Persian Poet Identification Using NLP

Sepanta Kamali
Computer Engineering Student at Mazandaran University
Professor: Dr.Roostaee
PhD in Computer Engineering, Professor at Mazandaran University

Abstract

In identifying the identity of the Persian poet, features of Persian poetry and literature are discussed, which lead to learning and identifying the identity of the poet. The presented approach deals with the analysis of the words used in the poem, the rhythm and format of the poem, and by calculating the weighted frequency[1] for these features, it trains a model to be able to recognize the author of the poem. This approach is a new way to identify the identity of the Persian poet, which is inspired by the method used by Persian poetry experts to identify the poet experimentally. In this approach, using text mining techniques, features are extracted from the text of poems and become a criterion for evaluating poetry. Finally, the poem is evaluated from the aspect of these characteristics. Then, the probability is calculated based on these features so that the most likely poet is chosen as the poet. A bag of words is created for words extracted from the text and unique words are considered as features. The rhythms and formats are inferred from the poems in the dataset. Features are added to the training set after preprocessing and TFIDF calculation. In pre-processing, Rule Based technique is used to classify and calculate features. The proposed model is compared in several stages and with several models, including different versions of the proposed model. Several test sets are created and the performance of the model is evaluated.

Keywords: Authorship Attribution, Natural Language Processing, Persian Poem, Weighted Frequency, Poet Identification, Text Mining, Rule Based Classification, Text processing.

1. Introduction

Identifying the author refers to recognizing or guessing the author of a text based on textual features. .The Poet Identification is an Authorship Attribution task. This issue was raised for the first time at the end of the 19th century (1887), when an American physicist named Mendenhall[2] used statistical methods to evaluate writing styles. Later, Mosteller and Wallace comprehensively implemented this concept on a collection of 85 political documents (also known as the Federalist Papers) written by Alexander Hamilton, John Jay, and James Madison. And obtained significant results by applying statistical analysis methods on the repetition of words. This work is based on the assumption that every writer has his own specific style, and as a result, his linguistic and stylistic features will be constant and specific to him in all his writings. Based on this, studies have been conducted on language characteristics and writing styles, which show that the writings of each author have characteristics that are constant in all his writings and sometimes specific to that author, such as the frequency of the words, the length of the words, the length of the sentences and so on. In new approaches, instead of manually extracting these features, computational methods are used. The identity recognition problem is a multi-class and single-label classification problem, whose labels are authors. In this problem, the features that best describe an author's style are extracted using Text Mining and NLP methods. Then these features are used by classification algorithms to predict the author of the text. In eager methods, a classifier is trained once and is used when a prediction query arrives. In contrast, in lazy methods, the model is built when receives a new prediction query. On the other hand, the identity identification approaches are also divided into two groups: profile-based and instance-based.

In the profile-based methods, all the texts of an author are concatenated to create a single document, but in the instance-based methods, each text has a role in the learning process (Reza Ramezani, 2021).

The selected features are extracted from these documents. In Persian poetry, all words have value and meaning. The characteristics of a poem are completely dependent on the poet and his literary style. The choice of words, the rhythm and the format of the poem, all constitute the poet's literary style (Ahmad Samiei, 2004). Persian poems have specific and defined rhythms and formats that play a very important role in identifying its composer (Omid Tabibzadeh, 2010); In other words, Persian poets use certain rhythms and formats more than other rhythms and formats, and even in some cases, some poets only write poems in a specific format or a specific format is used only by one poet. This also applies to vocabulary. Words, rhythms and formats have different values, uses and frequencies for different poets, as a result, calculating the weighted frequency of these features for each poet can provide us with valuable information to identify that poet. In this project, a definition of weighted frequency is presented, which can determine the degree of specificity and frequency of each feature for each poet. The similarity measure is an eager and instance-based classification algorithm, which by analyzing an unknown poem, the probability of its belonging features are calculated and the most probable poet is assigned to that poem.

E-mail address: sepantakamali@gmail.com.

[1] TF-iDF : Term Frequency inverse Document Frequency

The obtained results show that the presented approach has achieved 89% accuracy. This is despite the fact that a similar project that also used the same data and created even more complex and advanced models was able to reach 81% accuracy, that is, the presented approach is 8% more accurate despite its simplicity, which shows the effectiveness of the proposed solution.

2. Related Works

Vocabulary analysis is an integral feature of authorship attribution approaches. These techniques are based upon the extraction of the stylometric features that better discriminate candidate authors. This process requires a feature selection strategy followed by training or using an eager or lazy classifier to identify the most probable author of an anonymous document. In this approach, the core problem is the selection of features that provide uniqueness and best discriminate against the writing styles. Most of the existing approaches create a language model for each poet. These approaches follow the profile-based method, in such a way that for each poet, they create a document containing a large number of the verses of his poems and create a Unigram, Bigram, and/or trigram language model for each poet. Finally, for each unknown verse, the probability is calculated based on these calculated features and the most likely poet is assigned to that poem (Soroush Mehraban, 2021).

