A Generic Tool for Visualizing Patterns in Poetry

Onur Musaoğlu[1], Özge Dağ[2], Özge Küçükakça[1], Alkım Almıla Akdağ Salah[3], Albert Ali Salah[1]

[1] Boğaziçi University, Computer Engineering
[2] Boğaziçi University, Cognitive Science
[3] Istanbul Şehir University, College of Communications

1 Introduction

Visualization of text can be a useful exploration tool for looking at the corpus of a poet, especially when dealing with a prolific author with a large body of output over the years. In this work, we describe a flexible and extensible tool for analysing the corpus of a poet, and make a case study of Nâzım Hikmet Ran. Since poetry has its own challenges over plain text, we have developed novel ways of visualizing the structure, the rhythm and affective tone of each poem, as well as ways of looking at the continuities (or discontinuities) of features in the entire corpus over the years. The designed system integrates a database for holding meta-information, and a website for creating and linking interactive, parametrized visualizations.

Nâzım Hikmet Ran is one of the most famous poets of Turkey. Although he was a great patriot, he has spent many years in prison and in exile due to his communistic political views. His poetry is translated into more than fifty languages. We believe this tool can be particularly useful to compare different translations of the poet's work, to see how certain stylistic or semantic features are retained (or lost) during translations.

2 Related Work

Most poetry visualizations focus on the aesthetics of information rather than the functional aspects. An example is Diana Lange's visualizations that transform individual poems into beautiful visual displays, resembling flowers. A similar project is Poetry on the Road, which turns a text into an image through an arbitrary transformation function, for instance by treating every word as a location and creating a heat map of the entire text. The outcome of these projects do not tell us much about the poets or the œuvres in question.

In contrast to such artful renderings of poems, there are studies who focus solely on grammatical and structural problems of poetry writing. Such studies rather try to find quantitative ways to analyze poems, enabling a computational approach for the evaluation of technical quality and subtlety of the

rhymes [Opara, 2014,Dalvean, 2015]. In the same line of research, we encounter visualization tools such as Graphwave, SentimentGraph, SentimentWheel and Ambiances [Meneses and Furuta, 2015]. Such visualization examples constitute the starting point of our explorations for devising a new visualization system that is both scalable and modular in nature, i.e. a tool that would accommodate different natural language processing (NLP) tools, as well as new visualization techniques.

3 Methodology

In this section, we briefly describe the database structure, as well as the software tools used to create the described system.

3.1 Database

After his death, Nâzım Hikmet's collected works appeared in a single volume [Nâzım, 2007]. The digital version of this volume is not publicly available, but we received a special permission from the publishers to use this volume.

Nâzım is a poet who pays special attention to the visual structure of his poems and we wanted to retain this structure as accurately as possible. Consequently, we kept line indentation for each line, and fonts of the individual words. We also paid attention to the fact that the collected works included text written in prose as well. Thus, the database structure, depicted in Fig. 1, is entirely hierarchical and ordered according to his books, works in a book, lines in a work, words in a line, and characters in the words.

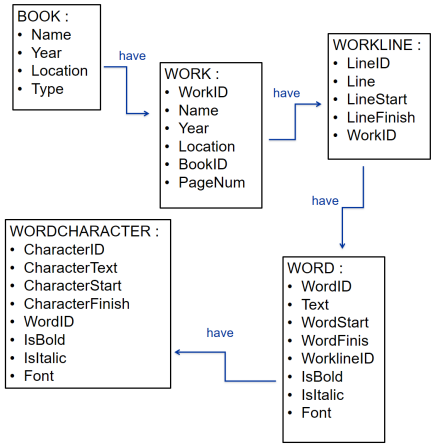

Fig. 1: Database structure,[Dağdelen, 2006]

3.2 Software

As we created a dynamic, parametric and interactive system, we used many software technologies. To keep structured data, user data and web page related content, we used a MySQL database. We employed Perl to read and parse data for the affect analysis tool. The main programing language of the project is Java and all back end code is also developed in this language. We use the Spring Framework to create the model-view-controller (MVC) structure of the application. In the front end of the application, AngularJS is used to create the MVC structure and to create a single page application. Moreover, to make the application responsive, we used CSS3 and Bootstrap technologies for mobile phone support.

