Word Sense Induction in Persian and English: A Comparative Study

Masood Ghayoomi
Faculty of Linguistics, Institute for Humanities and Cultural Studies, Tehran, Iran
m.ghayoomi@ihcs.ac.ir

Received: 05/Jan/2021 Revised: 20/Jun/2021 Accepted: 20/Jul/2021

Abstract
Words in the natural language have forms and meanings, and there might not always be a one-to-one match between them. This property of the language causes words to have more than one meaning; as a result, a text processing system faces challenges to determine the precise meaning of the target word in a sentence. Using lexical resources or lexical databases, such as WordNet, might be a help, but due to their manual development, they become outdated by passage of time and language change. Moreover, the lexical resources might be domain dependent which are unusable for open domain natural language processing tasks. These drawbacks are a strong motivation to use unsupervised machine learning approaches to induce word senses from the natural data. To reach the goal, the clustering approach can be utilized such that each cluster resembles a sense. In this paper, we study the performance of a word sense induction model by using three variables: a) the target language: in our experiments, we run the induction process on Persian and English; b) the type of the clustering algorithm: both parametric clustering algorithms, including hierarchical and partitioning, and non-parametric clustering algorithms, including probabilistic and density-based, are utilized to induce senses; c) the context of the target words to capture the information in vectors created for clustering: for the input of the clustering algorithms, the vectors are created either based on the whole sentence in which the target word is located; or based on the limited surrounding words of the target word. We evaluate the clustering performance externally. Moreover, we introduce a normalized, joint evaluation metric to compare the models. The experimental results for both Persian and English test data showed that the window-based partitioningK-means algorithm obtained the best performance.

Keywords*:* Corpus Linguistics; Word Sense Induction; Clustering; Word Embedding; Sense Embedding; Parametric Clustering; Non-parametric Clustering; Joint Evaluation Metric.

1- Introduction

Language, as a means of communication between human beings, is composed of two components [1]: form, and meaning. The ‘form’ can be represented either via an audio signal transmitted through a voice channel from a speaker to a recipient, or via an orthographic form through the writing system and the alphabetical set of the language. In text processing, the orthographic form of the language is taken into consideration. Ambiguity is a property of a natural language that causes challenges in text processing. There exist two types of ambiguities: a) syntactic ambiguity, and b) lexical ambiguity. The sentence ‘I saw the man with a telescope.’, for instance, is a sample of syntactic ambiguity to either mean ‘I used a telescope to see the man’ or ‘I saw the man who carried a telescope’.

There are two reasons to cause lexical ambiguity [2, p: 146]: (a) polysemy where a word has more than one meaning, such as /rošan/ (light/bright) in /ran ge rošan/ (light color) and /ʔotāqe rošan/ (bright room) in Persian or ‘plane’ in ‘fly by plane’ and ‘cut by plane’ in English; and (b) homonymy where the word is both homograph and homophone, such as /rox/ (rook/face/roc) in /mohreye rox/ (the rook piece [in chess]), /roxe zibāye ʔu/ (her beautiful face), and /parandeye rox/ (the roc bird) in Persian or ‘bank’ (financial place/side of river) in English. In Example (1)a-f, the sentences that contain the target word ‘bank’ are grouped (clustered) in Figure 1. Based on the semantic similarity of the target word ‘bank’ in the sentences, one group belongs to the concept ‘financial place’ (*bank*$_1$) and the other group belongs to the concept ‘side of river’ (*bank*$_2$).

(1) a. He cashed a check at the bank.
b. She sat on the bank of the river and watched the currents.
c. They detected frauds in the bank.
d. I saw a deer near the river bank.
e. That bank holds the mortgage on my home.
f. They pulled the canoe up on the bank.

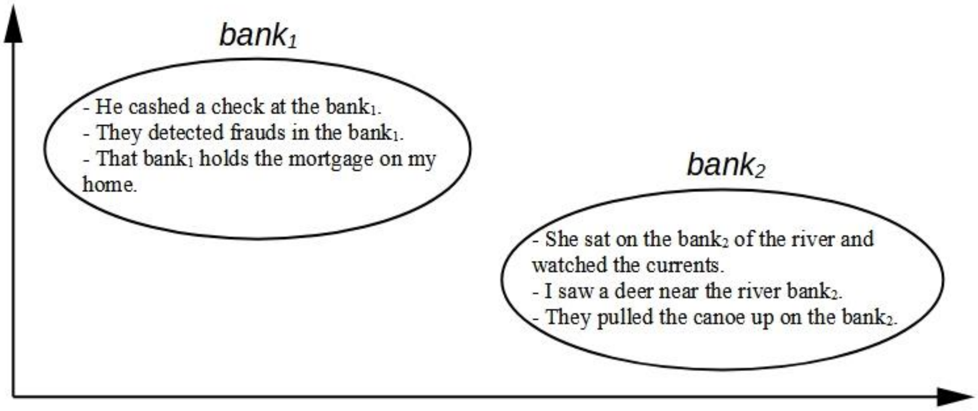
bank1
- He cashed a check at the bank1.
- They detected frauds in the bank1.
- That bank1 holds the mortgage on my home.
bank2
- She sat on the bank2 of the river and watched the currents.
- I saw a deer near the river bank2.
- They pulled the canoe up on the bank2.

Figure 1: Clustering result of instances for the target word "bank"

The lexical ambiguity in text processing is more pronounced in languages that use the Arabic script in their writing system, such as Persian, due to avoiding writing short vowels than languages that use the phonemic orthography, such as English. In text processing, both polysemy and homonymy are recognized as one problem. The context of the target ambiguous word plays a very important role to determine and to disambiguate the meaning.

The Word Sense Induction (WSI) task means that the machine has to induce word senses from the natural data automatically without prior knowledge. This task uses an unsupervised machine learning approach and it can be defined as a clustering task. The example in Figure 1 represents the idea of how clustering can identify the senses of a word. One property of this task is that no initial training data is required.

This paper focuses on WSI and aims at inducing the meaning of both polysemous and homonymous Persian and English words from their local contexts and comparing the performance of the clustering algorithms. One additional contribution of this paper is introducing a normalized, joint, external evaluation metric to be able to compare the models more accurately against the naïve baselines.