In (Amirali Kaboli, 2021) project, another version of the Ganjoor dataset is used, which is almost the same as the dataset used in this project. The data are analyzed in a balanced and unbalanced way and in the form of verse, complete poems and couplets as input instances to the model, on the other hand, the models are created with classic methods such as SGD, SVM and Random Forest as well as Fasttext and LSTM. TFIDF vectors are created by embedding and models are built upon them. According to the report provided by the author, the best output was of the SGD model which was generated on four-couplet inputs taken from the balanced data, which was able to achieve an accuracy of 81%, while the accuracy was in a fairly good range for almost all poets. Other projects mentioned at the beginning of this section, despite their high accuracy, are not reliable, because they are limited to a few poets (usually 3 poets) and are only based on a linguistic model, and their evaluation does not have reliable results. On the other hand, in identifying the poet's identity, we must obtain information about the poet's stylometric features, based on which we can recognize the poet of a poem, and not to obtain a probability for each poet based only on the words used in the poems. There is no collection of poems with a clear and unified structure that can be used for an easier analysis of poems. For this reason, we need to create such a collection. Ganjoor dataset is one of the most complete databases available. To create the desired collection from this database, several stages of data extraction from this database are required. These steps are explained in section 3.3. The text of the poems in this collection is first normalized. Then the text is evaluated until nothing remains except poetry and accepted characters in Persian language. The position of each verse in this collection is clear so that the pattern of the poem can be better examined. The way the verses are placed in the poem indicates the style of the poem, which helps in determining the format of the poem. There are other characteristics that play a significant role in identifying the poet's identity. Radif is a word or words that come at the end of a poem and after Ghafieh, which will be explained later on this paper. Radif plays an important role in poetry. Radif has contributed to the composition of the poem and is the basis for the use of literary features and the formation of imaginary images in the poem. Radif make rhyming easier. Another feature is the emotions. In Persian poetry, the poet evokes one or more feelings in the reader or listener, such as love, joy, sorrow, epic, fear, anger, surprise, disgust, regret, hope, and so on. Each poet deals with some of these feelings more than others. Now, if he has experienced it more or if it has been used purposefully. Even if the meaning of the poems of several poets is the same, the words used and their literary features and the way of writing the poems are different. In Persian literature, there are many literary features, including simile, tashkhis, allusion, irony, pun, metaphor, etc. Each poet uses more or less of each feature based on his poetic style and identity.

3. Proposed Solution

3.1 Feature selection

Considering all that has been said so far, we know that the weighted frequency of the features play a key role in identifying the poetic style and has a significant role in increasing the efficiency of the model. The words used in the poem are important and valuable as the main elements of the poem. Considering that Persian poets lived in different eras of history and the Persian language is a dynamic language that has grown and evolved throughout history, as well as the topics, concepts and words used in different eras are somehow different from each other. Of course, there are concepts that have existed in all eras and centuries. Therefore, it is not possible to say who the poet of a poem is based only on words, so we need another characteristic to add to our knowledge-base. This is the characteristic of poetic rhythms. In Persian poetry, each poem has a specific rhythm, even if this rhythm tends to be rhythmless, in which case there are other principles for that kind of poetry. Each poet uses a series of rhythms more than others according to his poetic style. There are also situations where some poets have their own kind of rhythm. According to these descriptions, this feature is also a valuable feature and can have a great effect on increasing the accuracy of the model. But some poets write poems in similar rhythms, which makes it difficult to distinguish them from each other. Of course, the statistics show that there is a slight difference in the amount of use of these rhythms between these poets, which eases this problem. As it was said, some poets do not follow a certain rhythm or their poems do not have a certain rhythm; Therefore, another feature is needed to add to our knowledge-base. Another key feature is the poetic format. Persian poems have specific formats. Unlike the previous feature, this feature applies to all poems and there is no poem that does not have a specific format, as a result, this feature complements the previous feature. As mentioned, there is a commonality in the use of these features among poets, but their frequency is different. As a result, by calculating the weighted frequency of each feature, we can gain very important information about the stylometric features of each poet.

$$\underset{\substack{feature \in FEATURES \\ poet \in POETS}}{TFIDF}(poet, feature) = \frac{TF(feature, poet)}{DF(feature, poet)}$$

Equation 1. TFIDF

$$TF(feature, poet) = \frac{FF(feature, poet)}{\sum_{feature \in FEATURES} FF(feature, poet)}$$

Equation 2. TF

$$DF(feature, poet) = \frac{\sum_{poet \in POETS} FF(feature, poet)}{\sum_{feature \in FEATURES} \sum_{poet \in POETS} FF(feature, poet)}$$

Equation 3. DF

3.2 TFIDF

The weighted frequency equation or TFIDF defined in this approach, is its simple definition, equation(1). Therefore, the weighted frequency of each feature for each poet is calculated as follows:

TF: frequency of that feature for the poet, to total frequency of all features of that poet.
DF: total frequency of that feature for all poets, to the total frequency of all features for all poets.
FF: Weighted frequency of the feature for the poet.
In this way, TFIDF is calculated for all feature and for all poets. Therefore, when we want to examine the poems of a poet from the aspect of a feature. We have to see how much this feature is used by this poet and what proportion of the poet's poetic features it. Also, to what extent other poets have used this feature. That is, how special is the feature for the poet. In this way, the degree of specificity and frequency of a feature is calculated for a poet.

