
Saeedeh Momtazi*

Computer Engineering and Information Technology Department
Amirkabir University of Technology, Tehran, Iran
momtazi@aut.ac.ir
Tel: +98(21)64542729

Omid Moradiannasab
Saarland University, Saarbrucken, Germany

Abstract

In this paper, we propose a novel approach for knowledge discovery from textual data. The generated knowledge base can be used as one of the main components in the cognitive process of question answering systems. The proposed model automatically extract relations between named entities in Persian. Our proposed model is a bootstrapping approach based on n-gram* model to find the representative textual patterns of relations as n-grams in order to extract new knowledge about given named entities. The main motivation for this work is the characteristic of the sentence structure in Persian which, in contrary to English sentences, is in subject-object-verb format. The proposed approach is a purely statistical one and no background knowledge of the target language is required. This makes our method applicable to any open domain relation extraction task. However, as for our test-bed, we focus on the domain of biographical data of international poets and scientists to build a knowledge base about them. Qualitative evaluations based on human assessment is an evidence for the efficacy of our method.

Keywords: Computational linguistics, information extraction, statistical language modeling, n-gram Model, relation extraction, textual pattern acquisition

*Corresponding Author

*a contiguous sequence of n items from a given sequence of text or speech
1 Introduction

Question answering is one of the main fields in computational linguistics which has been addressed using cognitive computing [1]. Cognitive computing can be used as an advanced approach that models how human beings does for any question through a computerized system [2, 3].

As described by Kaur and Singh [4] “cognitive systems are complex information processing ones, capable of acquiring information, putting it into action and transmitting knowledge”. This is the exact process that is performed inside a question answering system through the following steps [4]:

1. “Understanding natural language and human interactions
2. Generating and evaluating evidence based hypothesis
3. Adapting and learning from user selections and responses”

This process requires a large amount of knowledge to establish a high coverage and efficient question answering system [5, 6]. In such systems multiple knowledge formats from a large variety of natural language sources such as textbooks, encyclopedias, newspapers and literary works have to be processed to provide a knowledge base [7]. The process of discovering knowledge from textual resources is normally done using information extraction techniques which are one of the important computational linguistics tasks.

Information extraction focuses on finding relations between named entities. The extracted relations are then used to build knowledge bases which are the main resource in cognitive process of question answering systems.

As mentioned, cognitive computing refers to reading, reasoning, learning, and making inferences from vast sets of content [8]. The generated knowledge bases are the best example of such contents.

Bootstrapping approaches such as the one presented in this article hold great potential for knowledge base development. Success of knowledge base development depends on the existence of fast ways to mine the large amount of unstructured texts on the web to extract information that can be added to knowledge bases. Taking advantage of such knowledge bases, question answering systems are enabled to answer complex questions which require information from different sources. Previous research has shown that automatic relation extraction is feasible for question-answering tasks with decent results [9].

Having a huge amount of unstructured text on the web, provides us the opportunity to extract relations between named entities. Among various approaches that have been proposed for this task, semi-supervised machine learning techniques which use a bootstrapping algorithm received researchers’ attention. In these approaches, in order to extract the relation between named entities; e.g., a person and a location, first, the corresponding named entities in the text are tagged; i.e., all person names and location names in this example. Then, by focusing on the context and mainly on the terms that occur between the two entities, representative patterns are extracted which are later
used to distinguish the corresponding relation between the two entities. As an example, some of the possible relations between a person and a location are: \textit{place-of-birth}, \textit{living-place}, \textit{place-of-death}, and \textit{tomb}. The procedure explained above has achieved promising results for languages such as English with \textit{subject-verb-object} word order, because in most cases the two entities appear as subject and object (either direct or indirect) in a sentence, and the verb is served as the critical element to describe the relation between the entities and also to match the corresponding predicates. For example, “Albert Einstein was born in Ulm”, “James Hall was buried in New York” or “Daniel Gabriel Fahrenheit died in The Netherlands”. Such a model which mainly focuses on the text between two entities, however, cannot generate reasonable results for all languages, specially those which do not follow the \textit{subject-verb-object} word order. Applying this method to Persian (as one of the Indo-European languages) suffer from this problem. The word order in sentences of this language are in \textit{subject-object-verb} format. In this language, the verb always appears toward the end of the sentence and both subject and object appear next to each other toward the beginning of the sentence. As a result, many straightforward relations, whose representative patterns normally appear between subject and object in English, are not easily distinguishable in Persian.

Such a difference in the structure of Persian sentences, motivated us to study the relation extraction task for this language in more depth. To this aim, we propose a language model-based method to find the representative textual patterns of every relation as \textit{n}-grams to be used for extracting new information. Statistical language models have been widely used for various natural language processing and text retrieval tasks, including opinion mining [10], ad-hoc retrieval [11], sentience retrieval [12], and word prediction [13]. In this paper, we benefit from this approach for information extraction from unstructured texts. \textit{N}-gram model is an entirely statistical model that measures statistical properties of text strings in a corpus disregard to the vocabulary, lexical, or semantic properties of the document language. An approach based on \textit{n}-gram model without employment of any other natural language process tools with promising outcome is worth being further studied. In the current study, we focus on the domain of biographical information of international poets and scientists. For this purpose, we constructed an ontology containing biographical scheme to be expanded with information newly mined by our proposed relation extraction method. Although the use of biographical information of well-known public figures provides clarity and structure to the task, the proposed approach itself is not domain-specific and has the potential to be applied to other open-domain relation extraction tasks. Qualitative evaluations are provided as evidence for the efficacy of our approach. We show that our method yields reasonable results and major improvements over the baseline.