The construction of the paper is as follows: after the introduction, in Section 2, we describe the semantic representation methods to be used for the clustering task. Section 3 reviews the related works on WSI. In Section 4, our models for both Persian and English are proposed. The obtained results as well as our proposed, joint evaluation metric are discussed in Section 5; and finally, the paper is concluded in Section 6.

2- Semantic Representation

2-1- Distributional Semantics

Ambiguity is one of the properties of the natural language. According to the idea proposed by Wittgenstein [3], the meaning of a word can be determined by its usage in the language. Following this idea, Harris [4] proposed an idea in the framework of 'distributional semantics' such that the words which are used in the same local contexts intend to have a similar meaning. Based on this idea, the 'distributional hypothesis' was proposed, and Firth [5] emphasized that "the local context of the word plays an important role in determining words' senses". Miller and Charles [6] proposed 'strong contextual hypothesis' such that two words are to some extent semantically similar if they have similar contexts. Based on this hypothesis, the words 'year', 'date', and 'Wednesday' in Example (2) are semantically similar.

(2) a. I go to the cinema this year.
b. I go to the cinema on this date.
c. I go to the cinema this Wednesday.

Since the context plays a very important role to capture the meaning of a word, precise encoding of the word's context information is required. To this end, Peirsman and Geeraerts [7] introduced three types of linguistic contexts to be extracted from a large corpus: a) document-based model where the words in the same paragraph or in the same documents are used as the context [8, 9]; b) syntax-based model where words are compared according to their syntactic relations, from dependency relations [10, 11, 12, 13] to the combinatory categorial grammar [14]; and c) word-based model where word-word co-occurrence statistics are extracted from a fixed window size. These word co-occurrences resemble the ‘bag-of-words’ model [9].

Song et al. [15] introduced two general approaches to represent context information in ‘distributional semantics’: a) using the Bayesian model utilized in topic modeling [16], and b) using a feature-based model to represent the semantic information as a vector. The latter model uses a vector space model to represent the vectorized semantic information of words. The vectors can be used in the clustering task to induce words’ meanings. The advantage of using a vector space model is compressing the information about the words and their contexts, called ‘word embedding’. Computing the geometric distance between the vectors makes it possible to decide how two words intend to be similar. Euclidean distance and Cosine distance are two well-known methods for computing the geometric distance between the vectors [17]. However, there are studies that try to better represent the distributional semantics by combining word embeddings with the knowledge-bases known as ‘knowledge embedding model’ [18], enriching word embeddings with ontologies [19], and utilizing a contextualized knowledge embedding model as a joint model where word embedding and sense embedding (sense representations of the words in the local context from corpora that are sense tagged) are combined with knowledge-bases [20].

2-2- Modeling Methods

To use word embedding methods for capturing the local context information of a word and compressing the information to be represented in a vector, two methods can be utilized: a) using the matrix decomposition techniques, and b) using the neural network-based techniques. The Global Vector (GloVe) representation [21] uses the matrix decomposition technique to provide the distributional representation of words. Continuous Skip gram (Skip-gram) and Continuous Bag Of Words (CBOW) models [22] use the neural network-based technique to represent the contextual information of a word in a vector. In this paper, we use the Skip-gram model for capturing contextual information of the target word in a vector.

2-3- Context Clustering

There are two major clustering algorithms in terms of defining the number of clusters: parametric and non-parametric. The two well-known parametric clustering algorithms commonly used in natural language processing applications are partitioning and hierarchical. Partitioning clustering uses a centroid-based clustering and computes the distance of individual vectors to the centroid, such as the K-means algorithm [23]. The hierarchical clustering uses a statistical criterion to compute the clusters’ distance. This algorithm is either agglomerative (bottom-up) or divisive (top-down). We use the divisive clustering algorithm for the WSI task.

The common property of parametric algorithms is that they require a pre-defined number of clusters. Therefore, the State-Of-The-Art (SOTA) techniques in the field have performed their experiments on a pre-defined number of clusters; e.g., the proposed model by used the K-means algorithm with 3 clusters, i.e. *k=3*. To have a better estimation on the number of clusters, Ghayoomi [25] utilized the silhouette score [26] in Equation (1) as a metric to define the number of clusters. Using this method to identify the number of clusters outperformed the SOTA results.

$$s(i) = \frac{b(i) - a(i)}{\max\{a(i), b(i)\}} \tag{1}$$

where *a*(*i*) is the average dissimilarity of element *i* with other elements in the same cluster computed by Equation (2); and *b*(*i*) is the minimum distance between an element of a cluster with all other elements in the rest of clusters, computed by Equation (3).

$$a(i) = \frac{1}{|C_i| - 1} \sum_{j \in C_i, i \neq j} d(i, j) \tag{2}$$

$$b(i) = \min_{k \neq i} \frac{1}{|C_k|} \sum_{j \in C_k} d(i, j) \tag{3}$$

where *i* and *j* are two elements in cluster *C* and *d*(*i*,*j*) is the distance between *i* and j, C_i is the cluster in which element *i* belongs to and *j* is another element of this cluster, and C_k is cluster that element *i* is not its member.

In this research, we use the silhouette score as a metric for each cluster to decide about the best number of clusters: the higher the score, the better the clustering result.

Non-parametric clustering algorithms are another approach for the WSI task. The number of senses (clusters) is unknown in advance and the algorithms should try to find the senses. Chinese Restaurant Processing (CRP) [27] and Density-Based Spatial Clustering of Applications with Noise (DBSCAN) [28] are two non-parametric clustering algorithms that we use for this goal. CRP models the behavior of Chinese when they go to a restaurant: either to sit on a table that one has already sat on, or to take a new seat. The algorithm uses a probabilistic Bayesian model. The DBSCAN uses a density-based model to find the best partitioning of clusters.

In this paper, we compare the performance of both parametric and non-parametric algorithms for the WSI task in Persian and English. The study of the algorithms themselves and their properties are out of the scope of this paper.

