripts, the noticeable challenges are obvious in the following points.

1.1. Mapping Scripts (Many to One). In the transliteration process, a phoneme in the input script can be mapped into a single phoneme in the target script. In our case, most of the OTr are character pairs or triples mapping to a single character; for instance, {"1] {"ح", "خ] are represented by the letter "*h*," {"ص," "س," "ث"} to "*s*" and {"د," "ط," "ض"} to "*d*." Table 1 shows the remaining corresponding letters. This may ease our procedure and reduce the complexity while back transliteration is not needed here, which could be difficult to deal with in this situation.

1.2. Mapping Script (One to Many). The tricky part in transliteration occurs when there are multiple mapping options for a single phoneme. Some characters in OTr can be represented by more than one phoneme. In case of "و", it can be mapped to {"o", "u", "ü", "ö"} [1], the word "وطوس" which means "willow" can transliterate to ""Sogot," "sugut," "sögöt," "sügüt," "sogut," "sogüt," "sogöt," "sugöt," "sügüt," "sugot," "sögut," "sögot," "sögüt," "sügot," "sugut," and "sügöt" as 16 different versions; the correct one is söğüt. This case is repeated for "ى" which is mapped to {i, ı}, "ه" to {"h" or "e" or "a"}. Table 2 shows these cases.

1.3. Unicode of Ottoman Transcript. The Arabic Unicode is used to represent the alphabet as the script source is Arabic and Persian. Unicode of $(U + 0600–U + 06FF)$ represents extended Arabic Unicode to represent other languages as well; one of the script properties is the joint writing cursive(Tashkil); for instance, the memory representing of (ه ه ه هههه) ,ه)))is different while they have the same letters,the Unicode for (ه) is $(U + 200C)$ but the connected (ـه) is $(U + 2013)$[10].

Some letters in OTr act as a variable when transliterated. According to their position in the word, they behave like a vowel or consonant; the letter "ع" "ayin" pronunciation appears like "a, ı, i, u, ü, ö, or O) at the beginning of the words; "ilim" came from "علم" which means science; hence, and "arz" from "عرض" means Earth. "ع "example in other positions is shown in Table 3.

In the same way, "ء""Hemza" does not appear in one way; although it is different from "Ayin" as it does not come alone in the top of "ا، و، ي", it is transliterated to "ü, i, e" according to the position or even omitted sometimes.

1.4. Language Update. Another issue yelling for the transliteration of the language similar to the Ottoman Turkish language is the rich vocabulary and complexity, the language brought vocabulary and syntax from Arabic and Persian

Table 1: Ottoman character represented by Latin (many to one).

OTr	LTr
ج،ض	ç
د، ط، ض	d
ط ،غ	g
ك ،غ	ğ
ه، ح،خ	h
ك، ق	k
ب، ث	p
ث،ص،س	s
ط ،ت	t
ظ،ض،ز	Z

Table 2: Ottoman represented in Latin script (one to many).

OTr	LTr
و	v, *o*, ö, u, ü
ة	h, *e*, a
ى	y, ı, i
گ	k, g, g
غ	g, g

Table 3: The transliteration variety of "ع" and "ء".

Position	Ottoman script	L-Turkish	Meaning	Type
Beginning	علم	Ilim	Science	Vowel
Beginning	اترج	Ücret	Fee	Vowel
Middle	تعطيل	Tatil	Recess	Vowel
Middle	مسئله	Mesele	Issue	Vowel
Final	وعن	Nev	Type	Omitted
Final	جزء	Cüz	Piece	Omitted

beside the Turkish language itself, and the languages come from different family origins thus generating incompatibility and multirules following language in spelling and pronouncing [11]. Beside the update of the Turkish language scripts from the OTr to LTr, there was a continuous update for the language since 1929 [12, 13]. It is worth noting that Table 4 includes related examples. The current work excludes dealing with the language update; the OtoL transliteration tool covers the basic transliteration for now without dealing with updated words.