The remainder of this article is structured as follows: Section 2 briefly reviews the state-of-the-art approaches used for relation extraction. Section 3 describes the baseline approach against which we compared our results. Our proposed method is presented in Section 4. In Section 5, we describe the steps taken to create a biographical corpus in Persian. We report and discuss the
results obtained in Section 6. Finally, the article is summarized in Section 7.

2 Related Works

Information extraction is a challenging task from different perspectives. The input of this task is either semi-structured data or raw text. Semi-structured data includes mark-ups like HTML, Wikipedia info-boxes, Web tables or lists all of which can effortlessly be processed to extract required information[14, 15, 16, 17]. Information extraction from raw text, however, needs more complex analysis. The available approaches for extracting information from raw text are divided into three main categories: (1) manually built patterns, (2) supervised approaches, and (3) semi-supervised approaches.

Hindle [18] established employment of manually built patterns for deriving semantic relatedness. He provided hand-crafted textual patterns in order to extract specific relations from unstructured text. Later on, Hearst [19] proposed employment of lexico-syntactic patterns instead of textual ones in order to increase the precision. Several other computational linguistics methods are also used to improve the quality of the patterns [20]. The most common computational linguistics method for this task is part-of-speech tagging. To have a full awareness of grammatical structure of a sentence, a dependency parser is also potentially helpful. Especially, a Parser with the ability to detect far dependencies is advantageous for the subject-object-verb word order problem. Although the precision of such methods are high, they show low recall. To improve the recall for these systems, pattern repositories should be enriched, which is a labor intensive task.

Another approach for relation extraction is supervised learning. Jurafsky and Martin [21, 22] employ machine learning methods for this purpose. Some of the methods proposed for this approach are based on Hidden Markov Models and Conditional Random Fields [23, 24]. Another group of models for this task performs likelihood optimization processes [25, 26, 27, 28]. The weakness of these methods is their narrow awareness of context. As a result, finding high-quality patterns in long sentences with scattered keywords becomes complicated. It gets even worse when the order of sentence elements is subject-object-verb. Bayesian models, logistic regression, and support vector machines are the other methods examined to distinguish whether an input sentence is similar to a set of training sentences or not. By providing manually labeled sentences, a probabilistic model could be trained. However, the need for substantial amount of annotated data for training is the drawback of these approaches.

Semi-supervised approaches which iteratively search for better patterns were first proposed by Brin [29]. This process starts with a small set of records, called seed data, as a set of correct facts. The process then iteratively improves these records by adding new patterns to this initial set. This is done by searching through the corpus to construct patterns of texts or linguistic blocks. Extracted patterns are then utilized to find new relations. The newly extracted relations, which are potentially also correct facts, will be used again to enrich the patterns.
This process continues until a stopping criterion is satisfied. The outcome is used to enrich the knowledge base. So far, a variety of tools working in this scheme have been proposed, including Snowball [30], Semagix/SWETO [31], KnowItAll [32], Text2Onto [33], LEILA [34], TextRunner [35], and SEAL [36]. These systems also take advantage of natural language processing tools to improve the results by employing parts-of-speech tagging, lexical dependency parsing or using heuristics for entity disambiguation, etc.

In addition to the above works, there are a limited number of research on relation extraction for Persian. These works, however, have only focused on generic relations (e.g. is-a and part-of). Moradi et al. [37] defines 17 relations in a “concept-net” of generic relations which makes their work similar to (author?) [38] in the sense of relation types. Our baseline is designed to work in a similar manner to [38] as it also implements an iterative extraction of relation triples. However, the distinguishing factor between our work and theirs is that all our experiments focus on non-generic relations in contrast to [38] and [37]. In [37], the authors design a solution which makes use of a combination of Hearst method, machine translation and Wikipedia infoboxes. Basically, minimally supervised algorithms like that of Hearst do not show high performance for generic patterns since system precision greatly decreases from the introduced noise and bootstrapping deviates from the correct patterns after very first iterations. In order to control this deviation, [37] employs Google machine translation and Wikipedia Infoboxes as a supporting further knowledge to the knowledge extractor.

There are not many works on knowledge base population for Persian. One of the few is the work by Shamsfard and Barforoush [39] called Hasti project which also uses small seed sets to extract relations from sentences. The extraction process as explained in [39] is a combination of logical, template driven and semantic analysis methods. Beside the patterns of text for sentences containing certain relations, they also extract the implicit knowledge in a given sentence by logical reasoning at an inference engine which covers basic reasonings such as inheritance laws, transitive rules, etc. [40] is also a simple method to enrich the Persian WordNet by combining “direct semantic context” of some initial concepts. This work also mainly covers generic relations such as hypernymy, part-of and definition. Most of the other works on knowledge acquisition in Persian, such as [41] and [42], are focused on extracting terminologies and lexicons rather than ontological relations.

3 Baseline

To compare our model with the state-of-the-art methods, we focus on the work proposed by Ravichandran and Hovy [43], which is based on extracting patterns later used for question-answer pairs. Their approach involves a fixed set of seeds containing correct question term - answer term pairs; e.g., “Mozart” and “1756” as a seed for “birth year” questions. In order to obtain patterns for a question type, terms are extracted for each seed document containing both
the question term and the answer term. Occurrences of the question terms in these documents are replaced with \texttt{<question>} and occurrences of the answer terms with \texttt{<answer>}. Suffix trees are employed to extract the longest matching substrings. These substrings, along with their frequencies, are the raw patterns. In a later step, the authors use a precision scoring algorithm to rank these raw patterns. Their scoring mechanism is based on calculating the ratio of correct answers to all answers retrieved for a given pattern. We present an adaptation of their pattern extraction algorithm to extract patterns for subject-object pairs in a given relation; e.g., \texttt{born_in_year(Mozart, 1756)}.