3- Related Works on WSI

Clustering the context to distinguish senses of the target polysemous or homonymous word is one of the main approaches in WSI. In this approach, the number of clusters indicates the number of the target word's senses. Huang et al. [24] used the K-means algorithm with word embedding to cluster word contexts. Neelakantan et al. [29] predicted each sense of a word as a context cluster assignment. Their model worked based on the K-means algorithm. In these two researches, a fixed number of clusters, namely 3 clusters, was defined to run the K-means clustering. Li and Jurafsky [30] proposed using CRP as a non-parametric model to capture the senses dynamically. In their approach, the model decided either to generate a new sense for each context or to assign the context to an already generated sense. Wang et al. [31] proposed a model to use weighted topic modeling for sense induction. Amrami and Goldberg [32] utilized the BiML model, a bidirectional recurrent neural network model, proposed by Peters et al. [33] for WSI and extended the model such that predicted word probabilities were used in the language model. Alagić et al. [34] used the lexical substitution model to induce word senses. Therefore, words which belonged to a cluster should be able to be substituted in an appropriate context. The proposed model was compared against manual substitution along with other clustering evaluation metrics. Corrêa and Amancio [35] proposed a model to capture the structural relationship among contexts. To this end, they used the complex network proposed by Perozzi et al. [36] for context embedding. Tallo [37] used sentence embedding for WSI and investigated the encoding of linguistic properties of words in the embedding. Dong and Wang [38] used WSI in the medical domain to enhance sense inventories. They evaluated four models, namely using context clustering, two types of word clustering, and sparse coding in word vector space. Among them, the sparse coding model proposed by Arora et al. [39] outperformed the other models to discover more complete word senses.

As reported by Song et al. [15], the K-means parametric model used by Neelakantan et al. [29] outperforms the CRP algorithm proposed by Li and Jurafsky [30] based on the SemEval2010 WSI task [40]. As Song et al. [15] stated, the main reason for obtaining such results is the poor performance of CRP in making a decision to assign a word to a new cluster. In the results of the two models, the K-means algorithm used 3 clusters as the predefined, fixed number of clusters, while CRP ended to a lesser number of clusters on average than the best average number of clusters for both noun and verb categories in the SemEval2010 WSI task. This indicates that relaxing the predefined number of clusters in K-means can further improve the performance of the task.

4- Architecture of the Proposed Model

The clustering model we proposed in our research is represented in Figure 2. As can be seen in the figure, the model is constructed of three modules and datasets which are described below.

4-1- Major Modules of the Model

The model contains three modules: vectorization, clustering, and evaluation. In vectorization, first the words' vectors based on the big corpus of a language described in Section 4.2 are created. In vectorization of words, three parameters should be taken into consideration in advance: a) the number of dimensions of each vector; b) the number of the surrounding words of the target word in the local context; c) the information to be considered in vectorization which is the word forms in our case. The setting of the parameters is described in Section 5.3. The vector of the instances that contain the target word is created in two modes: a) in the first mode, thereafter called the 'SentContext' mode, the weighted vectors of the words in a sentence are summed up to build the vector of each instance that includes the target word. Then, this score is normalized based on the sentence length. In the second mode, thereafter called the 'WinContext', the limited surrounding context of the target word is used to build the sentence vector.

It has to be mentioned that not all words in a sentence are content words and there exists a closed list of functional words frequently used, such as preposition, conjunctions, coordinators, etc. These words can be considered as stop words. We use a weighting method to increase the impact of content words, and reduce the impact of functional words. To this end, we use TF-IDF[1] [41] to assign a weight to the words.

In the next step, all instances of the target word are clustered based on their vector representation. We assume that each cluster shows one sense of the word. In the clustering module, we utilize both parametric and non-parametric clustering algorithms described in Section 2.3. The parametric algorithms are run based on the two context modes. For clustering, the data should be reformatted from word forms to a vector space model described above. More precise vectors result in better clustering performance.

It should be added that a two step embedding process

[1] Term Frequency-Inverse Document Frequency

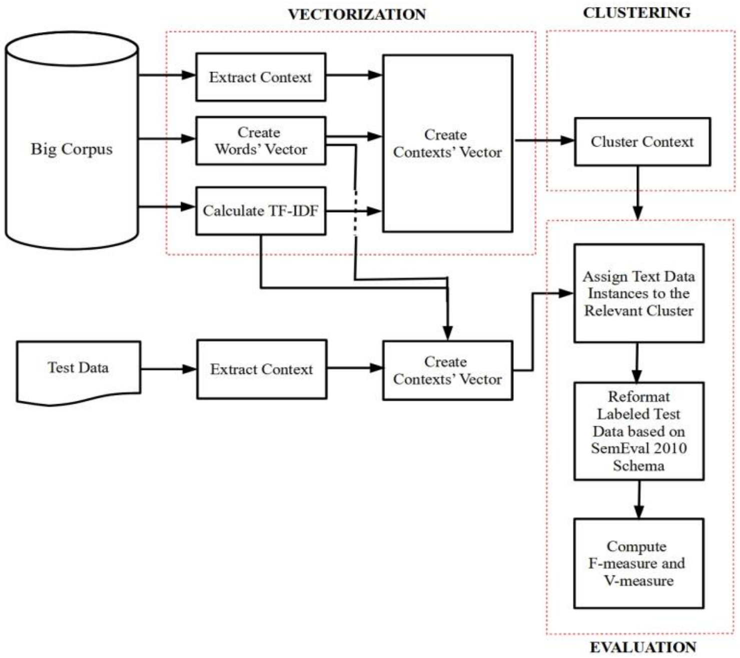

Figure 2: Architecture of our proposed model for WSI

is used in this model. The word embedding is first done based on the semantic distribution of words in a language. And after the clustering step, sense embedding is done for semantic distribution of the target word with respect to its meaning in the local context.

In the evaluation module, two evaluation criteria, namely F-measure and V-measure in addition to a joint metric, are used. These metrics are explained in more detail in Section 5.2. In the evaluation process, the instances of the test data are added to the data pool to be clustered and the induction results of the test data are compared with the corresponding gold standard labels. To this end, we used the toolkit developed in SemEval2010 WSI task [40] that does this mapping.[1]

4-2- Datasets

To run our experiments, we require three datasets for Persian and English: a big corpus, data pool, and test data. The big corpus is used for training word embedding as well as sense embedding to identify the senses of the target words based on the clustering output. The data pool is used for clustering the target words based on their context; and the test data is used for evaluating the models.

