Research Article
Journal of Intelligent Systems: Theory and Applications 3(1) 2020: 1-6

Assessment of Mutation Susceptibility in DNA Sequences with Word Vectors

Alper Yilmaz [1]

[1]Yildiz Technical University, Faculty of Chemical and Metallurgical Engineering, Department of Bioengineering, Istanbul/ Turkey

Abstract

With the advent of natural language processing (NLP) techniques empowered with deep learning approaches, more detailed relationships between words have been unraveled. Word2Vec is quite robust in discovering contextual and semantic relationships. Genome being a long text, is subject to similar studies to unravel yet to be discovered relationships between DNA k-mers. Dna2vec applies Word2Vec approach to whole genome so that DNA k-mers are represented as vectors. The cosine similarity queries on DNA vectors reveal unusual relationships between DNA k-mers.

In this study, we examined DNA sequence based prediction of mutation susceptibility. Initially, we generated word vectors for human and mouse genome via dna2vec. On the other hand, we retrieved coordinates of common and all mutations from dbSNP. For each coordinate, we extracted 8 nucleotide k-mers intersecting mutations and results are aggregated such a way that number of mutations for each 8-mer has been tabulated. These results are incorporated with dna2vec cosine similarity data. Our results showed that for a given k-mer, k-mers with highest cosine similarity coincide with highest mutation count k-mer. In other words, the neighbor with the highest cosine similarity for a k-mer was also seen to be the neighbor overlapping the mutation count. As a result of our studies, human and mouse, dna2vec vs. mutation overlap is 80% and 70%, respectively. In conclusion, dna2vec and other word embedding approaches can be used to reveal mutation or variation characteristics of genomes without sequencing or experimental data, solely using the genome sequence itself. This might pave the way for understanding the underlying mechanism or dynamics of mutations in genomes.

Keywords: mutation, word2vec, dna2vec, k-mer, cosine-similarity.

DNA Dizilerinde Kelime Vektörleri ile Mutasyon Yatkınlığının Değerlendirilmesi

Öz

Derin öğrenme yaklaşımları ile güçlendirilen doğal dil işleme (NLP) tekniklerinin ortaya çıkmasıyla, kelimeler arasındaki daha ayrıntılı ilişkiler ortaya çıkarılmıştır. Bu açıdan word2vec yöntemi bağlamsal ve anlamsal ilişkileri keşfetme konusunda oldukça gelişmiştir. Uzun bir metin olan genom, DNA k-merleri arasındaki ilişkileri henüz keşfedememiş olan benzer çalışmalara tabidir. Dna2vec, DNA k-merlerinin vektör olarak gösterilmesi için tüm genoma word2vec yaklaşımını uygular. DNA vektörleri üzerindeki kosinüs benzerlik sorguları DNA k-merleri arasında olağandışı ilişkiler ortaya koymaktadır.

Bu çalışmada, mutasyon duyarlılığının DNA dizisi temelli tahmini incelendi. Başlangıçta, insan ve fare genomu için dna2vec yoluyla sözcük vektörleri üretildi. Diğer yandan, ortak ve tüm mutasyonların koordinatlarını dbSNP'den alındı. Her koordinat için, kesişen mutasyonlar halinde 8 nükleotidlik k-merler çıkarıldı ve sonuçlar toplandı. Bu sonuçlar dna2vec kosinüs benzerlik verileri ile birleştirildi. Sonuçlarımız, belirli bir k-mer için, en yüksek kosinüs benzerliğine sahip k-merlerin en yüksek mutasyon sayısına sahip k-merler ile çakıştığını göstermiştir. Başka bir deyişle, bir k-mer için kosinüs benzerliği en yüksek komşunun, mutasyon sayısıyla çakışan komşu olduğu da görülmüştür. Çalışmalarımız sonucunda insanda ve farede, k-mer kosinüs benzerliği ve mutasyon örtüşmesinin oranları sırasıyla %80 ve %70 olduğu görülmüştür. Bu çalışmalar sonucunda dna2vec ve diğer kelime gömme yaklaşımları, sadece genom dizisinin kendisi kullanılarak, dizileme veya deney verileri olmadan genomların mutasyon veya varyasyon

* Corresponding Author.
E-mail: alyilmaz@yildiz.edu.tr

Recieved : 17 Jan 2020
Revision : 22 Jan 2020
Accepted : 07 Feb 2020

özelliklerini ortaya çıkarmak için kullanılabilir olduğu gösterilmiştir. Bu durum, genomlardaki mutasyonların altında yatan mekanizmayı veya dinamikleri anlamanın yolunu açabilir.

