Challenges of Machine Learning with Tamil Texts

from Ancient to Modern Tamil

Vasu Renganathan
University of Pennsylvania, Philadelphia, USA
(vasur@sas.upenn.edu)

Abstract

Digitization and implementation of machine learning algorithms with data from Tamil literature of the three genres namely Sangam, medieval and modern period has been a challenging task mainly due to their complex word structures, formation of multiple shades of meanings historically, and a host of others. The aim of this paper is not only to present methods of storing the existing digitized literature data using recent technologies, but also to devise suitable algorithm to manipulate them byplausible means, especially using relational database and JSON technologies. Without undermining the merits of any of the past technologies of Tamil information ages, including palm leaves, stone inscriptions, copper inscriptions and so on, this paper attempts to try to harness the power of digital technology in a number of noval ways. Our attempt will be to closely look into the power of the Java Script Object Notation (JSON) technology as well as Relational Database Structures along with other ways of manipulationof data using the Vue.Js technology. There have been many technologies to utilize the power of JSON format and relational databases, including Angular.js and others. But, we would like to show in this paper how the Vue.js technology along with the JSON and relational database structures has potentials to search and research vast amount of texts from Tamil's ancient traditions. The sites we would focus on illustrating and developing further in this paper include a) http://sangam.tamilnlp.com/mp/, which offers Tamil literature data in JSON format with a novel filter technique; b) http://sangam.tamilnlp.com/glossing.php, which provides a dynamic link between Tamil lexicon stored in a relational database with that of the texts through glossing technology; c) http://sangam.tamilnlp.com/read_poem.php, which offers a way to contextualize words among Sangam, medieval and modern texts and attempt to lay out a comprehensive historical information;d) http://sangam.tamilnlp.com/, which allows a wide-ranging search techniques across the three genres such as old, medieval and modern Tamil and finally plungingdeep into the morphological tagging possibilities as demonstrated in the site http://www.thetamillanguage.com/tamilnlp/tagit.html.

Tamil data in Java Script Object Notation (JSON) format:

As of now, we have many resources for e-texts of Tamil texts from Sangam, medieval, and modern Tamil in HTML format (cf. http://www.projectmadurai.org/, http://www.tamilvu.org/library/libindex.htm, http://ilakkiyam.com/and a host of others). Due to many constraints involved in HTML and text formats, using these sites for the purposes of machine learning and other novel ways analyzing the text including the use of advanced search techniques becomes harder, although not impossible. However, there are other resources including that of http://www.tamilpulavar.org/api.php, http://www.thetamillanguage.com/sangam etc., which use relational databases such as MySQL, Oracle etc., to store and retrieve data for the purposes of Natural Language Processing as well as search engines. Although the relational databases are popular and very powerful in many senses, they do still require a robust server technology as well as a front-end technology. In comparison to these two different data formats, the JSON format is considered to be more popular than the other formats for the important reason that it can be used from a client side processing without having to rely on any server side technologies. The site http://sangam.tamilnlp.com/mp/ that is developed as part of this work consists of a number of Tamil Sangam literature works in JSON format.

A simple JSON record would be as below:

```
[ {   "Number": 1,
      "Line1": "அகரமுதலஎழுத்தெல்லாம்ஆதி",
      "Line2": "பகவன்முதற்றேஉலகு.",
      "Translation": ""The sound ‘a’ superceds all linguistic sounds; the primeval superceds the world",
      "mv": "எழுத்துக்கள் எல்லாம் அகரத்தை அடிப்படையாக கொண்டிருக்கின்றன,  அதுபோல
உலகம் கடவுளை அடிப்படையாக கொண்டிருக்கிறது.",
      "sp": "எழுத்துக்கள் எல்லாம் அகரத்தில் தொடங்குகின்றன; (அதுபோல) உலகம் கடவுளில்
தொடங்குகிறது.",
      "mk": "அகரம் எழுத்துக்களுக்கு முதன்மை; ஆதிபகவன், உலகில் வாழும் உயிர்களுக்கு
முதன்மை",
"transliteration1": "akara mutala el̲uttellām āti",
      "transliteration2": "pakavan̲ mutar̲r̲ē ulaku"
   }]
```

As can be seen from this example, each line from Thirukkural along with its other related information such as translation, transliteration, commentaries from different authors are noted with a key so, they can all be accounted for programmatically using the ‘key’ vs. ‘value’ pair of object notation. In principle, each of these records in this type of notations can further be illustrated with sub-notations in any imaginable recursive manner. More recursive any JSON string can be more indepth information they can be stored in. So, any program that is designed to read these lines would be capable ofprocessing them in a number of different complex ways possible.

The site under discussion employes the Vue.js technology (https://vuejs.org/) to manipulate and process data from Tamil literature stored in JSON format. JSON format, which adheres to object notation techniques, can relate data in a number of different relational possibilities and thus making the machine to identify the relations between words, phrases etc., in a coherent manner possible. Further, the search engine that is built in this site allows us to search both from lemma as well as inflected forms. To cite one example, searching “நுத” can fetch instances where this word is used in the forms such as நுதலும், நுதற், நுதி etc. Further, this format can hold comparatively large amount of data in a single client side page and searching through the text is also relatively faster.