The big corpus that we use for creating the Persian words embedding contains over 538 million word tokens developed by Ghayoomi [42]. This corpus is a composition of several other corpora, including a) The Pesian Linguistic DataBase [43] which is a balanced Persian corpus containing both historical and contemporary Persian. In this research, we only use the contemporary dataset; b) The Newspaper Corpus which is a collection of news crawled from the online archive of several Persian newspapers; c) The Hamshahri Corpus [44] which is also another news corpus collected from the online archive of the Hamshahri Newspaper; d) The Bijankhan Corpus [45] which is a fraction of Peykare [46], the Persian Text Corpus; and e) The Persian Wikipedia corpus which contains 361,479 articles downloaded from the dump of Persian Wikipedia articles in July 2016.[2]

The big corpus that we use for creating English word vectors is the Westbury Lab Wikipedia Corpus developed by Shaoul and Westbury [47]. This corpus, which is freely available, is collected from the dump of English Wikipedia articles in April 2010. The corpus contains almost 990 million word tokens of the general domain and it has been used for similar tasks as reported in the literature [24, 29]. It should be mentioned that the documents with less than 2000 characters long are excluded from the corpus.

To evaluate the clustering results of the Persian WSI experiments, we use the test data developed by Ghayoomi [42]. This dataset is standardized based on the SemEval2010 framework. In this dataset, 20 Persian words which are either polysemous or homonymous, are selected from Farsnet [48], the Persian Wordnet. For each target word, 100 sentences are manually annotated; as a result,

[1] https://www.cs.york.ac.uk/semeval2010_WSI/files/evaluation.zip

[2] https://archive.org/details/fawiki-20160720

the test dataset contains 2000 instances in total. Moreover, 279,567 unannotated sentences which contain any of the target words are selected from the big corpus as the data pool.

To evaluate the clustering results of the English WSI experiments, we use the SemEval2010 dataset for the WSI task [40] that is mostly from the news domain. In total, 100 words (50 verbs and 50 nouns) are the target words in this dataset. This dataset contains 8,915 instances as test data with sense annotation and 888,722 unannotated sentences in the data pool. Table 1 summarizes the statistical information of the data pool, the test data, and the size of the big corpus for Persian and English.

Table 1: Statistical information of test and train datasets for Persian and English

Language			**Persian**	**English**
Data	**Pool**	**Instance (sentence)**	279,567	888,722
	Test	**Target Words**	20	100
		Instance (sentence)	2,000	8,915
		Average Sense	6.15	5.04
		Average Instance	100	89.15
	Big Corpus for Embedding	**Word Token**	538 million	990 million

5- Experimental Results

5-1- Baselines

To evaluate the performance of the clustering algorithms, we use two naïve baselines introduced in SemEval2010 [40]: a) the Most Frequent Sense (MFS) where all instances are assigned to a single cluster that contains the most frequent sense; b) one sense per cluster, thereafter called 1S1C, where each instance is assigned to an individual cluster; therefore the number of clusters is equal to the number of instances.

In addition, there are two SOTA results reported in the literature: a) the CRP algorithm utilized by Li and Jurafsky [30] for non-parametric clustering; and b) the K-means algorithm proposed by Neelakantan et al. [29] for parametric clustering. In this K-means algorithm, there is no optimization on the number of clusters and 3 senses are assumed as the pre-defined number of senses for each English word. Thereafter, we call this model K-means-3.

All of the basic baselines and the SOTA models are performed with the Persian data to compare the clustering performance, disregarding the dependency of the algorithm to the data.

5-2- Evaluation Metrics

To evaluate the performance of the clustering results, we utilize two known external evaluation metrics which are commonly used for WSI, namely F-measure [49] and V-measure [50]. In addition, we propose a new normalized, joint evaluation metric, called J-measure, for a fair evaluation of the models.

5.2.1 F-Measure

F-measure computes the accuracy of information retrieval as in Equation (4).

$$F - measure = \frac{(1+\beta) \times P \times R}{(\beta \times P) + R} \tag{4}$$

where P is precision, R is recall, and β is a weighting parameter. If $\beta > 1$, more weight is assigned to recall, and in case $\beta < 1$, more weight is assigned to precision. If $\beta = 1$, precision and recall are considered equally. Equations (5) and (6) compute precision and recall, respectively. In all equations, K is the CLUSTER set, which is the hypothesized clusters from the clustering output and C is the CLASS set, which is the correct partitioning of the data; i.e., for a target dataset with N elements, we have two partitions: the guess partition K, and the gold partition C.

$$P = \frac{n_{ij}}{|k_i|} \tag{5}$$

$$R = \frac{n_{ij}}{|c_i|} \tag{6}$$

where n_{ij} is the number of members of class $ci \in C$ that is the element of cluster $kj \in K$.

5.2.2 V-Measure

Another alternative to evaluate clustering is an entropy-based approach proposed by Rosenberg and Hirschberg [50]. Different entropy-based evaluation metrics have been proposed for clustering so far [51, 52]. Among them, the V-measure metric proposed by Rosenberg and Hirschberg [50] is the most popular one. V-measure computes the harmonic mean of homogeneity, h, and completeness, c, of clustering as stated in Equation (7).

$$V - measure = \frac{(1+\beta) \times h \times c}{(\beta \times h) + c} \tag{7}$$

Homogeneity means that in each CLUSTER, there are a few numbers of CLASSes. The best mode of homogeneity is when a cluster consists of only samples of one class. Completeness, which is the reverse of homogeneity, means that each CLASS is appeared in a few numbers of CLUSTERs. The best mode of completeness is when all samples of the same class are within a single cluster.

As Rosenberg and Hirschberg [50] explained, homogeneity and completeness are formally defined in (8) and (9):

(8)

$$h = \begin{cases} 1 & if \quad H(C,K) = 0 \\ 1 - \frac{H(C|K)}{H(C)} & else \end{cases}$$

where

$$H(C|K) = -\sum_{k=1}^{|K|}\sum_{c=1}^{|C|}\frac{a_{ck}}{N} log \frac{a_{ck}}{\sum_{c=1}^{|C|} a_{ck}}$$

$$H(C) = -\sum_{c=1}^{|C|}\frac{\sum_{k=1}^{|K|} a_{ck}}{N} log \frac{\sum_{k=1}^{|K|} a_{ck}}{N}$$

(9)

$$c = \begin{cases} 1 & \text{if } H(K) = 0 \\ 1 - \frac{H(K|C)}{H(K)} & \text{else} \end{cases}$$

where

$$H(K|C) = -\sum^{|K|}\sum^{|C|}\frac{a_{ck}}{\,} log \frac{a_{ck}}{\,}$$

$$H(K) = -\sum_{k=1}^{|K|}\frac{\sum_{c=1}^{|C|} a_{ck}}{N} log \frac{\sum_{c=1}^{|C|} a_{ck}}{N}$$

C={c_i | i = 1, ..., n} is the set of CLASS, K = {k_i | 1, …, m} is the set of CLUSTER, and N is the number of data points in the data set, and a_{ck} is the number of elements of class c in cluster k.