3.3 Feature Extraction

3.3.1 Database

The database we used for feature extraction is Ganjoor database, which contains 63,791 poems by 72 Persian poets. These poems create a collection of 1649240 stanzas (Ganjoor, Database, 2018).

In total, there are 814189 poems in this collection.

Poet	Number of couplets	Percentage of total
عطار	95015	11.67
صائب تبریزی	78888	9.69
مولوی	66206	8.13
فردوسی	49656	6.1
بیدل دهلوی	33047	4.06
نظامی	27778	3.41
جامی	27688	3.4
سنایی	26081	3.2

امیرخسرو دهلوی	23282	2.86
ملک‌الشعراء بهار	21500	2.64
قاآنی	20495	2.52
امیر معزی	18466	2.27
خاقانی	17432	2.14
سعدی	15754	1.93
مسعود سعد سلمان	15675	1.93
کمال‌الدین اسماعیل	15331	1.88
شاه نعمت‌الله ولی	14980	1.84
اوحدی	14673	1.8
انوری	13308	1.63
سلمان ساوجی	13103	1.61
سیف فرغانی	13009	1.6
فرخی سیستانی	11738	1.44
حکیم نزاری	11158	1.37
ناصرخسرو	10765	1.32
محتشم کاشانی	10715	1.32
وحشی	10595	1.3
فیض کاشانی	9844	1.21
فخرالدین اسعد گرگانی	8981	1.1
خواجوی کرمانی	8849	1.09
اسدی توسی	8808	1.08
رشیدالدین وطواط	8530	1.05
اقبال لاهوری	8137	1.0
احمد شاملو	6531	0.8
فروغی بسطامی	5863	0.72
عراقی	5788	0.71
پروین اعتصامی	5592	0.69
حافظ	4805	0.59
هلالی جغتایی	4301	0.53
عرفی	3899	0.48
عنصری	3776	0.46

ظهیر فاریابی	3675	0.45
عبید زاکانی	2925	0.36
فروغ فرخزاد	2839	0.35
منوچهری	2814	0.35
ازرقی هروی	2672	0.33
مهدی اخوان ثالث	2616	0.32
بهرام سالکی	2557	0.31
شهریار	2422	0.3
سهراب سپهری	1964	0.24
شیخ محمود شبستری	1684	0.21
ابن‌حسام خوسفی	1614	0.2
ابوسعید ابوالخیر	1589	0.2
هاتف اصفهانی	1587	0.19
رضی‌الدین آرتیمانی	1550	0.19
نیما یوشیج	1444	0.18
شیخ بهایی	1288	0.16
رهی معیری	1277	0.16
عمان سامانی	1116	0.14
رودکی	1043	0.13
باباطاهر	764	0.09
خیام	642	0.08
عبدالقادر گیلانی	641	0.08
شاطرعباس صبوحی	607	0.07
باباافصل کاشانی	500	0.06
سعدالدین وراوینی	498	0.06
مهستی گنجوی	380	0.05
عبدالواسع جبلی	320	0.04
کسایی	306	0.04
نصرالله منشی	243	0.03
هجویری	238	0.03
فایز	۲۲۴	0.03
خلیل‌الله خلیلی	108	0.01

3.3.2 Extracting Data

In the Ganjoor database, the poems are not directly attributed to the poets and are not in a collection. On the other hand, the poems are stored in verse form. We need a collection in which for each verse, the place of that verse in the poem, the poem and its poet is clear. As a result, we extract the data with this structure. Ganjoor database is a structured SQL database. The algorithm for extracting poems is as follows:

Dataset Preparation Pseudocode

```
Open sqlite3 ganjoor;
Extract poet names and IDs;
for each poet do
        find it’s categories from database;
end for
for each category do
        Find poem IDs;
end for
for each poem ID do
        Extract poem of each poem ID;
end for
Create sqlite database and add poems to it;
```

Our dataset is as follows (the first 4 rows of the set are given in the table below):

poetID	poet	poemID	vorder	position	text
2	حافظ	10095	1	0	الاای آهوی وحشی کجایی
2	حافظ	10059	2	1	مرا با توست چندین آشنایی
2	حافظ	10059	3	0	دو تنها و دو سرگردان دو بی‌کس
2	حافظ	10059	4	1	دد و دامت کمین از پیش و از پس

In this way, for each verse, we know which poem and poet that verse belongs to and what position it has in the poem. Now we need to clear the text of the poems from anything except for poetry. Some poems have additional explanations and signs, as well as some additional texts that were added to the poem for more information or created for better display. We only need poetry, so we analyze the text of each poem so that only poetry remains.
By tokenizing all the poems in the collection, a bag of words is created from which we extract words. Poems become 10631910 tokens, which include 266519 distinct and exclusive words.