In our project, a transliteration method from Ottoman Script to Modern Latin Turkish Alphabet is implemented; the method mainly depends on rules used in manual transliteration mixed with normalization steps and thus had been proposed before in [5, 8] without implementing or showing results. The first section summarizes the related work which concentrates on Turkish transliterated models. In Section 2, the Turkish written system is briefly explained. The third section clarifies the proposed system in detail; for simplicity, we call our system OtoL. Section 4 sheds light on the normalization level and error corrections, followed by the experiment results. Finally, the last section implies the conclusion of this work, and some tips for the future works have been suggested.

Table 4: Examples of language update.

Ottoman text	Transliteration	Current Turkish	Meaning
حال ناسل	Lisan hal	Hal dili	State language
قيل و قال	Kil u kal	Dedikodu	Gossip
منفى	Manfi	Olumsuz	Unfavorable
وارہا	Avare	Boş gezen	Wanderer
هستہا	Aheste	Sakin Sakin	Slowly

2. Related Works

Many natural language processing (NLP) applications use transliteration as a basic step, including translation, extracting terminologies, and intralingual data linking. In general, transliteration uses phoneme or a grapheme as a base for its procedures, both models showed different performances depending on the data sets with high heel for a grapheme-based method [14, 15]. On the other hand, the authors in [16, 17] deployed a phoneme-based procedure for this purpose. However, the authors in [18] implemented supervised and semisupervised models in their transliteration approach while the authors in [19] used finite state automata (FSA) as a named entity transliteration.

Since the dramatic change in Turkish script century ago, the transliteration techniques have been tried manually. One of the complete documented resources in this direction is [6]; in more than 34 pages, the author explains all theoretical the details and importance of the transliteration with examples. Nonetheless, some updates occur on the modern Turkish script and language, but the document is still precious. One of the first proposed frameworks was [8], and the authors summarized their suggested method by using a pipeline of natural language processing techniques, morphologic parsing, transliteration, dictionary, morphological synthesis, word disambiguation, error handling, and detecting nouns respectfully. The reserved dictionary included around 30000 words, and according to authors, this should be sufficient for newspapers and magazine transliteration. The followed approach was [5], after reporting the basic challenges and the need for the transliteration method explained. The authors were applying partial parsing for the text, getting its root, transliterating the root, and finally searching in two dictionaries conducted, a limited dictionary from Ottoman Turkish to Modern search directly and a more extensive Modern Turkish word list in which searched using a regular expression to generate from the Ottoman spelling of a presumptive root to reconstruct the text following modern Latin Turkish. Reference [7] is the final published attempts for language update, implemented supervised neural network. They used a corpus of old and modern Turkish books to train the system. The study is distinguished from previous ones by reporting its results. Theresults are related to a part of transliteration process which is normalization but achieving 33.8 BLUE score point for RNN neural network. The summary of the article which is related to the area is shown in Table 5.

2.1. Turkish Writing System. In fact, the Turkish language has an important role in expressing the ideology and the construction of the national identity. In the Ottoman period, as the Islamic government was in role, upgraded Arabic-Persian script was used as a writing system for the Turkish language. With the establishment of the modern Turkish government, Latin alphabets were started to use from the beginning of 1929 [20]. The Ottoman script holds the main characteristics of Arabic script, written direction right to left, letter format according to their position "beginning, end, middle, and isolated"; vowels are limited "almost all letters are consonant"; therefore, the context has a big role in reading and meaning while the number of letters has an updated version from the Arabic 28-alphabets with four Persian letters (ذ،ض،ث،ط), which are shown in Table 6 [21, 22]. For simplicity, we use OTr. to represent the Ottoman transcript and LTr. for Latin Turkish transcript.