5.2.3 The Proposed Evaluation Metric to Evaluate the Clustering Performance

The advantage of V-measure over F-measure is that in the evaluation, completeness as well as homogeneity are taken into consideration, while in F-measure only the distribution of classes in clusters, i.e. homogeneity in the clustering, is considered and it does not care about whether in each cluster the number of classes are minimized. This difference indicates that V-measure is more reliable than F-measure. On the other hand, V-measure alone dedicates a high score to the partitioning with one instance per cluster, because in such partitioning the number of classes in each cluster is perfectly minimized. This indicates that despite the advantages of V-measure, it is not a reliable metric. Therefore, to accurately evaluate the performance of the clustering result, we need to consider both metrics.

The results of the two metrics represent two extremes such that there is a trade-off between them, i.e. in most of the cases if V-measure is high, F-measure is low, and vice versa. For instance, if the SOTA scores based on V- and F-measures are compared against naïve baselines in the WSI task, it can be determined that the naïve baselines, namely 1S1C and MFS, obtain better scores than the advanced SOTA clustering algorithms and the SOTA models are not able to beat the simple baselines. This determines that V- and F-measures in Equations (4) and (7) are not perfect to compare the clustering performance accurately. As a result, we propose a normalized, joint metric, called J-measure in Equation (10) which is the harmonic mean of V- and F-measures. The obtained score is uniformed such that both homogeneity and completeness are included.

(10)

$$J - measure = \frac{(1+\beta) \times F \times V}{(\beta \times F) + V}$$

where F is F-measure and it obtains the result from Equation (4), V is V-measure and it obtains the result from Equation (7), and β is the weighting parameter. If β > 1, then more weight is assigned to F-measure; therefore only homogeneity in clustering is considered. In case β < 1, then more weight is assigned to V-measure to consider both homogeneity and completeness. If β = 1, then there is a uniform distribution over F- and V-measure.

If β = 1 in Equations (4), (7) and (10), then Equation (10) can be rewritten as Equation (11) to show how precision, recall, homogeneity, and completeness can relate to each other:

(11)

$$J - measure = \frac{2 \times \frac{2PR}{P+R} \times \frac{2HC}{H+C}}{(\frac{2PR}{P+R}) + (\frac{2HC}{H+C})}$$
$$= \frac{8PRHC}{(2PRH) + (2PRC) + (2PHC) + (2RHC)}$$
$$= \frac{8PRHC}{2(PRH + PRC + PHC + RHC)}$$
$$= \frac{4PRHC}{PR(H+C) + HC(P+R)}$$

Table 2: Results of the baselines, SOTAs, and the experimented models for Persian according to V-measure (V), F-measure (F) and J-measure (J) criteria

	Model	V (%)	H (%)	C (%)	F (%)	P (%)	R (%)	J (%)
Baseline	1S1C	**37.33**	100	23.57	00.07	00.70	00.04	00.13
	MFS	00.07	00.04	96.06	**59.51**	44.65	99.82	00.14
SOTA	CRP: SentContext	12.92	14.30	28.86	50.67	51.45	59.58	20.59
	Kmeans-3: SentContext	26.70	21.91	34.18	51.84	42.62	66.16	**35.25**
Models	DBSCAN: SentContext	02.36	1.36	45.94	**59.26**	44.82	97.83	04.53
	Kmeans-3: WinContext	31.97	30.29	35.61	56.09	54.14	58.18	40.72
	Divisive: SentContext	24.00	21.95	27.39	56.79	55.61	60.23	33.73
	Divisive: WinContext	29.63	26.49	36.41	59.94	56.21	66.56	39.43
	Kmeans- silhouette: SentContext	26.20	22.08	32.23	42.56	33.56	58.17	32.43
	Kmeans- silhouette: WinContext	**34.61**	37.95	34.64	51.95	61.73	50.27	**41.54**

Table 3: Results of the baselines, SOTAs, and the experimented models for English according to V-measure (V), F-measure (F) and J-measure (J) criteria

	Model	V (%)	H (%)	C (%)	F (%)	P (%)	R (%)	J (%)
Baseline	1S1C	**31.70**	23.51	48.66	00.09	00.05	00.50	**17.95**
	MFS	00.00	00.00	00.00	**63.40**	47.55	95.23	00.00
SOTA	CRP: SentContext	05.70	03.13	31.88	55.30	42.41	79.46	10.35
	Kmeans-3: SentContext	09.80	05.37	56.18	55.10	41.94	80.31	16.64
Models	DBSCAN: SentContext	04.66	03.20	44.78	**61.20**	46.84	93.99	08.66
	Kmeans-3: WinContext	15.63	16.22	18.12	49.30	49.96	52.81	23.74
	Divisive: SentContext	14.87	14.13	21.12	37.42	36.71	43.81	21.28
	Divisive: WinContext	16.15	12.57	43.51	53.77	48.69	65.61	24.84
	Kmeans- silhouette: SentContext	18.36	20.77	17.91	47.75	51.94	47.65	26.52
	Kmeans- silhouette: WinContext	**19.74**	23.90	18.09	43.03	52.00	39.21	**27.06**

5-3- Setup of Experiments

In this study, we experimentally compare the performance of several clustering algorithms to induce Persian and English word senses. The clustering algorithms require vector representation of the data. To this end, the Gensim Python[1] library is utilized to create the words' vectors according to this setups: a) employing the skip-gram model to capture the context of words; b) setting 8 words (4 words before and 4 words after the target word) similar to Huang et al. [24] to extract the information of the words' local contexts; c) setting the vector size to 300 dimensions similar to Neelakantan et al. [29]; and d) using the words with frequency 5 and above to build words' vector. In the next step, the weighted average of words' vector is created from the context vectors. Then, we compute TF-IDF of each word based on the idea proposed by Neelakantan et al. [29] and use it as a weighting value in each vector to compute the context vector.

The partitioning and hierarchy-based clustering algorithms are run in two modes, SentContext and WinContext modes, described in Section 4.1. In the WinContext mode, the context is set to 8 words to be similar to the context to build the words' vector. As a result, we perform our experiments by considering 4 words before and 4 words after the target word.