3.3.3 Rhythms

In general, there are 151 poetic rhythms in these 63,791 poems by 72 Persian poets (Ganjoor, Statistics, 2018). Prosodic rhythm or poetic rhythm is the measure by which the rhythm of the poem is measured. To find the rhythm of a poem, we divide and categorize verses into phonetic units and order them. These are the sounds that are important to us and the written form of the words is not important. In fact, we write everything we hear and say. In Persian language, like most languages in the world, writing and reading are not the same. Therefore, to understand the rhythm of Persian poetry, we must look for the phonetic units of the language. We call monosyllables "short syllables" and two-syllables "long syllables". We show the short syllable with “U” and the long syllable with “-“. Of course, we also have another syllable called "stretched syllable". In practice, what we feel from a long syllable is equivalent to a long syllable and a short syllable, and for this reason, we display it with “U-“. Actually, there is a slight difference, but due to similarity and order, it is defined in this way. Each syllable consists of a consonant and a vowel. Consonant is a letter without movement like b, p, t, etc. A vowel is also a movement that we give to a consonant, which makes sounds, and it is divided into two types, short and long. The short vowels are the same as fathah, kasrah and dhammah (ـُـ , ـِـ , ـَـ). Long vowels are also pronounced as "AA", "E" and “OO" (“او“, “ای“, “آ”). For example, the components of the word “رَسا” (“Rasa”) are as follows:

Rhythm = consonant “ر” (“R”) + short vowel “ َ ” + consonant “س” (“S”) + long vowel “ آ ” (“A”) .

In Persian poetry, syllables are defined as follows:

- **short syllable (U):**
  consonant + short vowel (like “رُ”, ”نَ”)
- **long syllable (-):**
  consonant + short vowel + consonant (like “دَر”)
  consonant + long vowel (like “یا”)
- **stretched syllable (U-):**
  consonant + short vowel + consonant + consonant (like “خُفت”(“khoaft”))
  consonant + long vowel + consonant (like “یار”(“yaar”))
  consonant + long Vowel + consonant + consonant (like “خواست”(“khaast”))

There are some points in writing the text that should be paid attention to. These items are mentioned below:

- Words with tashdid become two letters. For example, "فَرّ" becomes "فَرر".
- The letter "خ" that comes after the two letters “و” and “ا“ ,”و” is removed. Like "خوان" which becomes "خان".
- If the letter "ی" is followed by a consonant which is followed by "ا" or "و", a short vowel comes next.
- The connector letter "ه" becomes a short vowel “ ِ”.
- If the final letter of a word is "ن", it is not considered.
- If the letter "ا" comes between two consonants, which is read as stretched, a short vowel comes at the end of the word.

So far we have understood that the most important factor in finding the rhythm of a poem is its writing. Since nowadays, usually, Persian text is written without vowels, and almost all the poems in our dataset are like this, we need to do the editing ourselves. According to the researches that has been done, there is no suitable tool for this. Only in Arabic there is a tool (Abdelkrime Aries, 2022) which does not work well for Persian. The main problem is in the type of writing poetry. In the Arabic language, there are prepositions in the text that are not used in the Persian language, and the grammar rules of the Arabic language are different from the Persian language. As a result, this tool cannot work well for Persian poetry. At first, we tried to improve the output of this tool so that we could use it to edit Persian poetry. This tool removes letters from the poem and adds some other letters which is irrelevant. Also, it adds signs to the poem that are not used in Persian poetry and language.

For example, consider the verse “مرا می‌بینی و هردم زیادت می‌کنی دردم” (“You see me and every time you increase my pain”).

The output of this tool for this verse is “مَرَا مَبِن وَ هَر دَم زَادت مَن دَرد”.

In the following, the works that were done to modify this tool are stated. First, we must restore everything that was removed from the text. We compare the original text with this output and add the letters that are not in the output. Then we delete the characters that are wrong or not used in Persian language. We change some characters like “ٓ” to “َ“ which is used in Persian language, or we remove characters like “ْ“ which are not used in Persian language. It should be noted that not being used does not always mean non-existence, rather they do not have an effect in determining the poetic rhythm. Now we should edit the rest of the text based on the rules and patterns that exist in Persian language. It is not possible to target all the editing needed with a pattern, because, for some cases, such as connector vowel, it is not possible to find a pattern. Some patterns that can be identified and targeted are described below.

- There can always be only one space between the words of the poem.
- If the letter “ی” comes alone, it becomes “یِ”.
- If the letter “ی” is followed by the letter “و”, it becomes “یِ”.
- If the letter is consonant and the letter “ا” comes after it, if the two letters after it are the same and the third letter is not “ی”, then the first letter gets short vowel.
- If two consonants come alone, after first letter comes a short vowel.
- If the letter “ی” comes between two consonant letters and the second consonant letter is “ا” or “و”, then after the first letter comes a short vowel.
- If the letter “ا” comes and then a consonant, and there is no vowel after that, then a vowel comes after it.
- If the two letters before and the two letters after letter are in the form of “consonant + ی” and the middle letter is a consonant, then a short vowel comes after the middle letter.
- If the letter “ا” comes before a consonant and another consonant comes after it, then a short vowel comes after the letter.
- If we have three consonants and the middle consonant is “ی” or “ا”, then a short vowel comes after the first letter.
- After the letter “خ” which is followed by two letters “و” and “ا”, there is no short vowel.
- If there are two consonants other than “ا” and the only one letter “ی” comes after it, a short vowel comes after the first consonant.
- If there is two same consonants, a short vowel comes between them.
- If a consonant is followed by the letter “ی” and then a short vowel, if the next letter is “و” or “ا”. Then after the first letter comes a short vowel.
- Two short vowels do not come together.

