



Binârî: the Artificial Ghazalkhwan

A Poetry Generation System

Galip Ümit Yolcu, Advisor: Tunga Güngör

Boğaziçi University, Computer Engineering Department, CmpE 492 Senior Project

Introduction

- Ghazals are composed of semantically independent couplets, obeying a specific rhyme scheme and rhythmical metre called *aruz* metre
- The subject is the poet's love for the beloved
- Following Manurung [1], more formally, a ghazal is **grammatically** correct, **meaningful** and it satisfies the constraints on **poetic** form: the metre and the rhyme scheme
- **Problem Statement:** Given a rhythmic metre and rhyming word(s) to use at the ending of lines, generate a couplet in Ottoman Turkish satisfying the criteria of grammaticality, meaningfulness and poeticness.

Tahammül mülkünü yıktın Hülâgû **Hân mısın kâfir**
 Aman dünyâyı yaktın âteş-i süzân **mısın kâfir**

Figure: An example couplet by Nedîm

Method

- Take Hafez [2] as basis
- Generate a finite state transducer(FST), like shown in the figure, such that a path from the initial state to the output state traces a couplet that obeys the poetic constraints.
- Train a recurrent neural network to select good paths from the FST, using beam search.
- Reverse the FST, generate poem starting from the last word, going towards the first word, in order to generate words conditioned on the selected rhyme words used at the end.

Data

- We use data created in the Ottoman Text Archive Project[3]. Data is very limited: only 9385 couplets from 3 poets

First Model

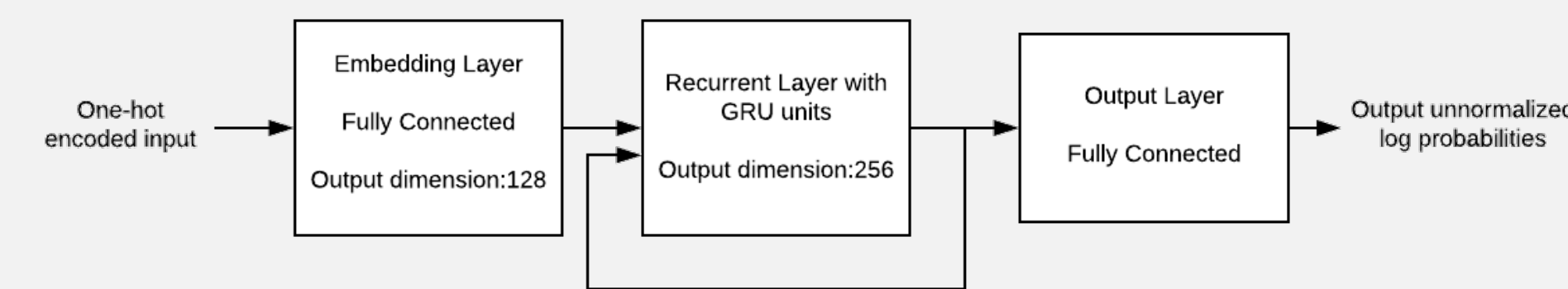
- Due to the agglutinative structure of the Turkish language, we used simple subword tokens(characters and syllables) to train our RNN.

Word Tokens: ["muhabbet", "mülkünü", "yıkdıñ", ...]

Syllable Tokens: ["mu", "hab", "bet", " ", "mül", "kü", ...]

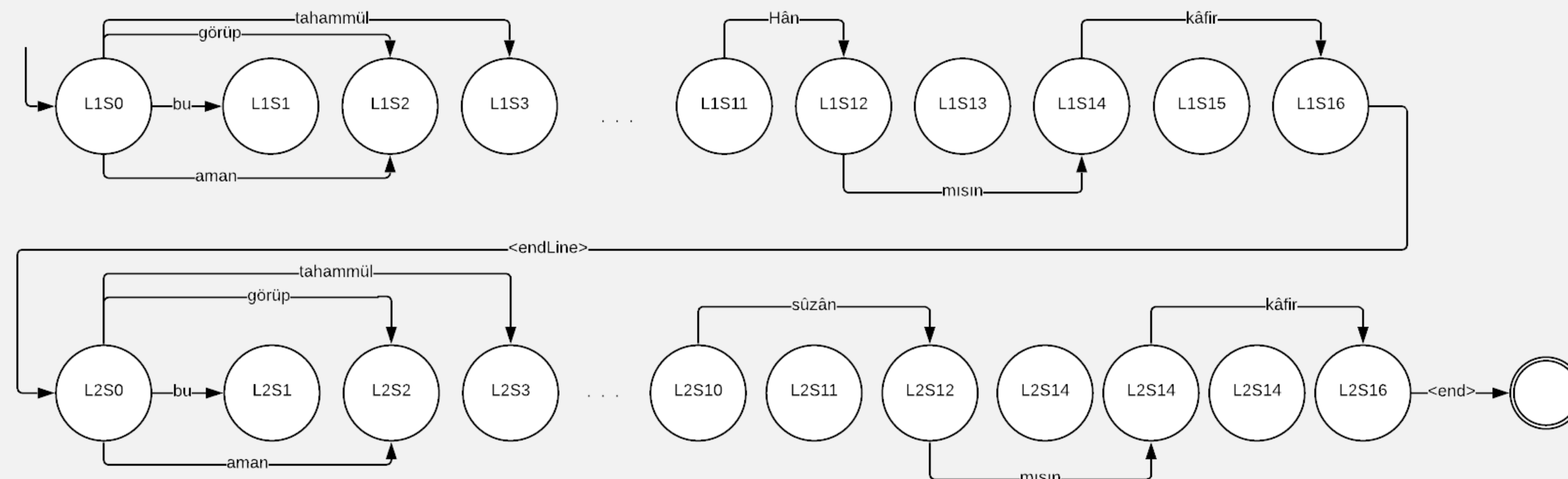
Character Tokens: ["m", "u", "h", "a", "b", " ", "e", "t", " ", ...]