On the other hand, the modern LTr system extended to have 29 letters by adding (Ç, G, I, I, Ö, S, Ü), seven modified letters to the original twenty-two letters. The important property of this written system is the representation of the sounds as they are not embedded unlike the Otr, which affects the transliteration efficiency and complexity. Other properties include written direction "left-to write" and capitalization in specific cases [6].

It is worth mentioning that our transliteration method is a single direction model from OTr to LTr. The reason behind that is to reduce the complexity while the OTr abounds for written or use only. It is useful to retrieve the old Ottoman archives and documents.

The materials and methods section should contain sufficient detail so that all procedures can be repeated. It may be divided into headed subsections if several methods are described.

2.2. Proposed Method. The proposed method OtoL is straight forward excluding any kind of dictionary or learning approach for now as shown in Figure 1, beginning with an optical character recognition; since the Ottoman documents that are available online are limited to image text resources, the current version of the OtoT uses an online software (http://miletos.co/en/showcase/ottoman-ocr) for this task (Figure 2). We do not deal with entire detail of the OCR process and the method used as far as we manually checked the validity of the OTr got. On the one hand, the data arranged in text format are converted to Unicode format, to remove suspicions between characters that are similar in shape and different in Unicode. On the other hand, to reduce errors and similarities in the current state of the system ("ء"), the hamza in cases (ؤ، ئ، أ) is neglected to be replaced by Unicode base letters (و، ا، a); for instance, the ("مؤمن") transliterated to ("mümin")(" تأريخ ") to ("tarih"), ("ئيسر") to ("reis"); this step also overcomes some complexity of mapping letters.

In fact, the proposed method depends on the phoneme properties of the Ottoman writing to overcome the previous mentioned challenges and struggles in mapping according to the transliteration process; for this purpose, the rules of [6] are used in our method, the rules collected for American library association and the library of the Congress as the Ottoman language were one of the major Islamic language in

TABLE 5: The most related works.

Authors	Method	Year	Results
Birnbaum [6]	Theoretical rules	1967	Not applicable
Kurt and Bilgin [8]	Machine transliteration framework	2012	Not applicable
Korkut [5]	Using dictionaries	2019	Just examples
Nahas et al. [7]	Supervised RNN	2020	Best score 12 over the base line

[6] https://www.jstor.org/stable/597394?seq=1#metadata_info_tab_contents. [8] https://www.researchgate.net/publication/332106732_an_Overview_of_Ottoman_to_Turkish_Machine_Transliteration_System. [5] https://www.semanticscholar.org/paper/Morphology-and-lexicon-based-machine-translation-of-Korkut/35419c804106bf3da509cf36644daeb50f33107b. [7] https://www.researchgate.net/publication/335351552_Supervised_Text_Style_Transfer_Using_Neural_Machine_Translation_Converting_between_Old_and_Modern_Turkish_as_an_Example.

TABLE 6: Ottoman alphabet with equivalent Latin alphabet.

Initial	Final	Medial	Alone	Romanization
ا	ـا	ـا	ا	a, e
بـ	ـب	ـبـ	ب	B
پـ	ـپ	ـپـ	پ	P
تـ	ـت	ـتـ	ت	T
ثـ	ـث	ـثـ	ث	S
جـ	ـج	ـجـ	ج	c, ç
چـ	ـچ	ـچـ	چ	Ç
حـ	ـح	ـحـ	ح	H
خـ	ـخ	ـخـ	خ	H
د	ـد	ـد	د	D
ذ	ـذ	ـذ	ذ	Z
ر	ـر	ـر	ر	R
ز	ـز	ـز	ز	Z
ژ	ـژ	ـژ	ژ	J
سـ	ـس	ـسـ	س	S
شـ	ـش	ـشـ	ش	Ş
صـ	ـص	ـصـ	ص	S
ضـ	ـض	ـضـ	ض	Z
طـ	ـط	ـطـ	ط	T
ظـ	ـظ	ـظـ	ظ	d, z
عـ	ـع	ـعـ	ع	' (ayn)
غـ	ـغ	ـغـ	غ	g, g
فـ	ـف	ـفـ	ف	F
قـ	ـق	ـقـ	ق	k
كـ	ـك	ـكـ	ك	K
گـ	ـگ	ـگـ	گ	k, g, g
-	ـڭ	ـڭـ	ڭ	N
لـ	ـل	ـلـ	ل	L
مـ	ـم	ـمـ	م	M
نـ	ـن	ـنـ	ن	N
و	ـو	ـو	و	v, o, ö, u, ü
هـ	ـة، ه	ـهـ	ة، ه	h, e, a
				y, ı, i