- After the vowel “ا” does not come a short vowel.
- A short vowel does not come alone.

These patterns (and other patterns that were used in the project, but were not addressed due to their partiality), were found experimentally and may not always be correct.

Finally, by applying these patterns to the output of the introduced tool, the writing is improved to a suitable extent. As mentioned, some cases do not have a specific pattern and cannot be found and targeted in a Rule-based manner. Prosodic rhythms can now be extracted from the improved output.

The improved output for the previous example is equal to “مَرا میبینِی و هَر دَم زِیادِت میکُنی دردم”.

The writing of the poem has a direct effect on the determination of the rhythm, in such a way that if the writing of the poem is exactly correct, the rhythm is also calculated exactly correct, and the more accurate the writing is, the more accurate the rhythm will be calculated. Since some points such as connector vowel (a vowel that connects two independent words, such as “ـِ” in “دردِ مشترک”) are not included in the editing, the writing of the poems is sometimes completely correct and sometimes almost correct.
As a result, the calculated rhythms are sometimes completely correct and sometimes almost correct.

For example, the rhythm of the poem “مرا می‌بینی و هردم زیادت می‌کنی دردم” with the presented algorithm is equal to “مفاعیلن مفاعیلن مفاعیلن مفاعیلن”, which is completely correct, and the rhythm of the poem “من با تو حدیث بی‌زبان گویم” is equal to “مفعول مفاعلن مفاعیلن فعلن”, which It is almost correct because it should be “مفعول مفاعلن مفاعیلن”.
To solve this problem, we make a comparison between the valid rhythms and the calculated rhythm to choose the closest rhythm as the correct rhythm. For example, after doing this comparison, the rhythm of the previous example becomes “مفعول مفاعلن مفاعیلن”, which is the correct rhythm.
Naturally, this solution does not solve the problem completely, but it improves the accuracy of rhythm calculation to some extent. To extract the rhythm, we syllabize the written poem according to the rules introduced earlier and specify its afaaeils (U, -, U-). In general, about 20 prosodic elements are widely used, which can be seen in the table below.

Monosyllable	Two Syllables	Three Syllables	Four Syllables	Five Syllables
- : فَعْ	- U : فَعَل	U U - : فَعِلُن	- U - - : فاعِلاتُن	- - U - - : مُستَفعَلاتُن
	- - : فَعْ لَن	- U - : فاعِلُن	- U - U : فاعِلاتُ	U - U U - : مُتَفاعِلُن
		U - - : فَعولُن	U U - - : فَعَلاتُن	
		- - - : مَفعولُن	U - U U : فَعَلاتُ	
		- - U : مَفعولُ	U - - - : مَفاعیلُن	
			U - - U : مَفاعیلُ	
			U - U - : مُفاعِلُن	
			- - U - : مُستَفعَلُن	
			- - U U : مُستَفعَلُ	
			- U U - : مُفتَعِلُن	

Table 1. afaaeils

After calculating the rhythms, we calculate the corresponding elements according to table(1).

Rhythm Finding Algorithm

```
rhythm {
        input : verse
        output : rhythm
        input verse to aruudy and take the output;
        rectify aruudy output and update verse;
        build rhythm based on patterns;
        return rhythm
        }

for verse in poem do
        add rhythm of verse to rhythms;
end for
for each rhythm in rhythms do
        find valid rhythm;
end for
take the most frequent and the most probable rhythm as poem's rhythm;
```

3.3.4 Formats

In general, there are about 13 styles in Persian poetry. Generally, the difference between formats is in their rhyming pattern. Although some formats have the same pattern, these formats differ in the content, meaning, emotions, and number of verses. In the following, we will discuss the formats.

- **Ghazal**: The first verse and even verses are rhyme and the number of verses is usually between 7 and 14.
- **Qaseedeh**: It is like Ghazal, with the difference that the number of verses is usually between 14 and 70. In addition, they are usually different in meaning.
- **Masnavi**: Each couplet has its own rhyme. In each couplet, two verses are rhyme.
- **Gheteh:** It has at least two couplets and its even verses are also rhymed.
- **Tarjeeband:** several sonnets that are connected by a couplet with a different rhyme than the sonnets, and the connecting couplets always have a fixed rhyme.
- **Tarkeebband:** It is like Tarjeeband, with the difference that the connecting couplet's rhyme changes every time.
- **Robaie:** It is four verses, all four of them rhyme together or all of them rhyme except for the third verse. A robaie begins with a long syllable.
- **Dobeitie:** It is like Robaie with the difference that it starts with a short syllable.
- **Chaharpareh:** It is a set of connected Dobeities. In a Chaharpareh, sometimes only even verses are rhyme.
- **Mosammat:** consists of several parts, each part has at least 3 verses and its number is odd, and its verses are rhyme, a verse comes with a different rhyme, and then the next section begins, which has a different rhyme from other parts and verses.
- **Mostazad:** It is a type of ghazal in which a short piece of rhyme poetry is added to the end of each verse, and these pieces of poetry also rhyme with each other. There is another mode where at the end of each verse there is a short verse with rhyme, which are rhyme together, or after each couplet, there is a verse which has rhyme.
- **Mofrad:** It is a one-couplet lyric in which the two verses may not rhyme.
- **Noe:** It does not have a special pattern. It can have a rhyme or not, the rhythm and rhyme in this format are variable.