We also compute the two-tailed *t*-test to compare the performance of the models and study how statistically significant the difference between the models is.

5-4- Results and Discussion

Tables 2 and 3 summarize the obtained results of using various algorithms for inducing Persian and English words' senses. Among the basic baselines, the 1S1C has obtained a higher score for V-measure than the MSF baseline, but the score of F-measure is the lowest. The obtained results for the MFS baseline are vice-versa. Although the 1S1C baseline considers homogeneity and completeness properties, the MFS baseline takes only homogeneity into consideration.

[1] https://radimrehurek.com/gensim/index.html

Among the two clustering approaches used for the SOTA models, the parametric clustering algorithm implemented in the Kmeans-3 model obtained a higher result than the CRP model based on the J-measure criterion. The difference between the models based on the J-measure was statistically significant ($p < 0.05$). It has to be mentioned that the F-measure results for both models are almost the same. This showed that in terms of homogeneity, the models behaved the same; but considering the completeness property, the advantage of the Kmeans-3 model over the CRP model was highlighted.

In addition to the SOTA techniques, we utilized different parametric and non-parametric methods in our study. We utilized DBSCAN model, as a non-parametric algorithm, for inducing word senses. The model could not beat the CRP model as a baseline according to the J-measure results for both Persian and English. We further observed that the performance of the DBSCAN model was very similar to the MFS baseline since it had a high score for F-measure which means that this clustering algorithm ends up to one single cluster in most of the cases and only homogeneity was taken into consideration.

As mentioned, we used two modes in our experiments, SentContext and WinContext. To have a fair comparison between the modes, we ran the WinContext mode based on the Kmeans-3 model for both Persian and English to be able to compare the results with the SentContext mode of Kmeans-3 as one of the SOTA models.

According to the results, the WinContext mode of the Kmeans-3 model for both Persian and English had beaten the Kmeans-3 model in SentContext mode based on V-measure. The difference between the modes of the Kmeans-3 model was statistically significant ($p < 0.01$). The superiority of the Kmeans-3 model in WinContect mode was reflected in the J-measure. This result indicated that the surrounding words of the target word in the local context have a major impact on determining the meaning of the target word, and all of the words in the sentence are not effective. Comparing the results based on F-measure, the WinContext mode obtained a higher result than the SentContext for Persian; however, SentContext achieved better F-measure than the WinContext for English.

Comparing the proposed models of parametric clustering in either Win- or SentContext mode with the baselines indicated that none of the models had been able to beat the two naïve baselines: the 1S1C baseline based on V-measure, and the MFS baseline based on F-measure. Therefore, it was not possible to compare and to rank the models fairly. J-measure, however, filled the gap. According to the results of the proposed evaluation metric, i.e. J-measure, the proposed WSI models outperformed the naïve baselines. The score of the joint metric has made it possible to compare the proposed models with the SOTA models as well.

We further utilized two parametric algorithms to induce word senses. First, we used the divisive algorithm in SentContext and WinContext modes for both Persian and English. According to the V-measure results, the divisive algorithm had beaten the MFS baseline as well as CRF.

As can be seen, the WinContext mode of the divisive algorithm for both Persian and English obtained a higher result than the SentContext mode. This showed that the divisive algorithm required a narrow context to determine the meaning of words. The differences between the modes were statistically significant ($p < 0.05$). It had to be mentioned that neither of the modes of the divisive clustering algorithm for Persian were able to beat the respective mode of the Kmeans-3 model according to J-measure. While WinContext mode of the divisive clustering for English dataset had been able to beat the respective mode of the Kmeans-3 model based on V-measure which was also reflected in J-measure.

In addition to the divisive algorithm, we used the K-means algorithm enhanced with the silhouette score, thereafter called Kmeans-silhouette, for finding the best number of clusters in the two modes for both Persian and English. According to the results of V-measure, the WinContext mode of this algorithm for both datasets had beaten the SentContext mode. The difference between the modes of this clustering algorithm for the Persian data was statistically significant ($p < 0.05$) but not for the English dataset. Comparing this clustering algorithm to the Kmeans-3, as the SOTA baseline, it had to be mentioned that the WinContext mode of Kmeans-silhouette model for both datasets was able to beat the respective mode of the Kmeans-3 model according to V-measure with statistically significant difference ($p < 0.05$). This shows that the surrounding words in the local context are important for K-means clustering to induce word senses. The SentContext mode of the English data had beaten the SentContext mode of the Kmeans-3 model with statistically significant difference ($p < 0.05$), but not the Persian data, where a slightly poor performance of the SentContext mode was obtained. Comparing the Kmeans-silhouette model to the divisive algorithm for both modes of the two languages, the Kmeans-silhouette model had beaten the divisive clustering algorithm with statistically significant difference ($p < 0.05$).

We further ranked the models and found the best model which has a reasonable good performance based on J-measure. In general, the WinContext mode of the Kmeans-silhouette model for both Persian and English performed the best. This determined that the surrounding words in the context play a significant role in determining the meaning of the word and all of the words in the sentence do not play a major role. This achievement results in reduction of the computation time to produce words’ vectors and perform clustering.

6- Conclusion

In this paper, we studied the performance of various clustering algorithms, from parametric to non-parametric, to induce words' senses automatically. The algorithms were run by using Persian or English datasets. Furthermore, two modes, WinContext or SentContext, were used to build words' vectors. Finally, we utilized two evaluation criteria, namely V- and F-measure. There is always a trade-off between these metrics and a model evaluated with these metrics cannot beat a naïve baseline. Therefore, we contributed to propose J-measure as a harmonic mean of V- and F-measure to ease comparing the models. The results were compared with two basic baselines, 1S1C and MFS, and two SOTA models, CRP and Kmeans-3. By comparing the experimental results, we concluded that the parametric clustering algorithm performs better than the non-parametric clustering algorithm for inducing word senses. Among the parametric clustering algorithms, the Kmeans-silhouette clustering model in WinContext performed the best to induce senses of both Persian and English words. This result indicated that the surrounding words of the local context are highly effective in determining the meaning of words than other words in the sentence.

Devlin et al. [53] proposed a model for language representation known as the Bidirectional Encoder Representations from Transformers (BERT) model. This model is currently the SOTA model. One direction of this study as the future work is using the BERT embedding model for the WSI task and comparing the results with the Word2Vec-based embedding model.