The second important source for implementing our baseline is the work of Pantel and Pennacchiotti [38], which present the Espresso algorithm for iterative extraction of relation triples from the web by leveraging generic patterns. Their approach is also based on an initial seed set of known relations. These seeds are fed to a pattern extraction algorithm similar to [43]. Unlike that work, however, the authors of [38] extract relations over several iterations, adding the top \( k \) most reliable patterns to extract new seeds for subsequent iterations. The contributions of their work further include metrics for pattern and instance reliability based on association strength and mutual information. The ranking algorithm they apply to patterns is fairly complex but superior to the one described in [43]. Our baseline is designed to also work in a similar manner. We add pattern and instance reliability metrics such as the ones used by [38] and continue the processing flow over several iterations, always adding the most highly ranked relations extracted in each iteration as seeds in the subsequent iteration. Figure 1 shows the process flow within such a bootstrapping approach.

The main overlap between the work of [38] and our baseline is the process of instance extraction. However, in order to assess the precision and productivity of a given pattern, we rank the patterns based on the frequency of their occurrence in the corpus considering the corresponding correct subject-object pairs in the knowledge base. We also rank the extracted instance relations based on the frequency value of the patterns used to extract them. We use an accumulative approach for ranking instance relations; i.e., if an extracted subject-object relation is verified by more than one pattern its score will be the summation of the frequency values of all those patterns.

Replacing the seeds in the corpus is the first step towards extracting patterns. For each relation of a person, we replace all its possible subjects and objects in the corresponding biography file with \texttt{<subject>} and \texttt{<object>}, respectively. We thus generate a new version of the corpus for each relation, with annotated subjects and objects. We call this subset of the corpus \textit{candidate sentences}.

We follow [43]'s algorithm and use a suffix tree-like structure for finding frequent patterns. In our implementation, we make use of suffix array structure instead of suffix trees. The suffix array is a stripped-down data structure based on the suffix tree, yet pattern matching is very fast. The suffix array is slower in some pattern searches than the suffix tree but uses less space, and is more widely used than the suffix tree. We use this data structure to compute the longest frequent substrings containing both \texttt{<subject>} and \texttt{<object>}. Each of these substrings is considered a \textit{pattern}. The process we implemented fur-
thermore counts the frequency of each pattern and returns this number along
with the pattern itself. The quality of a pattern is assessed via its frequency
of occurrence in candidate sentences, assuming that more frequent patterns are
more productive and more reliable.

Before proceeding to extract relations using the discovered patterns, addition-
al processes are applied to the corpus. One of these pre-processes is Named
Entity Recognition (NER) with the aim of improving relation extraction pro-
cess. This is also suggested by Ravichandran and Hovy [43] to improve precision.
The extraction process is applied to the NER tagged corpus. While extracting
relations, the strings of the <subject> and the <object> positions are verified
to agree with the expected Named Entity types, for example, that the subject
of date-of-birth relation is a Person and its object is a Date.

In analogy to the frequency-based ranking of the patterns [43], we also rank
the extracted instance relations based on the frequency of the patterns which
were used to extract them. We use an accumulative approach for ranking in-
stance relations, that means if a relation is verified by more than one pattern
we calculate its score by summing all the pattern frequencies. In other words,
the ranking criterion is a (cumulative) score of the matching patterns.

4 Proposed Method

Continuous accretion of knowledge represented in unstructured texts over the
World Wide Web affirms the requisite for effective methods to automate in-
formation retrieval tasks. Many approaches have currently been proposed to
introduce retrieval methods with high precision and/or recall. Our method is
mainly based on algorithms integrating n-gram language models to identify seg-
ments of the text with the aim of gathering some predefined semantic relations.

Since a bootstrapping approach is used in our work, a set of sample relations
are required as the initial seed data. This means that given each semantic
relation formulated in our predefined ontology, a number of correct subject-
object pairs pertaining that relation are inserted in the seed data. We took
advantage of Wikipedia info-boxes to aggregate this data. The seed data in
addition to the textual corpus form the input of our system. The system is
designed to use the seeds to extract a ranked set of textual patterns for each
given relation. No special care is taken while gathering the seeds. In other words,
seeds are selected disregard to whether the subject-object pairs are mentioned
anywhere in the corpus or not. Actually, this makes some of the seeds effectless
in the whole process. As a result, even after providing several records to the
system, only a fraction of them might exist in the corpus and the others are
flagged as out-of-corpus. The first fraction, altogether forms an initializer seed
of effective records. Table 1 shows the number of records used as the seed for
each relation in our first experiment.

In order to assess the impact of the number of seeds on the quality of relation
extraction, in a second experiment we also use a seed set of a larger size. Table
2 provides a comparison between the small and the large set.
Having the seed data, each record is represented by a triple with the following structure: \(<\text{subject—predicate—object}＞\). Subject is always the person about whom we are trying to extract relevant information. Predicate is the type of information (attribute) we are interested in; e.g., where the person is buried (tomb). Object is the target named entity holding the relation to the subject.

As mentioned, in contrast to languages like English, whose grammatical structures are in subject-verb-object order, Persian sentences are in subject-object-verb form. As a result, approaches like Hearst’s [19] which tries to construct patterns of words between subjects and objects do not seem promising for Persian. To overcome this problem, our approach is based on the idea to find frequent patterns in the form of \(n\)-grams which possibly occur anywhere in the sentence. A ranked list of such patterns with a weight factor assigned to each are then used to nominate candidate sentences which represent the target semantic relation.

For every relation defined in the ontology, the whole corpus is searched to find all sentences that include both subject and object of the seed records of that relation. These sentences are assumed to most likely include the lexicosyntactic structure representing the corresponding semantic relation in natural language. As an example, for place-of-birth relation, all the records with this predicate in the seed are processed and the whole corpus is searched for a set of sentences containing subjects and objects of each of these record. We call such a set \(U_{\text{place-of-birth}}\) or \(U\) in a more general view. Sentences in \(U\), altogether, are then processed to create a statistical model as described below.