It should be noted that there are other formats that are either a special type of defined formats or have a low frequency, and they can be considered a subset of defined formats. The Noe format has several subsets, since they are structurally similar, there is no need to separate them.
To identify these formats, we must first find the rhymes (the words that usually come at the end of each verse and have rhyme with the end of the other verse) so that we can get the pattern of the poem. The part of the verse that is usually at the end of the verse and is of the same rhythm and harmony with the next verse is called rhyme. If the rhyme has exactly the same letters, vowels and meaning, it is Radif and must be skipped to reach the rhyme.
After finding the rhyme, based on the obtained pattern and the pattern that were expressed, We select the format corresponding to it.

Format Finding Algorithm

```
for each verse of the poem do
        find the rhyme of the verse;
end for
compare it's pattern with defined patterns and select the right format;
return format
```

3.4 Train Set

Train set is a knowledge-base that has information about the poet's poetic style. This information is the weighted frequency of each of the features that we have discussed so far and learned how to calculate them. 266519 unique words, 151 rhythms and 14 formats, including the added case for undefined format and rhythm, we have a total of 266684 features.
Below are some excerpts from the set:

Set(1):

Words:

poet	poetID	دم	…	گویم
حافظ	2	1.508171	…	0.855642
خیام	3	0.101390	…	0.000000
فردوسی	4	2.864258	…	2.818586

Rhythms:

poet	poetID	فعولن فعولن فعولن فعل	…	مفعول مفاعلن فعولن
حافظ	2	0.998172	…	0.999306
خیام	3	0.998172	…	0.999306
فردوسی	4	1.002963	…	0.999306

Formats:

poet	poetID	غزل	…	مثنوی
حافظ	2	0.887210	…	0.003999
خیام	3	0	…	0.007924
فردوسی	4	0	…	0.717264

The point that should be noted is that, as stated earlier, the rhythms and formats are calculated according to the stated patterns. As a result, sometimes the rhythm or format that is calculated may not be correct. But since it was calculated according to the pattern, it is not wrong, but the pattern matched the selected feature. Therefore, if we want to check the calculated rhythms and formats, we can see that the accuracy of their calculation is something around 60-80%. The low accuracy of the features affects the accuracy of the model, but according to the evaluations, the output is favorable. One reason is the frequency of the rhythm. Because if a feature is wrongly calculated, because the number of mistakes is small compared to the number of correct ones, its weighted frequency will be less than the correct feature. For example, in the Train Set(1) we see in the format section, since Khayyam and Ferdowsi do not have poems in the form of ghazal, their weighted frequency is the lowest and the same. While Hafez has a much larger Rate. And also in the form of Masnavi, which is the only format in which Ferdowsi wrote poetry, the rhythm frequency is higher, and the other two poets who are not Masnavi composers, have a lower and almost the same Rate. Of course, accuracy can be increased by prioritizing formats. But because the pattern is more important in the poetic style. This can be ignored. For example, the fact that most verses in Khayyam's poems rhyme is more important than whether this pattern is a Robaie or a Ghazal. And for this reason, because the style of poets is different from each other and is discriminative for each poet, in the end, it is the ratio of the values of these features that are important. And because our model is pattern-oriented, the low accuracy of exact feature detection does not affect it dramatically.

The values in the cells are the calculated weighted frequency that determines the degree of specificity and probability of that features for that poet. In this Set, if a poem does not have any features, zero value is considered for it. For the answer, it is necessary to first clarify how the classification method works. When a poem is given to the Model, first the words of that poem are extracted. Then the rhythm and format of the poem is calculated. Finally, a probability is calculated for each poet, which is equal to the product of the weighted frequency of the features in the poem for that poet. After calculating the probability for all poets, the most probable poet is selected as the poet. If a word is used in the poem that a poet has not used so far, then it is not possible that this poem belongs to this poet.

Next, we compare another set with the main set.

Set(2):

Words:

poet	poetID	دم	…	گویم
حافظ	2	1.508171	…	0.855642
خیام	3	0.101390	…	1
فردوسی	4	2.864258	…	2.818586

Rhythms:

poet	poetID	فعولن فعولن فعولن فعل	…	مفعول مفاعلن فعولن
حافظ	2	0.998172	…	0.999306
خیام	3	0.998172	…	0.999306
فردوسی	4	1.002963	…	0.999306

Formats:

poet	poetID	غزل	…	مثنوی
حافظ	2	0.887210	…	0.003999
خیام	3	1	…	0.007924
فردوسی	4	1	…	0.717264

If we consider the set as Set(2), which has put a value of one instead of zero values, then if there is a word in the poem that a poet has not used so far, the probability that the poem belongs to this poet will be less, but It will not be zero; Because only the effect of that word has been ignored in the calculation and the calculation will be based on other features.