References

[1] F. de Saussure, *Cours de linguistique générale*, C. Bally, A. Sechehaye, and A. Riedlinger, Eds. Lausanne, Paris: Payot, 1916.

[2] J. Lyons, *Language and Linguistics: An Introduction*. Cambridge, UK: Cambridge University Press, 1981.

[3] L. Wittgenstein, *Philosophical Investigations*. Oxford, UK: Blackwell Publishing Ltd, 1953.

[4] Z. S. Harris, "Distributional structure," *Word*, vol. 23, no. 10, pp. 146–162, 1954.

[5] J. R. Firth, "A synopsis of linguistic theory 1930-1955," *Studies in Linguistic Analysis* (special volume of the Philological Society), pp. 1–32, 1957.

[6] G. A. Miller and W. G. Charles, "Contextual correlates of semantic similarity," *Language and Cognitive Processes*, vol. 6, no. 1, pp. 1–28, 1991.

[7] Y. Peirsman and D. Geeraerts, "Predicting strong associations on the basis of corpus data," in *Proceedings of the 12th Conference of the European Chapter of the Association for Computational Linguistics*. Stroudsburg, PA, USA: Association for Computational Linguistics, 2009, pp. 648–656.

[8] T. K. Landauer and S. T. Dumais, "A solution to Plato's problem: The latent semantic analysis theory of acquisition, induction, and representation of knowledge," *Psychological Review*, vol. 104, no. 2, pp. 211–240, 1997.

[9] M. Sahlgren, *The Word-space Model: Using Distributional Analysis to Represent Syntagmatic and Paradigmatic Relations Between Words in High-dimensional Vector Spaces*. Ph.D. dissertation, Stockholm University, Stockholm, Sweden, 2006.

[10] Z. S. Harris, *A Theory of Language and Information: A Mathematical Approach*. Oxford, England: Oxford University Press, 1991.

[11] D. Lin, "Automatic retrieval and clustering of similar words," in *Proceedings of the 17th international conference on Computational linguistics*. Morristown, NJ, USA: Association for Computational Linguistics, 1998, pp. 768–774.

[12] S. Padó and M. Lapata, "Dependency-based construction of semantic space models," *Computational Linguistics*, vol. 33, no. 2, pp. 161–199, June 2007.

[13] O. Levy and Y. Goldberg, "Dependency-based word embeddings," in *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics*, vol. 2. Baltimore, Maryland: Association for Computational Linguistics, June 2014, pp. 302–308.

[14] K. M. Hermann and P. Blunsom, "The role of syntax in vector space models of compositional semantics," in *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics*, vol. 1, Sofia, Bulgaria, 2013, pp. 894–904.

[15] L. Song, Z. Wang, H. Mi, and D. Gildea, "Sense embedding learning for word sense induction," in *Proceedings of the 5th Joint Conference on Lexical and Computational Semantics. The *SEM 2016 Organizing Committee*, 2016, pp. 85–90.

[16] D. M. Blei, A. Ng, and M. Jordan, "Latent dirichlet allocation," *Journal of Machine Learning Research*, vol. 3, pp. 993–1022, 2003.

[17] D. Jurafsky and J. H. Martin, *Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition*, 2020, https://web.stanford.edu/~jurafsky/slp3/ed3book.pdf.

[18] S. K. Jauhar, C. Dyer, and E. Hovy, "Ontologically grounded multi-sense representation learning for semantic vector space models," in *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. Denver, Colorado: Association for Computational Linguistics, May 2015, pp. 683–693.

[19] S. Rothe and H. Schütze, "AutoExtend: Extending word embeddings to embeddings for synsets and lexemes," in *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing*, vol. 1. Beijing, China: Association for Computational Linguistics, July 2015, pp. 1793–1803.

[20] S. Ramprasad and J. Maddox, "CoKE: Word sense induction using contextualized knowledge embeddings," in *Proceedings of the AAAI 2019 Spring Symposium on Combining Machine Learning with Knowledge Engineering*, 2019.

[21] J. Pennington, R. Socher, and C. D. Manning, "GloVe: Global Vectors for word representation," in *Proceedings of*

the 2014 Conference on Empirical Methods in Natural Language Processing, vol. 14, 2014, pp. 1532–1543.

[22] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean, “Distributed representations of words and phrases and their compositionality,” in *Proceedings of the 26th International Conference on Neural Information Processing Systems*, C. J. C. Burges, L. Bottou, M. Welling, Z. Ghahramani, and K. Q. Weinberger, Eds. Curran Associates, Inc., 2013, pp. 3111–3119.

[23] J. MacQueen, “Some methods for classification and analysis of multivariate observations,” in *Proceedings of the 5th Berkeley Symposium on Mathematical Statistics and Probability*, vol. 1. Berkeley, California: University of California Press, 1967, pp. 281–297.

[24] E. Huang, R. Socher, C. D. Manning, and A. Ng, “Improving word representations via global context and multiple word prototypes,” in *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics*, vol. 1. Jeju Island, Korea: Association for Computational Linguistics, July 2012, pp. 873–882.

[25] M. Ghayoomi, “Density measure in context clustering for distributional semantics of word sense induction,” *Journal of Information Systems and Telecommunication*, vol. 8, no. 1, pp. 15–24, 2020.

[26] P. J. Rousseeuw, “Silhouettes: A graphical aid to the interpretation and validation of cluster analysis,” *Journal of Computational and Applied Mathematics*, vol. 20, no. 1, pp. 53–65, November 1987.

[27] D. M. Blei, M. I. Jordan, T. L. Griffiths, and J. B. Tenenbaum, “Hierarchical topic models and the nested Chinese restaurant process,” in *Proceedings of the 16th International Conference on Neural Information Processing Systems*. MIT Press, 2003, pp. 17–24.

[28] M. Ester, H.-P. Kriegel, J. Sander, and X. Xu, “A density-based algorithm for discovering clusters in large spatial databases with noise,” in *Proceedings of the 2nd International Conference on Knowledge Discovery and Data Mining*, E. Simoudis, J. Han, and U. M. Fayyad, Eds. AAAI Press, 1996, pp. 226–231.