Following are some other notations which are used in this section to describe our model:

- \(p\) is a variable \(n\)-gram with a size ranging from that of a unigram to 4-gram.
- \(w\) is a variable word token.
- \(f(p)\) is the frequency of \(p\) in \(U\).
- \(p_2(w)\) is the most frequent bigram in \(U\) containing \(w\).
- \(p_3(w)\) is the most frequent trigram in \(U\) containing \(w\).
- \(p_4(w)\) is the most frequent 4-gram in \(U\) containing \(w\).

Our method uses an integration of most frequent \(n\)-gram patterns for each relation in its corresponding \(U\) set. At first, all \(n\)-grams ranging from unigrams to 4-grams occurring in \(U\) are generated and counted. \(N\)-grams consisting of nothing but stop-words are removed from the model, but any combinations of stop-words with non-stop-words are maintained for further use. After that, the process goes through the unigram list and expands each word \(w\) in this list to a bigram if and only if

\[
f(p_2(w)) > f(w)/2\tag{1}
\]
Following this rule, for each word \( w \), the most frequent bigrams containing \( w \) are maintained if and only if their frequency is higher than half of the frequency of \( w \). This expansion process continues for every \( n \)-gram to its higher order \( n \)-grams with the following condition which is a general form of formula 1:

\[
f(p_n(w)) > f(w)/2
\]

As an example, consider the word \( w \) with a frequency of 38 is one of the most frequent terms in \( U \). First, the word itself is stored as a useful unigram and then the most frequent bigram containing \( w \) \((p_2(w))\) is found. If \( f(p_2(w)) > 19 \), then the bigram is stored as well and the expansion process continues toward the trigram level by finding the most frequent trigram containing \( w \) \((p_3(w))\) and counting its frequency \((f(p_3(w)))\). If \( f(p_3(w)) > 19 \), we record the trigram and expand the pattern to 4-gram, given the condition is satisfied. In each step, the expansion process terminates if the condition is not satisfied by the condition.

After this expansion process goes over all the words \( w \) in \( U \), the score of each pattern is calculated as follows:

\[
S(p_n(w)) = f(w) + 2 \times f(p_2(w)) + 3 \times f(p_3(w)) + 4 \times f(p_4(w))
\]

\( f(p_n(w)) \) is set to 1 if the word is not expanded at that level. This means that \( n \)-grams of higher order get higher scores than the ones of lower order containing the same word \( w \).

In Persian, compound-complex sentences can be of lengths as high as 50 words; i.e., the highest in our corpus. These sentences often represent several facts corresponding to different semantic relations in an ontology. For example both place-of-death and place-of-birth of a person can be mentioned in a single sentence:

\[\text{ˇsāmũlū dar tehrān motevaled ũd va dar karaj dargozasht.}\]

Shamlu was born in Tehran and died in Karaj.

In this sentence, motevaled ũd (born) is a pattern representing place-of-birth relation and dargozasht (died) corresponds to place-of-death relation. Therefore, the above-mentioned strategy to rely only on frequency of \( n \)-grams to find the best pattern for a target relation fails because it returns both objects; i.e., Tehran and Karaj, for both relations; i.e., place-of-birth and place-of-death, for the target person; i.e., Shamlu which is of course not precise. Therefore, by applying such method to a bootstrapping process, after a number of iterations the results of these two relations will be a mix of both and a deviation from the initializing seeds happens. In this example, after some iterations there will not be any discernment between motevaled ũd and dargozasht in the process.

To overcome this issue, we defined a string metric and took it as another factor into account: the character-wise distance between the object entity and
the n-gram patterns. For each occurrence of a n-gram pattern in any sentence of \( U \), the distance of the n-gram pattern from the object entity is calculated and an average of these distances over all sentences in \( U \) is computed (\( A(p_n(w)) \)). This metric as well as \( S(p_n(w)) \) defined in formula 3 are used to assign a weight to every pattern. The following formula shows how these two factors are taken into account:

\[
R(p_n(w)) = S(p_n(w)) + (50 - A(p_n(w)))
\]  

(4)

50 in this formula is the maximum length of a sentence in our corpus. After calculating \( R(p_n(w)) \) for each n-gram pattern, they are ranked in descending order of their weight and the top five patterns are selected as the representatives of the target semantic relation. The following examples represent the patterns and their weight values identified and ranked in the first iteration for place-of-birth relation.

1) 892.46
be donia āmad
to world came
‘was born’

2) 500.85
dar šahr
in city
‘in city’

3) 362.57
češm be jahān gošud
eye to world opened
‘was born’

4) 242.92
yeki az
one of
‘one of’

5) 214.92
motevaled šod
born became
‘was born’

These five patterns and the values assigned to each are used to collect and rank sentences from the whole corpus. We call this set of sentences candidate segments or \( T \) and anticipate them to represent the target relation between new pairs of entities.

To construct \( T \), we search through all sentences of the corpus and assign a score to each sentence according to the sum of corresponding weights of the
patterns that occur in that sentence. Filtering out the sentences with zero score, we have a set of segments, inside each of which at least one of the patterns is occurred. Those segments which include both a named entity of a person and a corresponding named entity of the object type of the target relation are selected. Following our assumptions, it is very likely these sentences represent a subject-object pair holding the target semantic relation. Segments not including the mentioned named entities are filtered out.