the near eastern world, and the author collected most of the published attempt till that time. Some of them are as follows.

(1) Front vowels (e, ü, *i*, ö) and back vowels (a, *o*, u, ı) follow each other in the syllable, for instance, ("كرام") transliterate to ("kiram"), ("مقبل") to ("mukbil") as shown in Algorithm 1.

(2) Back constants ("ح", "خ", "ص", "ض", "ط", "ظ", "ع", "غ", "ق") are generally followed by back vowels. For instance, "قول" is transliterated to "kul." Algorithm 2 applies this.

2.3. *Data.* The collection of textual data in the Ottoman text is regarded to be as one of the primitive obstacles, as we mentioned before; writing in the Ottoman letters ended about a century ago; to cross these trammels, we do our best to tackle and explore two ways: first: gathering a collection of Ottoman scanned books and articles on the Internet through the archive website (https://archive.org/details/books?query=osmanli&sin=); then using optical character recognition to convert images into texts of the Ottoman script; in this regard, we used a miletos (http://miletos.co/en/showcase/ottoman-ocr). The second resource is a book of
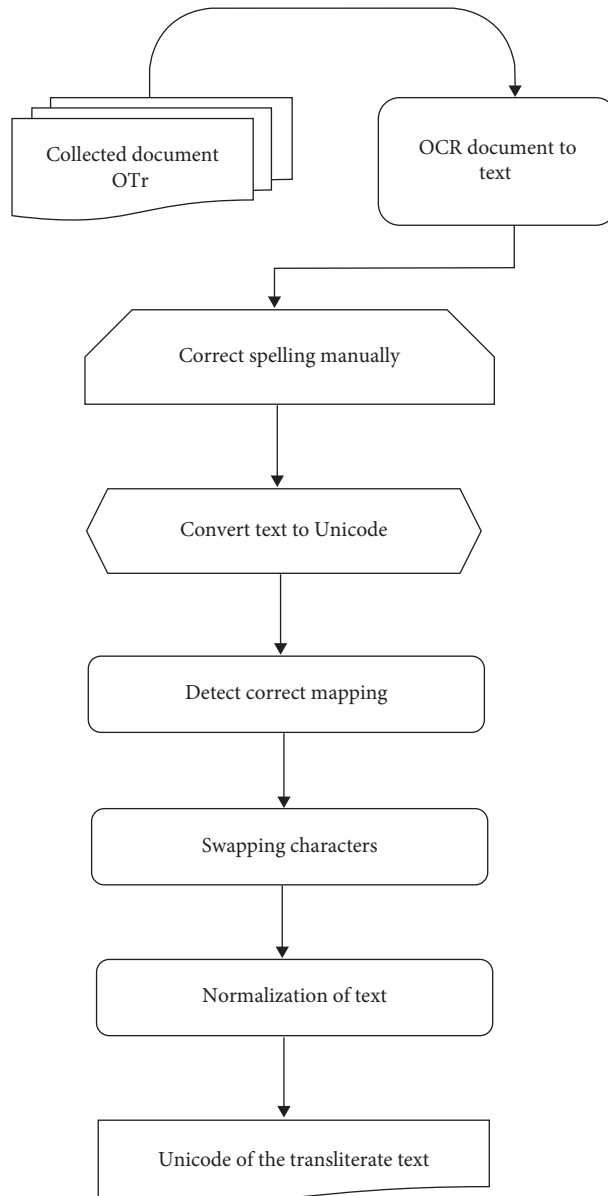
FIGURE 1: The OtoL flow chart.