3.3 NLP Challenges

In order to parse Nâzım's corpus, we made use of a Turkish Morphological Parser and Disambiguator [Sak et al., 2008]. With the help of this tool, we get part of speech tags of the words, as well as some grammatical information about verbs (i.e. conjugation and tense) and about words' grammatical number. For certain instances, the morphological parser suggests more than one form or number. To solve such problematic instances, we applied another off-the-shelf tool, i.e. the disambiguator. The results of the disambiguation tool suggests the most appropriate form for a given context, which helps in making a decision on the preferred form of a given verb, noun, or pronoun. Additionally, we enhanced our system with a text-based affect analysis tool, which returns valence, arousal and dominance values for a given sentence and each word in that sentence [Aydın Oktay et al., 2015]. One of the challenges in parsing poems versus prose texts is the lack of a specific notation indicating end of a sentence. For the sake of simplicity, we decided to treat each line of a poem as a sentence, and computed valence, arousal and dominance values for every word and line individually. These values are stored in the database for fast retrieval.

4 The System Interface

We generated a system that works as an interactive visualization tool with a web interface. For the user experience of the web system, we prepared a responsive interface that can even be reached via smart phones. Also, to keep data alive and to allow flexible operations, we created a single page application where users can surf between different tabs without losing information. We designed the system in an expandable way. We separated each work unit such that new visualizations can be easily added to the system. The system can also be tailored to visualize a new database easily. The only requirement is that the work of the artist be parsed in the same hierarchical way, and placed on a SQL-capable database.

5 Visualizations

One of the motivation behind the visualizations is to give information about the analysis on the corpus of Nâzım, and other poets when the database is expanded by the addition of new authors. The tool incorporates a search function, and allows different visualizations to be prepared from the results of the search. Most of the prepared visualizations are interactive charts. They can be used for showing a terms usage over the years, over geographic locations and over publications. The search can be conducted on a collection of works, or in a single work. We also created a page for searching for a sequence of words, to prepare comparative visualizations.

We will describe only two visualizations here. The first visualization is called the "poetry barcode" (see Fig. 2). In this visualization, each line of a poem is represented by a horizontal line. The length or the color of each line is set according to NLP and sentiment analysis results, and the lines form six different columns, which show the change of line lengths, usage of active/passive phrases in a line, inflections of words in a line according to person information, valence-arousal-dominance values of each line.

Nâzım has a lot of stylistic features in his poems. To be able to analyze and extract these features, we have prepared visualizations about alliteration and his unique verse structure. Alliteration is a stylistic device, in which a number of words, having the same consonant sounds, occur close together in a series. To quantify alliteration, we used a measure that uses the background frequency of each letter in the poets corpus. By using a sliding window evaluation, letter frequencies are calculated, and compared with the base frequency of that letter in the corpus. Fig. 3 illustrates the automatic alliteration detection. Fig. 4 shows a number of additional visualizations in a bird's eye view.

6 Conclusions

A complete web page opened to the wider public is in construction, as it requires some security features due to copyrights of the works in the database. But the system is operational in the offline mode, and already provides many visualization options.

We created the system in a way that data and visualizations are entirely separate, but easily connected. This will help us to add new artists to our system. The tool also can be used as a platform to evaluate poetry translation. We show grammatical and affective features of the words in some visualizations like "poetry barcode". For example, we can use these visualizations to get an idea about translation quality in literal translations, where the emphasis is on word-for-word translation.

Fig. 2: A poem's barcode, visualizing the structure of the poem together with verb conjugations, passive/active verb usage, and emotional tone.

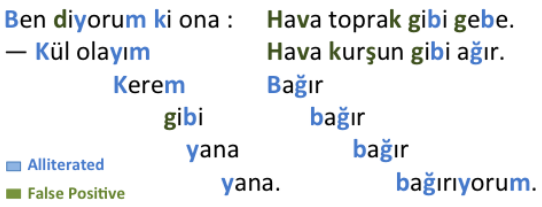

Fig. 3: Visualizing alliteration in the poem. Best viewed in color.