[29] A. Neelakantan, J. Shankar, A. Passos, and A. McCallum, “Efficient nonparametric estimation of multiple embeddings per word in vector space,” in *Processing of the Conference on Empirical Methods in Natural Language*. Doha, Qatar: Association for Computational Linguistics, 2014.

[30] J. Li and D. Jurafsky, “Do multi-sense embeddings improve natural language understanding?” in *Proceedings of the Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, 2015, pp. 1722–1732.

[31] J. Wang, M. Bansal, K. Gimpel, B. D. Ziebart, and C. T. Yu, “A sensetopic model for word sense induction with unsupervised data enrichment,” *Transactions of the Association for Computational Linguistics*, vol. 3, pp. 59–71, 2015.

[32] A. Amrami and Y. Goldberg, “Word sense induction with neural biLM and symmetric patterns,” in *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*. Brussels, Belgium: Association for Computational Linguistics, 2018, pp. 4860–4867.

[33] M. Peters, M. Neumann, M. Iyyer, M. Gardner, C. Clark, K. Lee, and L. Zettlemoyer, “Deep contextualized word representations,” in *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, vol. 1. New Orleans, Louisiana: Association for Computational Linguistics, 2018, pp. 2227–2237.

[34] D. Alagić, J. Šnajder, and S. Padó, “Leveraging lexical substitutes for unsupervised word sense induction,” in *Proceedings of the 32nd Conference of the Association for the Advancement of Artificial Intelligence*. New Orleans, LA, 2018.

[35] E. A. Corrêa and D. R. Amancio, “Word sense induction using word embeddings and community detection in complex networks,” *Physica A: Statistical Mechanics and its Applications*, vol. 523, pp. 180–190, 2019.

[36] B. Perozzi, R. Al-Rfou’, V. Kulkarni, and S. Skiena, “Inducing language networks from continuous space word representations,” in *Complex Networks*, P. Contucci, R. Menezes, A. Omicini, and J. Poncela-Casasnovas, Eds. Cham: Springer International Publishing, 2014, pp. 261–273.

[37] P. T. Tallo, *Using Sentence Embeddings for Word Sense Induction*. Master’s thesis, Electrical Engineering and Computer Science, University of Cincinnati, Ohio, USA, 2020.

[38] Q. Dong and Y. Wang, “Enhancing medical word sense inventories using word sense induction: A preliminary study,” in *Proceedings of the 6th International Workshop on Data Management and Analytics for Medicine and Healthcare, in conjunction with the 46th International Conference on Very Large Data Bases*, 2020, pp. 151–167.

[39] S. Arora, Y. Li, Y. Liang, T. Ma, and A. Risteski, “Linear algebraic structure of word senses, with applications to polysemy,” *Transactions of the Association for Computational Linguistics*, vol. 6, pp. 483–495, 2018.

[40] S. Manandhar, I. P. Klapaftis, D. Dligach, and S. S. Pradhan, “Semeval-2010 task 14: Word sense induction & disambiguation,” in *Proceedings of the 5th International Workshop on Semantic Evaluation*. Stroudsburg, PA, USA: Association for Computational Linguistics, 2010, pp. 63–68.

[41] G. M. Salton, A. Wong, and C.-S. Yang, “A vector space model for automatic indexing,” *Communications of the ACM*, vol. 18, no. 11, pp. 613–620, November 1975.

[42] M. Ghayoomi, “Finding the meaning of Persian words automatically using word embedding,” *Iranian Journal of Information Processing & Management*, vol. 35, no. 1, pp. 25–50, 2019.

[43] S. Assi, “Farsi linguistic database (FLDB),” *International Journal of Lexicography*, vol. 10, no. 3, p. 5, 1997.

[44] A. AleAhmad, H. Amiri, E. Darrudi, M. Rahgozar, and F. Oroumchian, “Hamshahri: A standard Persian text collection,” *Knowledge-Based Systems*, vol. 22, no. 5, pp. 382–387, 2009.

[45] M. Bijankhan, “naqše peykarehāye zabāni dar neveštane dasture zabān: moʻarrefiye yek narmafzāre rāyāneyi [“The role of corpora in writing a grammar: Introducing a software”],” *Journal of Linguistics*, vol. 19, no. 2, pp. 48–67, 2004.

[46] M. Bijankhan, J. Sheykhzadegan, M. Bahrani, and M. Ghayoomi, “Lessons from building a Persian written corpus: Peykare,” *Language Resources and Evaluation*, vol. 45, no. 2, pp. 143–164, 2011.

[47] C. Shaoul and C. Westbury, “The Westbury Lab Wikipedia Corpus,” 2010,

http://www.psych.ualberta.ca/~westburylab/downloads/westburylab.wikicorp.download.html.

[48] M. Shamsfard, H. S. Jafari, and M. Ilbeygi, "STeP-1: A set of fundamental tools for Persian text processing," in *Proceedings of the 7th International Conference on Language Resources and Evaluation*, N. Calzolari, K. Choukri, B. Maegaard, J. Mariani, J. Odijk, S. Piperidis, M. Rosner, and D. Tapias, Eds. Valletta, Malta: European Language Resources Association (ELRA), May 19–21 2010, pp. 859–865.

[49] C. J. V. Rijsbergen, *Information Retrieval*, 2nd ed. Newton, MA, USA: Butterworth-Heinemann, 1979.

[50] A. Rosenberg and J. Hirschberg, "V-measure: A conditional entropy-based external cluster evaluation measure," in *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*. Prague, Czech Republic: Association for Computational Linguistics, June 2007, pp. 410–420.

[51] B. E. Dom, *An Information-theoretic External Cluster-validity Measure*. IBM, Tech. Rep., 2001.

[52] M. Meilă, "Comparing clusterings – an information based distance," *Journal of Multivariate Analysis*, vol. 98, no. 5, pp. 873–895, May 2007.

[53] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," in *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies. Minneapolis, Minnesota: Association for Computational Linguistics*, 2019, pp. 4171–4186.

Masood Ghayoomi received his PhD degree in Computational Linguistics from Berlin Freie University, Berlin, Germany in 2014, and M.S. degree in Computational Linguistics from Nancy2 University, Nancy, France and Saarland University, Saarbrücken, Germany, in 2009. Currently he is a faculty member at the Institute for Humanities and Cultural Studies. His research interests include Computational Linguistics, Natural Language Processing, Machine Learning, Corpus Linguistics, Syntax and Lexical Semantics.