FIGURE 2: Ottoman OCR system.

```
Input: Word X
Output: b_Vowel, f_vowel,
(1) fv = {e, ü, i, ö}
(2) bv = {a, o, u, ı}
(3) L = length of X
(4) From 0 to length L
(5) If X[i] ∈ {fv}.
(6) If X[i+1] ∈ {fv} Y = X[i+1] swapped to {bv}
(7) Return Y
```

ALGORITHM 1: Detecting front and back vowels LTr.

```
Input: Word X
Output: b_Consonant, f_consonant
(1) fc = {ت، ث، ج، د، ز، ژ، س، ش، ك}
(2) bc = {ح، خ، ص، ض، ط، ظ، ع، غ، ق}
(3) If X[0] ∈ bc
(4) Target word begin with bc
(5) Else X[0] ∈ fc
(6) Target word begin with fc
```

   (1) In the case of the beginning with nonback constant, ending with back constant, and a vowel syllables in between; for example, "مطبوع" is transliterated to "metbua."

   (2) Beginning with nonback consonant, then the next syllable starts with a back consonant, the first syllable vowel is a back one, and "ضار" in OTr is "rıza" in LTr.

   (3) Beginning with "ا", then back consonant, the "ا" is rendered as "i or I", like "ادتصاق" is "iktisad," "اميلق" is "iklim."

   (4) When we use and "و" as a conjunction between words that begin with "ال", "و" is translated as a "ve"; for instance, "القمر والشمس" is "kamer ve elşems".

   (5) Sets of preserved suffix are treated as special case like ("ملى", "مدن", "كن", "لقجه", "لق", "دان", "دق", "مق", "رل", ...). For instance, the plural suffix "–ler or–lar" follows a specific rules: when the final syllable has one of the front vowels {e, ü, i, ö}, the word accept–ler as in "göz" which has become "gözler" meaning eyes, the opposite case like "kuş" which means bird, and the plural case is "kuşlar" birds, as the last syllable have a back vowel {a, o, u, ı}.

ALGORITHM 2: Detecting syllable in OTr.

Ottoman Nutuk, and this book is available already in both forms of OTr and LTr. However, these two versions are not transliterated but are simply translated or rewrote.

For this reason, we build our own OtoT corpus; this corpus contains the text in both of OTr and LTr. The corpus includes 1000 sentences in each part, with an attached scanned document in case of using OCR applications.

3. Results and Discussion

In the case of the current system OtoL, the transliteration is one way, unlike other systems that operate as bidirectional. The main reason for this situation is the uselessness of converting Latin Turkish to Ottoman.

Table 7 shows the results of the machine in transliteration of the OTr to the LTr. The results are shown according to the program steps and the analysis of the percentage of true and false. This method of showing results has not been done in previous studies; they just showed some sample of the outcomes [5, 7, 8]. The results in column two and three show a basic rule-based step, which is detecting the beginning syllable of the words, as far most of the mapping situation depending on this step according to [6], while the last column shows the correct transliteration in the word count.

Since the Ottoman Turkish language derives words and expressions from different languages, especially Arabic and Persian, in addition to the Turkish language itself, a specific law is applied and limiting all rules of writing to it is very difficult; nonetheless, we were keen to apply some rules related to transliteration mentioned in source [6].

In detecting the consonant letters and assigning them to forward and backward classes, since it is a straight forward option, the program did an excellent job, finding all scripts in this regard with an efficiency of 97%. On the other hand, dealing with phonetic syllables is considered a complex case

TABLE 7: The OtoL result analysis.