After this process, each of the remaining sentences includes a person name, a target named entity and at least one or more patterns and has a weight value assigned. The person named entity forms the subject and the target named entity forms the object of the relation triple <subject—predicate—object>. The triple also has a weight as a confidence score. In case of re-occurrences of the same triple, the score of the already-existing triple will be additively incremented. This is according to the assumption that if a triple relation is repeated through the text it is a more reliable triple.

An example of an extracted sentence is as follows:

Doctor Eric Bern was born on May 10, 1910 in Montreal.

This sentence is one of the top elements in $T_{\text{place-of-birth}}$ because it contains two of the best patterns for place-of-birth relation: “be doniā āmad” and “šahr”. The confidence score of this sentence is the sum of the weights of these two patterns: $892.46 + 500.85 = 1393.31$. Both the person entity (Eric Bern) and the location entity (Montreal) are tagged in the corpus. Therefore, below relation is extracted from the above sentence:

\[
\text{Eric Bern} \quad \text{place-of-birth} \quad \text{Montreal}
\]

In case of compound-complex sentences it is likely that the sentence includes more than one target entity. In such a situation an adjunct policy is followed. Below is an example of such a sentence with two location entities; i.e., Tehran and Karaj.

Shamlu was born in Tehran and died in Karaj.

The above-mentioned process yields two relations between the person and the two location entities. In this case beside the confidence score, a supplementary measure is specified to make a preference between these two location entities. $D(x)$ is defined as the average of the string distance of every one of the patterns included in the sentence from the object entity. In case of naturally one-to-many relations; e.g., literary-works, there is no need to pick up only one of the target entities. That is why simply all object entities in such cases
can be added to knowledge base holding the corresponding relation to the subject. However, in case of one-to-one relations; e.g., date-of-birth we follow the assumption that the object entity with the lowest average distance to the representative patterns of a semantic relation hold that relation with the subject entity. This means that we choose the object entity with the lowest $D(x)$.

The extracted triples in each iteration are ordered by the confidence score assigned to each. After the end of each iteration several such candidate triples are extracted, however only the top 10 triples are added to the knowledge base. According to a similar assumption to Hearst’s [19], the newly extracted triples are used as additive records to the initial seed. In the subsequent iteration, all the above-mentioned procedure is repeated by the updated seed in order to improve the patterns and extract more triples. In subsequent iterations also new triples are found and ranked and 10 most confident triples are added to the initial seed data. This procedure enriches the seeds and as a result is presumed to improve the patterns and the extraction mechanism in each step.

5 Data

In order to evaluate the proposed relation extraction approach in this article, availability of a named-entity tagged corpus of raw text including biographies is necessary. This section describes the process of preparing such a corpus.

5.1 Corpus

The corpus used in the current study consists of 1,932 Persian text documents including 6412 tokens. This corpus contains biographies of international scientists and poets, including both contemporary and classic ones. It is meant to be a corpus of commonplace biography texts representing various writing styles. The corpus used in this study is arranged by collecting textual documents from several sources available on the web. The data is crawled using a crawler to collect web pages from specific online sources. The fetched documents go through some further processing steps explained in 5.2.

The corpus is formed as a set of numbered text documents. These documents do not necessarily include the name of the target person in the first few lines due to removing the title of the web document in the cleaning process. for some of the target persons, more than one biography with different writing styles exists in the corpus. However, the respective features of that person; e.g., date-of-birth, are guaranteed to be the same in all of such documents (not necessarily in the form of representation though).

5.2 Pre-processing

5.2.1 Cleaning and Normalizing

The corpus documents are cleaned by removing HTML markups; i.e. scripts, styles and also tables. A text normalization process is also applied on the text as
a post-process. The current implementation of the normalization process leaves some noise such as conflated words due to stripped white-space, or conflated sentences resulting from processing tables. This noise affects the performance of the implemented system in our experiments. More tuning of the corpus therefore highly recommended for future evaluations.

5.2.2 Sentence Splitting

In order to extract textual patterns from within sentences, a sentence splitting process is also applied on the corpus. Manual inspection reveals that the sentence splitting performed reliably, except in cases where some noise from the cleaning process interferes, for example when punctuation is accidentally removed or is missing due to processing tables of a web page.

5.2.3 Co-reference Resolving

Due to lack of a co-reference resolver for Persian at the time of implementing our system, a minimal one was developed in the course of this study in order to replace the named entities and pronouns referring to the target person of each biography text.

For this purpose, all occurrences the target person’s name and also all mentions of third-person pronouns in Persian were counted as a reference to the target person of the biography text. Table 3 lists all Persian third-person pronouns. This process is of course subject of many errors which definitely affects the precision of our method in a negative way.

5.2.4 Named Entity Tagging

We implemented a named entity tagger able to identify the entities in our ontology which makes use of both gazetteers and regular expressions. Table 4 shows the entity types and the methods employed in each case.

5.3 Ontology Schema

The proposed approach for relation extraction is thoroughly statistical. Therefore, it should be capable of extracting relations of any kind with the exception of the generic relations discussed in [38]. However for evaluation purposes we designed an ontology schema of biographical information by importing a DBpedia ontology dataset with minor modifications. Protégé ontology editor† was employed for the design purpose. The ontology is stored in RDF format and JENA semantic web framework‡ is used to insert newly extracted relations into the ontology. Table 5 represents an overview of the schema used in the design of our ontology. The purpose of this ontology is storing the biographical attributes of a person. That is why the subject of nearly all relations is a person entity and the object is an attribute of that person.

†http://protege.stanford.edu/
‡https://jena.apache.org/
6 Experimental Results

This section presents the results obtained from our approach in comparison to the baseline in order to assess the strength and utility of the proposed approach. We carried out an experiment with the whole corpus and the seed set as the input in order to compare the performance of our approach to that of the baseline in terms of precision at different ranks and also mean average precision.