This site can further be enhanced in such a way that it can be allowed to fetch records that are historically relevant by linking more JSON text of literature belonging to different periods of time. The three genres of Tamil namely Sangam, medieval and modern Tamil underwent a massive number of changes in word senses as well as in the context of formation of new words. In order for one to make an extensive research in such historical contexts, one needs to store text from the three genres as provided in the site http://sangam.tamilnlp.com/. Although this site uses Oracle back-end and PHP front-end, it performsthe fetching of records from many combinations within old, medieval and modern Tamil data. The word முகில், for example, when searched from all of the texts belonging to the three period, one can immediately notice that it occurs more in religious literature than in secular literature such as the Sangam text. Similarly, when the word புணை is searched in all of these three genres, one can observe that this word is used more in Sangam literature and less in religious literature.Further this word may be found to be occurring with the meaning of 'boat' throughout in Sangam literature but only used in the meaning of 'tie together' in the context of medieval literature. What one may presume from these two simple examples is that the religious literature can be considered to make use of nature more extensively than the Sangam literature. Similarly, the use of the technology surrounding the word புணை is observed more with the livelihood of people during the Sangam period than those during the medieval period. Obviously, this type of analyses is possible only with e-texts stored either in JSON or RDBS formats, but neither the HTML or text formats would envisage such opportunities. For further details about historical research conducted for Tamil using this type of relational databases see Renganathan (2009) and Renganathan (2010).

Glossing technology:

Glossing is a process that is used widely in computer assisted learning and teaching. Additional interpretations of any word or phrase that occur within any e-text can be offered by linking the word with an electronic dictionary. Usually, the 'div' structure that is used in HTML technology is employed for this purpose extensively. When the pointer of a mouse is placed on any word within a text, theJavascript code written in AJAX technology is capable of consulting the electronic dictionaries stored in server side relational databases and fetch the dictionary entry in an asynchronous fashion. These technologies are used to read any Tamil text of all the three genres by consulting the Tamil lexicon stored in Oracle database. This is evident from the site http://sangam.tamilnlp.com/glossing.php. The advantage of this site is that it serves as an API to read any e-text from any site simply by adding the url in GET format as in 'http://sangam.tamilnlp.com/url_gloss.php?url='.

1)http://sangam.tamilnlp.com/url_gloss.php?url=export/etext_copy/aacaarakkoovai.txt
2)http://sangam.tamilnlp.com/url_gloss.php?url=http://www.projectmadurai.org/pm_etexts/utf8/pmuni0008_01.html

In this context, almost all of the e-text can be read consulting the Madras University Lexicon in a dynamic manner. Further, the glosses that are fetched in this page is not restricted to just the head entries of the lexicon, but all of the records where particular word occurs within the lexicon isalso fetched and displayed as part of the gloss. One advantage of this method is that when a particular word is inflected and the corresponding form is not available in the dictionary as a head word, fetching the related instances of using the inflected word in the other part of the dictionary can throw further light on the word. This way, even the inflected forms can still be glossed with the help of this technology. A comprehensive machine learning algorithm can still be devised to split inflected words into corresponding lemma so the corresponding entries can be fetched dynamically. Development of such a tagger can be possible for modern Tamil texts than forSangam Tamil for the reason that in Sangam Tamil texts the words occur in manyconvoluted combinations (ex.அவளிவளுவளெவள்)and in unusually split word forms based on meter(ex. இலனதுவுடையனிதெனநினை). (see Renganathan 2016 for a discussion on the limitations of tagging Sangam corpus).

Contextualized References of words and machine learning techniques:

As already mentioned, fruitful use of electronic Tamil data lies in the way how we store them and how we retrieve and synthesize them. Subsequently, there lies the element of machine learning by which the machine is made to learn from the algorithm we write, rather than following the algorithm

itself in a sequential manner. In other words, the fundamental idea behind machine learning is that the machine must be able to make new algorithms through the already available algorithms and constantly build its repertoire of knowledge. Although what we discussed so far in the context of JSON format and Glossing technology can not truly be defined as machine learning techniques as such, they can still be considered as the basis of machine learning for the fact that they offer a fundamental infrastructure to build such machine learning systems. Successful machine learning systems can be built more easily when the data is in a structured format, as in JSON or RDBM rather than in text or HTML format. In this context, this section attempts to discuss a developed system that is written in the PROLOG language using a list manipulation technique. The system in question can be tested in the url:
http://www.thetamillanguage.com/tamilnlp/tagit.html