This matter is important because this poem may be a new poem by that poet and the poet has used this word for the first time. But since our model is a Eager algorithm and classification is done based on previous knowledge, if a word has not been used by a poet, according to the Model's knowledge, this poem cannot belong to this poet. Because the Model knows this poet does not use such a word.

According to these descriptions, the first Set is considered as the Train Set. Another point is about the balance of the data, the features have values for all poets, and since these values are weighted, they are valued according to their specificity and probability. So this makes this Set a balanced set.

3.5 Classification

In this project, the Classification is done by calculating the probability of the poem belonging to the poet, in such a way that the probability of belonging to each poet is calculated for each poem and the most likely poet is selected as the poet. The probability calculation equation is as follows:

$$probability\,(poem, poet) = \prod_{feature \in poem's\ extracted\ features} TFIDF\,(feature, poet)$$

Therefore, when a poem is received, first its words are extracted and the rhythm and format of the poem are calculated. Now, for each poet, the product of the weighted frequency of each of the features extracted from the poem is calculated for the poet. Finally, the poet who has achieved the highest possibility is chosen as the poet.

4. Evaluation

4.1 Test Set

To evaluate the presented approach, we need a collection of poems whose poets are unknown. Our Test Set contains 142 anonymous poems.

In the table below, a section of one of the Test Sets can be seen.

ID	Poet	poem
2831		u0628\u0647 \u0645\u0631\u06af\"] ...\u06a9\u0648
2825		u0628\u06cc\u0627\u0645\u062f\u\"] ...\ 06cc\u0645
1857		u0634\u0628\"] ...\u0646\u06cc\u0633\u062a \u062f

This Set contains a series of anonymous poems presented as a list of verses and saved in JSON format.
To evaluate the model, this collection is created and tested several times.
Also we have done separated tests for each poet (we had chosen a few based on the number of poems) to evaluate the model's performance on poets with less poems available.

4.2 Random Attribution

To better evaluate the presented approach, the Test Set is labeled once randomly. (A section of the Set):

ID	Poet	poem
2831	سیف فرغانی	u0628\u0647 \u0645\u0631\u06af\"] ...\u06a9\u0648
2825	عرفی	u0628\u06cc\u0627\u0645\u062f\u\"] ...\ 06cc\u0645
1857	مهستی گنجوی	u0634\u0628\"] ...\u0646\u06cc\u0633\u062a \u062f

In this way, for the test samples, we randomly select a poet from among the 72 existing poets. This model is created solely for comparison with the original model.

5. Evaluation Results

To better evaluate the proposed approach, the model has been tested several times with different test sets. In the following, the evaluation result of the presented approach, the random model and the similar project (Amirali Kaboli, 2021) are compared.
Since the number of poets is large, accurate evaluation of the model requires a collection that has a suitable number of poems from all poets. But there is no such collection, because the number of existing poems from poets is different. Some have thousands of poems and some only a few. For better evaluation, the model is tested with collections containing 100 anonymous poetry samples. Initially, these collections contain anonymous poems from all poets, randomly selected. This test has been done several times and the best result is reported. Then, collections of up to 100 samples of anonymous poems were tested for each poet separately. This test is done to better evaluate the performance of the model. In this test, the accuracy of the model has been evaluated for poets with the least number of poems and more. The table below shows the calculated accuracy for several poets.

Accuracy	عطار	فردوسی	ملک‌الشعرا ء بهار	حافظ	سهراب سپهری	نیما یوشیج	رودکی	خیام	خلیل‌الله خلیلی
Poet	99	100	90	83	100	100	85	57	50

And also the overall evaluations:

	Accuracy
Random Classification	3
Presented Model	**89**
Similar Model	81

A similar project that was introduced earlier was able to reach an accuracy of 81% in the best case. According to the results, we can see that only by adding features such as the rhythm and format of the poem to the words used in it, the accuracy of Poet Recognition increases by 8%. The random model shows that with the high number of poets in the collection, it is much more difficult to recognize the poet than when there are only a limited number of poets (such as other projects that had a collection of 3 poets) in the Set. For example, for a Set that has 3 poets, the probability of the correct label for each test sample is 33.33%, while for our Set with 72 poets, this probability is 0.014%. For poets such as Sohrab Sepehri, who has 6 poems, the Model has achieved 100% accuracy, and Attar, who has 246 poems, has achieved 99% accuracy, which indicates the appropriate performance of the model. According to the tests, the Model has reached 89% accuracy in the best case.