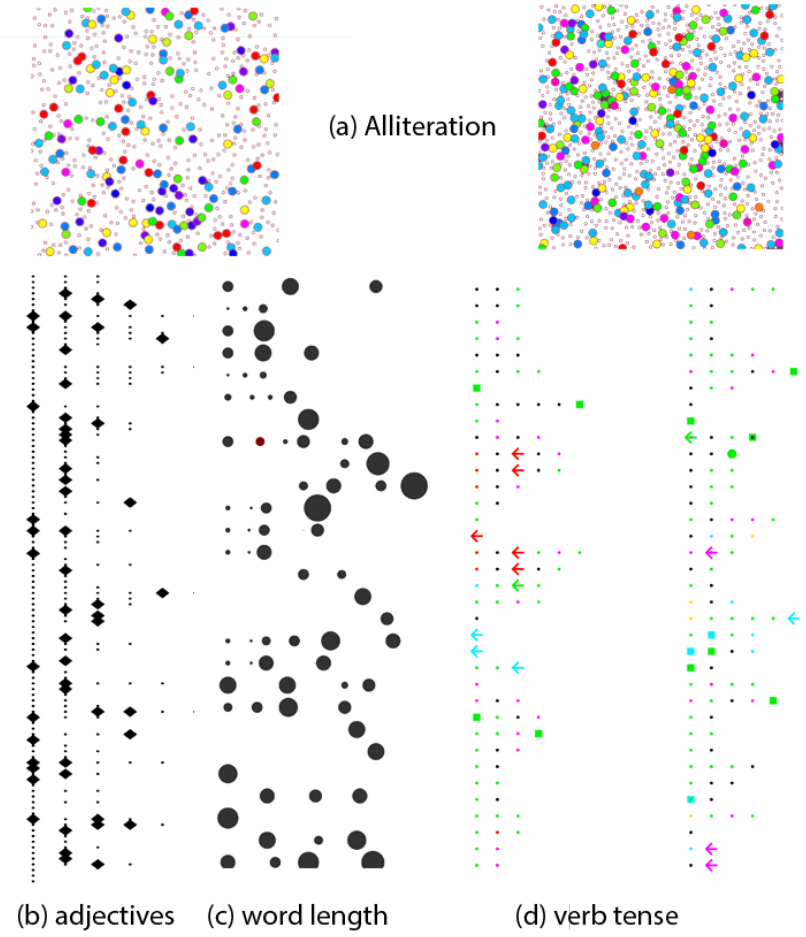

Fig. 4: A birds eye view of several visualization options in the system. Best viewed in color.

7 Acknowledgments

We would like to thank Yapı Kredi Yayınları for giving us permission to use the digital version of Nâzım's works. This study is realized in collaboration with Nâzım Hikmet Kültür ve Sanat Araştırma Merkezi.

References

[Aydın Oktay et al., 2015] Aydın Oktay, E., Balcı, K., and Salah, A. A. (2015). Automatic assessment of dimensional affective content in Turkish multi-party chat messages. In *Proc. Int. Workshop on Emotion Representations and Modelling for Companion Technologies*, pages 19–24. ACM.

[Dağdelen, 2006] Dağdelen, İ. (2006). Alman mavileri, 1913–1914 i. *Dünya Savaşı Öncesi İstanbul Haritaları, I–II, İstanbul*.

[Dalvean, 2015] Dalvean, M. (2015). Ranking contemporary American poems. *Digital Scholarship in the Humanities*, 30(1):6–19.

[Meneses and Furuta, 2015] Meneses, L. and Furuta, R. (2015). Visualizing poetry: Tools for critical analysis. *paj: The Journal of the Initiative for Digital Humanities, Media, and Culture*, 3(1).

[Nâzım, 2007] Nâzım, H. (2007). Bütün şiirleri. *İstanbul: YKY*.

[Opara, 2014] Opara, K. R. (2014). Grammatical rhymes in Polish poetry: A quantitative analysis. *Digital Scholarship in the Humanities*, page fqu029.

[Sak et al., 2008] Sak, H., Güngör, T., and Saraçlar, M. (2008). Turkish language resources: Morphological parser, morphological disambiguator and web corpus. In *Advances in natural language processing*, pages 417–427. Springer.