The main idea behind this system is to use the concept of set theory to make the machine learn from a list structure. When a sentence in Tamil is given as input, this system parses the words and make a list structure with all the suffixes tagged using a predefined set of tags. For example, when the sentence **ஒருவகை சிவப்பு எறும்புகளுக்கு இறக்கைகள் கொண்டு பறக்கக்கூடிய வசதி வாய்ப்பு இருக்கும்.** is made as input, this system identifies the phrases, suffixes and other information such as noun, verb etc., using its dictionary and algorithm and consequently builds its database in a list form as in:

```
[["adj","oru"],["nom","vakai","noun"],["nom","civappu","tr"],["dat","eRumpu","tr","pl"],["nom","iRa
kkai","tr","pl"],["nom","koNTu","tr"],["pa_ajp","paRakkakkuuTu"],["nom","vacati","noun"],["nom","
vaayppu","noun"],["pr","iru","neut.sg","conj"],["nom",".","period"],["nom",".","period"]]
```

With is structure in memory, when posed with questions such as **யாருக்கு வசதி வாய்ப்பு இருக்கிறது? எறும்புக்கு என்ன இருக்கிறது?** and so on, this system parses the input questions into corresponding list forms and attempts to match them with the existing database of list structures by employing PROLOG’s logical operators such as ‘subset’, ‘sublist’ and so on. Subsequently, it gets the correct answers based on the matches it finds (see Gazdar and Chris Mellish 1989 for a discussion on list manipulation techniques using PROLOG). In a sense, this concept of responding to structures based on the list manipulation technique is identical to how human interprets natural language sentences (cf. Johnson-Laird 1983). Human’s understanding of sentences, obviously, go beyond list structures, and uses many complex semantic analyses including presupposition, hyponymy, implications etc. The fundamental concept that is intended to illustrate here is that any successful machine learning algorithm for natural languages can or should begin in this type of list structures and build upon them successively by incorporating more semantic as well as syntactic knowledge in the form of list structures.

This system when posed with a simple question **என்ன?** would fetch all the information from the database and offer respective answers in Tamil sentences. What is unique about this system is that it is built with suitable algorithms to both decoding of Tamil words as well as generating sentences in a natural language format by employing necessary morphological operations encompasing Tamil morphology. In this respect, this system is adequeately built with morphological and sysntactic knowledge base of Tamil. However, what is lacking, perhaps can be developed further, is a sound semantic knowledge encompasing all possible word senses in the form of such semantic information like ‘presuppostion’, ‘hyperonymy’, ‘hyponymy’, ‘synonymy’, ‘polysemy’ etc., which mainly play a crucial role in the interpretation of natural language sentences. (see Renganathan 2016 for the interrelationship between syntax and semantics in the context of interpretation of human languages by machine).

Conclusion:

This paper, on the one hand, is an attempt to harness the power of the technologies of JSON, Vue.js, Relational database, PHP and others to the fullest extent possible, and on the other hand it lays out a comprehensive algorithm as to how these technologies can be exploited within the context of the data from the three genres of Tamil to store, retrieve and synthesize. In essence, this research is a continuation of my ongoing thrust to empower Tamil data with emerging digital technologies, and in no sense it can be considered complete. Particularly, the list manipulation technique that is described in detail in this paper and in my earlier works need fullest and continued consideration in order to foresee a comprehensive machine learning application that can interpret Tamil sentences like any human. Tamil is a complex language in many respects and its mirage of complexities can be accounted for only when the power of technology and the extensive and indepth linguistic knoweldge of Tamil can cross their paths in a productive manner.

References:

Gazdar, Gerald and Chris Mellish. (1989). *Natural Language Processing in Prolog*. Addison-Wesley Publishing Company: Wokingham.

Johnson-Laird, P. N. (1983). *Mental Models.* Cambridge: Cambridge University Press.

Renganathan, Vasu (2016) *Computational Approaches to Tamil Linguistics*. Cre-A., Chennai.

______________________(2014). "Computational Phonology and the development of Text to speech application for Tamil". Paper presented and published in the proceedings of the Tamil Internet Conference, 2014. Pondicherry University: Pondicherry. (http://text2speech.tamilnlp.com/).

______________________(2013). "தமிழை அறிய கணினிக்கு எத்தனை விதிகள் வேண்டும்". Paper presented and published in the Proceedings of the Tamil Internet Conference, 2013. University of Malaya: Kualalumpur, Malaysia.

______________________ (2010) "Evolution of Tamil grammatical suffixes and writing Historical Grammar for Tamil" (In Tamil), Paper presented at the World Classical Tamil Conference, Coimbatore, India.

______________ (2009) "The Process of Grammaticalization and Evolution of Modern Tamil Forms". Paper presented at the Prof. Agesthalingom Commemoration conference, Annamalai University, India (August 19th to 21st, 2009).

______________ (2002) Interactive Approach to Development of English-Tamil Machine Translation System on the Web", in Proceedings of the Conference in Tamil and Internet, Foster City, San Francisco.

______________ (2001) "Development of Morphological Tagging for Tamil", In Proceedings of the International Conference on Tamil Internet 2001, Kuala Lumpur, Malaysia.

______________ (1997) "On Significance of Creation of Modern Tamil Corpora on the Web" paper presented at the First International Conference on Computerization of Tamil. National University of Singapore. May, 1997.