6. Conclusion and Future Works

Persian poet recognition refers to identifying or guessing the author of a poem based on the characteristics of the poetic style and information about the poet and how he writes his poems. Past approaches have only examined the language model, i.e. the evaluation of the words of the poems, and have calculated the probability based on this language model. In this article, an approach was presented that, in addition to the vocabulary, also addressed the key features of Persian poetry, i.e., the Rhythm of the poem and the Format of the poem, and calculated their weighted frequency in order to evaluate these features based on the degree of specificity and probability. Finally, the Model chooses the most probable poet based on the probability calculation based on a very simple definition which is only the product of the weighted frequency of the features in the poem.

This project beautifully shows the importance of simplicity while being effective and optimal. It shows that the right and clear method, even though it is simple, can lead to a great results.

The performance evaluation of the Model should be checked with a Set including all poets and the presence of a suitable number of samples from each, so that a more accurate evaluation can be made.

At first, a broader approach was considered, but due to the lack of necessary tools and conditions, another approach was followed. For future works, the necessary tools and conditions should be prepared, and as a result, the approach will be feasible. Some of these tools are tools for detecting the emotions and theme of the poem, which examines the real sense of the poem and its meaning which is very valuable in Persian poetry. Poetry writing tool, to write Persian poetry correctly, so that the rhythm of the poems can be calculated accurately. A tool for recognizing literary features such as simile, allusion, irony, etc., which has a decisive role in identifying the literary style of poets and as a result, increasing the effectiveness of the identity identification Model. And finally, creating more advanced and optimal Models and algorithms that help the process of identifying the author's identity using Machine Learning and Artificial Intelligence.

7. References

Reza Ramezani. (2021). A language-independent authorship attribution approach for author identification of text documents. Published: Elsevier, Expert Systems with Applications, Volume 180, 15 October 2021, 115139.

Hobson Lane, Cole Howard, Hannes Max Hapke. (2019). Natural Language Processing in action. Manning publications Co. .

Akshay Kulkarni, Adarsha Shivananda. (2019). Natural Language Processing recipes. Apress Berkeley, CA.

Bojan Babic, Nenand Nesic, Zoran Miljkovic. (2008). A review of automated feature recognition with rule-based pattern recognition. Published: Elsevier, Computers in Industry, Volume 59, Issue 4, April 2008, Pages 321-337.

Soumya George K, Shibily Joseph. (2014). Text Classification by Augmenting Bag of Words (BOW) Representation with Co-occurrence Feature. Published: IOSR Journal of Computer Engineering 16(1):34-38.

Yaakov HaCohen-Kerner, Daniel Miller, Yair Yigal. (2020). The influence of preprocessing on text classification using a bag-of-words representation. Published: PLoS One. 2020; 15(5): e0232525.

Abdelkrime Aries. aruudy. (2022). GitHub Repository.

Amirali Kaboli. persian poet detection. (2021). GitHub Repository.

Ganjoor. database, v2.81. (2018). GitHub Repository.

Ganjoor, statistics, (2023). Ganjoor

Soroush Mehraban. Poet Detection Using NLP. (2021). GitHub Repository.

Arash Hajisafi. NLP Poet Identification. (2021). GitHub Repository.

MohammadJavad Ardestani. NLP Persian Peot Identification. (2022). GitHub Repository.

امید طبیب‌زاده، لیلا ضیامجیدی. بررسی وزن اشعار عامیانه ی فارسی بر اساس نظریه ی وزنی. (۱۳۹۰). انتشارات: ادب پژوهی، شماره ۱۶، صفحات ۵۹-۷۹.

امید طبیب‌زاده. ساخت وزنی در شعر عروضی فارسی. (۱۳۸۹). انتشارات: زبان و زبان شناسی، شماره ۱۱، صفحات ۱-۲۰.

احمد رضایی، محمود بشیری. تحلیل و بررسی مفاهیم سبک در شعر شاعران پارسی گوی تا اوایل قرن نهم. (۱۳۹۱). انتشارات: فنون ادبی، شماره ۷، صفحات ۳۷-۵۲.

مهدی کمالی، علی‌اصغر قهرمانی مقبل. جایگاه هجای کشیده در وزن شعر فارسی. بررسی تاریخی و تحلیلی. (۱۳۹۹). انتشارات: ادبیات پارسی معاصر، شماره ۲، صفحات ۲۹۵-۳۲۸.

ملک الشعراء بهار. درباره شعر و ادب فارسی. (۱۳۵۱). انتشارات: نگین مهر، شماره ۸۹، صفحات ۵-۶ و ۶۲-۶۳.

احمد سمیعی. درباره سبک و سبک شناسی شعر فارسی. (۱۳۸۴). انتشارات: نامه فرهنگستان، شماره ۲۸، صفحات ۱۴۹-۱۵۳.

یحیی طالبیان، مهدیه اسلامیت. ارزش چند جانبه ردیف در شعر حافظ. (۱۳۸۴). انتشارات: پژوهشهای ادبی، شماره ۸، صفحات ۷-۲۸.

شهربانو رنجبر، محمدعلی داوودآبادی، محمدرضا زمان‌احمدی. بررسی زیبایی شناسی زبان و آرایه های ادبی در شعر سعدی. (۱۴۰۰). انتشارات: زیبایی‌شناسی ادبی، شماره ۴۸، صفحات ۲۱۵-۲۵۸.