- The neural network architecture is as follows:



Second Model

- Notice that the RNN is constrained to use the words already existing in the vocabulary
- This means bad performance with little data in an agglutinative setting
- This also ensures every word in the output will be an actual word
- We try taking a RNN trained at the syllable level and constraining its syllables according to the rhythmic metre. We continue by generating one syllable at a time.
- This is conceptually equivalent to adding a transition in the FST for each legal syllable, instead of each word



Results

- **First Model, Character Level Training:**

	Rhyme Constraints: cân, süzân	No Rhyme Constraints
500 epochs	ne it hûbân-ı hîffet zevkî hîffet mehlîkâ-yı cân ki töz hîffet idin 'ışkumda 'ışkum 'ışkunû süzân	ku-ı 'ışkıñ diyen hõrşiddin hõrşid 'ışkumda ki bať hõşdem-i içsem 'ışkunû hõrşid 'ışkumda
1000 epochs	ne hoş-ı fahr hayrân eyleyüp şeydalanursın cân bu bať-ı haşm koc hûbân koyub-ı vaşidan süzân	bu hûn-ı hışm-ı hûbân felekler gülbeşekkerdür ne boş hûbân-ı hayrân hayr-ı mahmud ibrâm

RNN seems to be stuck on certain letters and words

- **First Model, Syllable Level Training:**

	Rhyme Constraints: cân, süzân	No Rhyme Constraints
500 epochs	şu -ı -ı derc akşemseddinûñ ekşitdügünñ cân ne boş ir 'ışkudr efsâne akşemseddinûñ süzân	bu az bağdâddan ekşitdügünñ âbdârûñdan bu ur gûstâhlık ma'mûresinden bisa'âdetler
1000 epochs	ye boş illâh akşemseddinûñ ekşitdügünñ cân ko zehrâlud illa'llâh müşgâsâsıdr süzân	bu had-ı fenn akşemseddinûñ hõrşidruhsârı di boş agyâra şekkerrizden hõrşidruhsâr

RNN seems to choose long words whenever possible. It eventually has to select shorter words in order to fit the line length.

- **Second model:**

	Rhyme Constraints: cân, süzân	No Rhyme Constraints
500 epochs	ki gûyâ mülkiçün bâleb-i pürçünkim kâlis-i cân şafaķdur hâne-i devlet serâser mest-i süzân	bu tûbâ çünki şol kim gül çerâzın bezmümüzdüb kış şa-yı miñnet anuñ kim anmazın ancaķ kemer dirler
1000 epochs	eger çün seyr hey hey bezmhânûñçün benümçün cân yyâ tãvûsdr gülgün şabâ olmış berâ süzân	'aceb mül teşnedir kâfir muķılmaķ etdi gerlârı 'acebdür bir kâdem şanmañ anuñ efsanedür dirler

Average word length seems natural and the model is not stuck on any words or syllables, it has grammatical freedom, but it generates some nonsensical words.

Discussion

- Poeticness is satisfied.
- Meaningfulness and grammaticality are not satisfied
- The oddities concerning the first model's outputs, noted in the Results section, suggest a problem in the training process. However, letting the RNNs generate text without being constrained by the FST, we see that these oddities do not persist. We see this in the second model's outputs too.
- The language model seems to be learning some patterns
- The main problem is data scarcity
- If we can find a good language model, we can generate poetic couplets with it

Future Work

- First, find or manually prepare more data
- Afterwards we can
 - Use sophisticated word embeddings
 - Increase model complexity
 - Find better subword tokens to accomodate the agglutinativity of Turkish language
- in order to better satisfy grammaticality and meaningfulness.

References

- [1] H. Manurung, "An evolutionary algorithm approach to poetry generation," 2004.
- [2] M. Ghazvininejad, X. Shi, Y. Choi, and K. Knight, "Generating topical poetry," in *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pp. 1183–1191, 2016.
- [3] "Ottoman Text Archive Project."
<http://courses.washington.edu/otap/>.