Number of words	1200	1200	Total trans. test
Correct	Find consonant syllable 759/782	Find vowel syllable 383/451	887
Noncorrect	23	68	313
Percentage	97%	84.9%	73.9%

TABLE 8: Sample of Otol transliteration.

Original text as pic	
Text after OCR	بـر قـطـرﻉ ان جـشـمﻩ پـر خـون فـنـادن باشـن آلﻩﻤاز بـر دخـﻯ بـاران بـلـادن
Manually correction	بـر قـطـرﻩ ایـجـﻨ جـشـمﻩ پـر خـون فـنـادن باشـن آلـﻩ ﻤاز بـر دخـﻯ بـاران بـلـادن
Transliteration by OtoL	bir ktre açin cişme pr hun fnadan başn ale maz bir dhi baran biladan
Manually transliterated	Bir katre içen çeşme pür hûn fenâdan Başın alamaz bir dahî bârân-ı belâdan

in comparing to consonants because the Ottoman writing, like Arabic, contains movements and phonetic connotations. In 451 cases, the OtoT Program was unable to find 68 cases correctly, with a total accuracy rate of 84.9%.

Finally, the OtoL program obtained 73.9% of accuracy in general since the transliteration from the Ottoman to Latin is not direct and there are many deviations and challenges in this regard mentioned before; for example, not restricting, the letter (و) in the OTr can act on its behalf in LTr {"o", "u", "ü", "ö"} in the case of transliteration. Table 8 shows a sample of the OtoL result.

4. Conclusion and Feature Work

The results of the method used in OtoL for the translation of the Ottoman alphabet into the Latin alphabet are reasonable in comparison with the complexities of the language and contrast between the use of Parso-Arabic and Latin letters. Most of the challenges and difficulties in transliteration were taken into consideration, and the straightforward and simplicity of the procedure are the basic merit of this method. Although our method can reach above 91% detecting vowel and consonant syllables, which are the one of the rules for correct mapping from one-to-many characters especially in the cases of {و, {ی}, o}}, unfortunately most of the wrong transliterates are in this area.

To improve the results of the program, the following steps must be applied for newer versions:

(1) Dictionaries are used to improve the performance of the program, especially there are many words that change when transliterated from Ottoman to Latin alphabets, regardless of the continuous updates in the Turkish language to get rid of words of Arabic, Persian, and other origins and replace them with original Turkish words.

(2) Statistical and phonological aspects are used to resolve the case of one-to-many mappings, especially in the cases of {ی}, {و}, and {o} which map to {v, o, ö, u, ü }, {y, ı, i}, and {h, e, a}, respectively.

(3) Bigger and well prepared data set are used; as it is obvious through obtaining the Ottoman writing text, we use OCR process for the images which include small errors and missing some notations.

(4) Heavy normalization should be implemented, especially if we urge to get an updated Turkish language script, this method has been applied in many language reforms and social media comment normalization as in [20, 23].

Data Availability

The data and code are available from the corresponding author upon request (ashti.a@garmian.edu.krd).

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

References

[1] J. H. Oh and K. S. Choi, "An ensemble of grapheme and phoneme for machine transliteration," in *Proceedings of the International Conference on Natural Language Processing*, pp. 542–547, Jeju Island, Korea, October 2005.

[2] J. F. Ozymandias and H. Daumé, *A Generalized Framework for Machine Transliteration*, Maryland University, College Park, MD, USA, 2014, http://honors.cs.umd.edu/uploads/thesis/file/176/Generalized_Framework_for_Machine_Transliteration.pdf.

[3] A. L. Davids, *A Grammar of the Turkish Language: With a Preliminary Discourse on the Language and Literature of the Turkish Nations, a Copious Vocabulary, Dialogues, a*

Collection of Extracts in Prose and Verse, and Lithographed Specimens of Various Ancient and Modern M, Parbury & Allen, London, UK, 1832.