Tables 6 and 7 show an overview of the number of relations extracted using this procedure in the first iterations for the baseline as well as our approach respectively. According to the statistics, the small seed set produces overall much fewer new relations as compared to the large seed set in case of the baseline. In contrast to the baseline, this difference is less in our approach. That means our approach is less sensitive to the number of initiating seeds than the baseline and it can generate more patterns and as a consequence introduces more new triples even in case of smaller seed sets. It is worth noting that the number of relations also varies greatly from one relation type to another: while e.g., date-of-birth is very productive, literary-works is much more rare.

Despite the fact that at each iteration only the top 10 records are used in the subsequent iterations, the top 50 retrieved records are evaluated for correctness. Since no ground through data is available for this task, the results of each iteration are manually assessed and precision at ranks 10, 20, and 50 as well as mean average precision are reported. Figure 2 shows these measures for the small seed set in the first iteration comparing our approach and the baseline. We can see that the results of our approach significantly outperform the baseline. Figure 3 shows the same metric using the large seed set in the first iteration. We have the same observation when comparing our method with baseline using large seed data. Comparing the results of using a smaller and larger seed set (Figure 2 vs Figure 3), we can see that while the baseline method fails to produce enough results for literary-works and tomb, our approach is able to produce results for all relations even with small seed, which shows the robustness of our model. It indicates that our model is less sensitive to the size of seed data; i.e., it is able to produce results even with small number of seed data.

To show the performance of our approach after passing some iterations, we also represent results of the 10th iteration in tables 8 and 9. As can be seen in the results, our approach outperforms the baseline in the next iterations too.

The difference in performance of the proposed method over different relation types can be partially caused by the natural difference of the patterns and grammar of sentences representing each relation. In addition to that, the discrepancy in the efficiency and precision of the named entity tagger we employed for detecting different entities can also be a reason to that.

It is also worth to mention that the above results are achieved while our study suffers from different limitations including:

- the limited size of corpus compared to related studies,
- the lack of a reliable and easy-to-use co-reference resolution mechanism
for Persian to facilitate detection of co-references mentions,

• lack of a high accurate NER tagger for Persian.

These issues do not detract from the project’s main aim of demonstrating the efficacy of the proposed approach in comparison to the baseline over a target language of subject-verb-object word order. However, possible solutions or improvements to the limitations mentioned above would increase the overall performance of the system we developed in the course of the current study.

7 Conclusion and Perspectives

In this research work, we proposed an approach to identify textual patterns for predefined semantic relations of a biographical domain from a corpus of Persian textual documents. For this purpose we gathered a corpus of raw text about people biographies. The corpus was then tagged with a named entity tagger developed in the course of current research work to specifically identify entities within the biography domain.

Our ontology schema consisted of a number of triple relations corresponding to biographical attributes of a person. The method work with some records of correct relations as seed data. The implemented system operates in an iterative manner to add newly extracted records of information to the knowledge base. Our method is thoroughly based on an statistical model of n-grams. The results of the experiment with the proposed method both with small and big seed data are promising, beating the suffix-tree method as the baseline.

Since the proposed method is entirely relying on statistical methods, we believe it should not be difficult to adapt it to new domains and new languages. The main motivation for this work was the difference in the structure of Persian sentences which, in contrary to English sentence structure, is in subject-object-verb format and as we showed state-of-the-art approaches do not work well on such languages. However, our method is not dependent on any specific sentence structure and we believe it has the potential to perform well on other languages too. Moreover, we showed that our approach does not depend on the size of seed data and can achieve reasonable performance even on small seed sets.

Expanding the current research on other domains and also evaluating the results in a full QA system is part of our future works.

Acknowledgment

The authors would like to thank Manuela Hürlimann and Esther van den Berg who were of great help with implementation of the baseline.
References


[34] Suchanek, F. M., Ifrim, G., and Weikum, G. “Combining linguistic and statistical analysis to extract relations from web documents”. In Proc. of the International Conference on Knowledge Discovery and Data Mining (KDD) (2006).


Biography

Dr. Saeedeh Monttazi is currently an assistant professor at Amirkabir University of Technology, Iran. She completed her BSc and MSc education at Sharif University of Technology, Iran. She received the PhD degree in Artificial Intelligence from Saarland University, Germany. As part of her PhD, she was a visiting researcher at the Center of Language and Speech Processing at Johns Hopkins University, US. After finishing the PhD, she worked at the Hasso-Plattner Institute at Potsdam University, Germany and the German Institute for International Educational Research, Germany as a post-doctoral researcher. Natural language processing is her main research focus. She has worked in this area of research for more than 12 years.
Figure 1: Bootstrapping system architecture

Table 1: Number of effective initializing seeds for each relation in the small set

<table>
<thead>
<tr>
<th>Relation Name</th>
<th>Total Number</th>
<th>Out-of-Corpus</th>
<th>Effective</th>
</tr>
</thead>
<tbody>
<tr>
<td>literary-works</td>
<td>152</td>
<td>126</td>
<td>26</td>
</tr>
<tr>
<td>tomb</td>
<td>12</td>
<td>5</td>
<td>7</td>
</tr>
<tr>
<td>date-of-birth</td>
<td>40</td>
<td>13</td>
<td>27</td>
</tr>
<tr>
<td>date-of-death</td>
<td>38</td>
<td>16</td>
<td>22</td>
</tr>
<tr>
<td>contemporaneity</td>
<td>33</td>
<td>18</td>
<td>15</td>
</tr>
<tr>
<td>literary-style</td>
<td>44</td>
<td>34</td>
<td>10</td>
</tr>
<tr>
<td>place-of-birth</td>
<td>30</td>
<td>14</td>
<td>16</td>
</tr>
<tr>
<td>living-place</td>
<td>12</td>
<td>3</td>
<td>9</td>
</tr>
<tr>
<td>place-of-death</td>
<td>18</td>
<td>10</td>
<td>8</td>
</tr>
<tr>
<td>religion</td>
<td>12</td>
<td>7</td>
<td>5</td>
</tr>
<tr>
<td>nationality</td>
<td>21</td>
<td>11</td>
<td>10</td>
</tr>
</tbody>
</table>