[4] J. W. Redhouse, *A Turkish and English Lexicon: Shewing in English the Significations of the Turkish Terms*, American Mission, Constantinople,Turkey, 1884.

[5] J. Korkut, *Morphology and Lexicon-Based Machine Translation of Ottoman Turkish to Modern Turkish*, Princeton University, Princeton, NJ, USA, 2019.

[6] E. Birnbaum, "The transliteration of Ottoman Turkish for library and general purposes," *Journal of the American Oriental Society*, vol. 87, no. 2, pp. 122–156, 1967.

[7] A. Nahas, M. S. Tunali, and Y. S. Akgul, "Supervised text style transfer using neural machine translation: converting between old and modern Turkish as an example," in *Proceedings of the 27th Signal Processing and Communications Applications Conference (SIU)*, pp. 1–4, IEEE, Sivas, Turkey, April 2019.

[8] A. Kurt and E. F. Bilgin, *The Outline of an Ottoman-To-Turkish Automatic Machine Transliteration System*LREC, Istanbul, Turkey, 2012.

[9] S. Kirmizialtin and D. Wrisley, "Automated transcription of non-Latin script periodicals: a case study in the ottoman Turkish print archive," 2011, https://arxiv.org/abs/2011.01139.

[10] J. D. Allen, D. Anderson, J. Becker et al., *The Unicode Standard*, Mountain view, CA, USA, 2012.

[11] O. languages Woodhead, *The Ottoman World*, Woodhead, Illinois, USA, 2011.

[12] S. Doğançay-Aktuna, "An evaluation of the Turkish language reform after 60 years," *Language Problems and Language Planning*, vol. 19, no. 3, pp. 221–249, 1995.

[13] F. Tachau, "Language and politics: Turkish language reform," *Review of Politics*, vol. 26, no. 2, pp. 191–204, 1964.

[14] Y. A. Onaizan and K. Knight, "Machine transliteration of names in Arabic texts," in *Proceedings of the ACL-02 workshop on Computational approaches to semitic languages*, pp. 1–13, Philadelphia, PA, USA, July 2002.

[15] M. M. Kashani, F. Popowich, and A. Sarkar, "Automatic transliteration of proper nouns from Arabic to English," in *Proceedings of the Second Workshop on Computational Approaches to Arabic Script-based Languages*, Geneva Switzerland, August 2004.

[16] B. Glover Stalls and K. Knight, "Translating names and technical terms in Arabic text," in *Proceedings of the Workshop on Computational Approaches to Semitic Languages*, Montreal Quebec Canada, August 1998.

[17] V. Pervouchine, H. Li, and Bo Lin, "Transliteration alignment," in *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP*, Suntec Singapore, August 2009.

[18] H. Sajjad, H. Schmid, A. Fraser, and H. Schütze, "Statistical models for unsupervised, semi-supervised, and supervised transliteration mining," *Computational Linguistics*, vol. 43, no. 2, pp. 349–375, 2017.

[19] S. Noeman and A. Madkour, "Language independent transliteration mining system using finite state," in *Proceedings of the 2010 Named Entities Workshop*, Uppsala Sweden, July 2010.

[20] O. B. Albachten, "The Turkish language reform and intralingual translation," *Tradition,Tension and Translation in Turkey*, John Benjimins, Amsterdam, Netherlands, pp. 165–180, 2015.

[21] N. Gök, *Osmanlicayi Herkes Kolayca Ogrenebilir Mi*, Zaman Turkish Magazine, Zaman Gazetesi, Istanbul, Turkey, 2004, http://www.zaman.com.tr/haber.do?haberno=96286.

[22] W. B. Barker, *A reading Book of the Turkish Language*, James Madden, New York, NY, USA, 1854.

[23] G. Eryi and D. Toruno, "Social media text normalization for Turkish," *Natural Language Engineering*, vol. 23, no. 6, 2017.