Table 2: Number of effective initializing seeds in small and big set

<table>
<thead>
<tr>
<th>Seed-set size</th>
<th>Minimum</th>
<th>Maximum</th>
</tr>
</thead>
<tbody>
<tr>
<td>small</td>
<td>7 (tomb)</td>
<td>27 (date-of-birth)</td>
</tr>
<tr>
<td>large</td>
<td>133 (literary-works)</td>
<td>166 (place-of-birth)</td>
</tr>
</tbody>
</table>
Table 3: Third-person pronouns in Persian

<table>
<thead>
<tr>
<th>Persian</th>
<th>English</th>
</tr>
</thead>
<tbody>
<tr>
<td>u</td>
<td>he/she</td>
</tr>
<tr>
<td>vei</td>
<td>he/she</td>
</tr>
<tr>
<td>xod</td>
<td>self</td>
</tr>
<tr>
<td>xodaš</td>
<td>himself/herself</td>
</tr>
<tr>
<td>xiš</td>
<td>self</td>
</tr>
<tr>
<td>'išän</td>
<td>he/she</td>
</tr>
<tr>
<td>'in šäer</td>
<td>this poet</td>
</tr>
</tbody>
</table>

Table 4: Methods used for tagging named entities

<table>
<thead>
<tr>
<th>Entity</th>
<th>Method</th>
<th>Format</th>
</tr>
</thead>
<tbody>
<tr>
<td>place</td>
<td>gazetteer + regular expression</td>
<td><a href="">PLACE:xxxxx</a></td>
</tr>
<tr>
<td>country</td>
<td>gazetteer</td>
<td><a href="">COUNTRY:xxxxx</a></td>
</tr>
<tr>
<td>nationality</td>
<td>gazetteer</td>
<td><a href="">NATIONALITY:xxxxx</a></td>
</tr>
<tr>
<td>religion</td>
<td>gazetteer</td>
<td><a href="">RELIGION:xxxxx</a></td>
</tr>
<tr>
<td>literary-style</td>
<td>gazetteer</td>
<td><a href="">STYLE:xxxxx</a></td>
</tr>
<tr>
<td>literary works</td>
<td>regular expression</td>
<td><a href="">BOOKS:xxxxx</a></td>
</tr>
<tr>
<td>date</td>
<td>regular expression</td>
<td><a href="">DATE:xxxxx</a></td>
</tr>
</tbody>
</table>

Table 5: Ontology Schema

<table>
<thead>
<tr>
<th>Subject</th>
<th>Predicate</th>
<th>Object</th>
</tr>
</thead>
<tbody>
<tr>
<td>person</td>
<td>literary-works</td>
<td>literary work</td>
</tr>
<tr>
<td>person</td>
<td>date-of-birth</td>
<td>date</td>
</tr>
<tr>
<td>person</td>
<td>date-of-death</td>
<td>date</td>
</tr>
<tr>
<td>person</td>
<td>place-of-birth</td>
<td>place</td>
</tr>
<tr>
<td>person</td>
<td>place-of-death</td>
<td>place</td>
</tr>
<tr>
<td>person</td>
<td>tomb</td>
<td>place</td>
</tr>
<tr>
<td>person</td>
<td>living-place</td>
<td>place</td>
</tr>
<tr>
<td>person</td>
<td>nationality</td>
<td>nationality</td>
</tr>
<tr>
<td>person</td>
<td>religion</td>
<td>religion</td>
</tr>
</tbody>
</table>
Table 6: Overview statistics of extracted relations by baseline

<table>
<thead>
<tr>
<th>Relation</th>
<th>Iteration</th>
<th>Small seed set</th>
<th>Large seed set</th>
</tr>
</thead>
<tbody>
<tr>
<td>literary-works</td>
<td>1</td>
<td>18</td>
<td>124</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>20</td>
<td>128</td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>45</td>
<td>128</td>
</tr>
<tr>
<td></td>
<td>4</td>
<td>45</td>
<td>128</td>
</tr>
<tr>
<td></td>
<td>5</td>
<td>44</td>
<td>130</td>
</tr>
<tr>
<td></td>
<td>6</td>
<td>44</td>
<td>130</td>
</tr>
<tr>
<td></td>
<td>7</td>
<td>–</td>
<td>130</td>
</tr>
<tr>
<td></td>
<td>8</td>
<td>–</td>
<td>130</td>
</tr>
<tr>
<td></td>
<td>9</td>
<td>–</td>
<td>131</td>
</tr>
<tr>
<td></td>
<td>10</td>
<td>–</td>
<td>133</td>
</tr>
<tr>
<td>tomb</td>
<td>1</td>
<td>0</td>
<td>882</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>0</td>
<td>883</td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>–</td>
<td>883</td>
</tr>
<tr>
<td></td>
<td>4</td>
<td>–</td>
<td>884</td>
</tr>
<tr>
<td></td>
<td>5</td>
<td>–</td>
<td>884</td>
</tr>
<tr>
<td></td>
<td>6</td>
<td>–</td>
<td>884</td>
</tr>
<tr>
<td></td>
<td>7</td>
<td>–</td>
<td>884</td>
</tr>
<tr>
<td></td>
<td>8</td>
<td>–</td>
<td>884</td>
</tr>
<tr>
<td></td>
<td>9</td>
<td>–</td>
<td>884</td>
</tr>
<tr>
<td></td>
<td>10</td>
<td>–</td>
<td>884</td>
</tr>
<tr>
<td>date-of-birth</td>
<td>1</td>
<td>1649</td>
<td>1654</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>1649</td>
<td>1654</td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>1649</td>
<td>1654</td>
</tr>
<tr>
<td></td>
<td>4</td>
<td>1750</td>
<td>1654</td>
</tr>
<tr>
<td></td>
<td>5</td>
<td>1750</td>
<td>1654</td>
</tr>
<tr>
<td></td>
<td>6</td>
<td>1750</td>
<td>1656</td>
</tr>
<tr>
<td></td>
<td>7</td>
<td>1750</td>
<td>1688</td>
</tr>
<tr>
<td></td>
<td>8</td>
<td>1752</td>
<td>1688</td>
</tr>
<tr>
<td></td>
<td>9</td>
<td>1752</td>
<td>1688</td>
</tr>
<tr>
<td></td>
<td>10</td>
<td>1752</td>
<td>1688</td>
</tr>
</tbody>
</table>
Table 7: Overview statistics of extracted relations by our approach.

<table>
<thead>
<tr>
<th>Relation</th>
<th>Iteration</th>
<th>Small seed set</th>
<th>Large seed set</th>
</tr>
</thead>
<tbody>
<tr>
<td>literary-works</td>
<td>1</td>
<td>476</td>
<td>467</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>486</td>
<td>467</td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>502</td>
<td>467</td>
</tr>
<tr>
<td></td>
<td>4</td>
<td>502</td>
<td>467</td>
</tr>
<tr>
<td></td>
<td>5</td>
<td>511</td>
<td>467</td>
</tr>
<tr>
<td></td>
<td>6</td>
<td>511</td>
<td>467</td>
</tr>
<tr>
<td></td>
<td>7</td>
<td>511</td>
<td>467</td>
</tr>
<tr>
<td></td>
<td>8</td>
<td>511</td>
<td>504</td>
</tr>
<tr>
<td></td>
<td>9</td>
<td>511</td>
<td>504</td>
</tr>
<tr>
<td></td>
<td>10</td>
<td>511</td>
<td>504</td>
</tr>
<tr>
<td>tomb</td>
<td>1</td>
<td>1370</td>
<td>1692</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>1406</td>
<td>1692</td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>1587</td>
<td>1663</td>
</tr>
<tr>
<td></td>
<td>4</td>
<td>1587</td>
<td>1663</td>
</tr>
<tr>
<td></td>
<td>5</td>
<td>1738</td>
<td>1663</td>
</tr>
<tr>
<td></td>
<td>6</td>
<td>1768</td>
<td>2193</td>
</tr>
<tr>
<td></td>
<td>7</td>
<td>1877</td>
<td>2193</td>
</tr>
<tr>
<td></td>
<td>8</td>
<td>1794</td>
<td>2193</td>
</tr>
<tr>
<td>place-of-birth</td>
<td>1</td>
<td>1261</td>
<td>1290</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>1261</td>
<td>1248</td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>1115</td>
<td>1248</td>
</tr>
<tr>
<td></td>
<td>4</td>
<td>1493</td>
<td>1248</td>
</tr>
<tr>
<td></td>
<td>5</td>
<td>1493</td>
<td>1248</td>
</tr>
<tr>
<td></td>
<td>6</td>
<td>1289</td>
<td>1248</td>
</tr>
<tr>
<td></td>
<td>7</td>
<td>1289</td>
<td>1248</td>
</tr>
<tr>
<td></td>
<td>8</td>
<td>1289</td>
<td>1248</td>
</tr>
<tr>
<td></td>
<td>9</td>
<td>1289</td>
<td>1248</td>
</tr>
<tr>
<td></td>
<td>10</td>
<td>1289</td>
<td>1248</td>
</tr>
</tbody>
</table>

Table 8: Results at the 10th iteration: small seed set, baseline

<table>
<thead>
<tr>
<th>slot/metric</th>
<th>P@10</th>
<th>P@20</th>
<th>P@50</th>
<th>MAP</th>
</tr>
</thead>
<tbody>
<tr>
<td>literary-works</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td>tomb</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td>place-of-birth</td>
<td>0.7</td>
<td>0.6</td>
<td>0.54</td>
<td>0.669</td>
</tr>
<tr>
<td>date-of-birth</td>
<td>1</td>
<td>0.9</td>
<td>0.88</td>
<td>0.920</td>
</tr>
<tr>
<td>date-of-death</td>
<td>0.1</td>
<td>0.1</td>
<td>0.1</td>
<td>0.129</td>
</tr>
</tbody>
</table>
Figure 2: Comparing the results of our approach and the baseline at the first iteration for small seed set.

Figure 3: Comparing the results of our approach and the baseline at the first iteration for large seed set.
Table 9: Results at the 10th iteration: small seed set, our approach

<table>
<thead>
<tr>
<th>slot/metric</th>
<th>P@10</th>
<th>P@20</th>
<th>P@50</th>
<th>MAP</th>
</tr>
</thead>
<tbody>
<tr>
<td>literary-works</td>
<td>0.6</td>
<td>0.8</td>
<td>0.8</td>
<td>0.761</td>
</tr>
<tr>
<td>tomb</td>
<td>0.1</td>
<td>0.25</td>
<td>0.24</td>
<td>0.238</td>
</tr>
<tr>
<td>place-of-birth</td>
<td>0.6</td>
<td>0.65</td>
<td>0.66</td>
<td>0.681</td>
</tr>
<tr>
<td>date-of-birth</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>date-of-death</td>
<td>1</td>
<td>1</td>
<td>0.92</td>
<td>0.975</td>
</tr>
</tbody>
</table>