Anahtar kelimeler: mutasyon, word2vec, dna2vec, k-mer, kosinüs benzerliği.

1. Introduction

Underlying problem of language modeling is that dimensionality. Most known and used technique to solve this problem is word2vec, word embedding method (Mikolov *et al.* 2013).

This method includes two models; Continuous Bag of Words (CBOW) and Skip Gram. These models converts word in the corpus into vectors. Purpose of these models is to determine contextually similar words. In line with this objective, they utilize "context window". All words within *n* units from a target word in context window belong to context of that target word. This architecture based on context window can predict either target or the context. A shallow neural network is used for training of words in these architectures. Output of this neural network gives the semantic or syntactic relation among words. Word vectors can be added or subtracted in order to reveal or extend relationships between words, as shown below:

$$vec(king) - vec(man) + vec(woman) \approx vec(queen)$$
(Mikolov *et al.* 2013)

Moreover, word2vec can capture word relations such as past-tense, ownership, language spoken in the country or the capital of country information (Levy and Goldberg, 2014). For many machine learning algorithms, an input of fixed-length vector is required. Word2vec generates fixed-length embedding for words which can be later used as input to subsequent machine learning applications. Thus, word2vec has been used in studies in different areas such as document representations for classification of documents and sentiment analysis (Le and Mikolov, 2014; Kusner *et al*. 2015; Chen, 2017), network detection to predict the link between the edges and nodes (Perozzi *et al.* 2014; Chen and Lawrence Zitnick, 2015), image caption generation (Pedersoli *et al.* 2017; Kiros *et al.* 2014), identification of public user tendency such as political preference or occupational class (Preoţiuc-Pietro *et al.* 2015; Yang *et al.* 2018; Preoţiuc-Pietro *et al.* 2017), limited contextual information transformation (Dos Santos and Gatti, 2014; De Boom *et al.* 2015), image annotation to provide computer vision and pattern recognition (Uricchio *et al.* 2017), using biomedical literature to predict drug interactions and capturing medical semantic similarities (Wang *et al.* 2018; Zhao *et al.* 2016), emotion detection (Abdul-Mageed and Ungar, 2017; Eisner *et al.* 2016), improving word similarity detection (Schwartz *et al.* 2015; Faruqui *et al.* 2016) and polysemy detection (Arora et al. 2018; Jauhar *et al.* 2015) and removing gender bias in embeddings (Bolukbasi *et al.* 2016). The applications of word2vec can extend to domains far from computer science such as urban planning (Yao *et al.* 2017). Word2vec has been calculated for Turkic languages (Akın and Akın, 2007) as well.

The human genome consisting of 3 billion letters can be considered as a long string and thus subject to word embedding studies. This long string can be divided into constant or variable-length k-mers to produce words. In order to reveal semantic and syntactic relationships between the k-mers in human genome, the dna2vec study was carried out by using word2vec method (Ng, 2017).

Dna2Vec method, using variable-length k-mers training in two-layer neural network, demonstrated correlation between Needleman-Wunsch alignment score and dna2vec cosine similarity of k-mers. Needleman-Wunsch algorithm (Needleman et al. 1970) measures sequence similarity between to k-mers by global alignment approach. In addition, summing dna2vec embedding vectors for two k-mers is equivalent to concatenated k-mer. For example;

$$vec(\text{AAC}) + vec(\text{TCT}) \approx vec(\text{AACTCT})$$
or
$$vec(\text{AAC}) + vec(\text{TCT}) \approx vec(\text{TCTAAC})$$

In this study, we investigated relation between mutation dynamic and vector embeddings. Mutation is change of a letter in the genome, usually causing adverse effects in gene products resulting in diseases. There are no algorithms for predicting mutations and only technique to detect mutations is experimentally sequencing the DNA. In order find a technique to predict mutations, millions of known mutations and their affected words were compared with dna2vec vector embeddings. Our results show there is correlation with vector embedding and mutation count per k-mer suggesting vector embeddings have potential to predict mutations in genome.

2. Methods

2.1. Data Retrieval and dna2vec Script

Human genome hg38 and mouse genome mm10 downloaded from NCBI, common SNPs for human and mouse are downloaded from dbSNP (dbSNP link). Human dna2vec model was retrieved from dna2vec repository (GitHub link). Python scripts from dna2vec study are used to generate human and mouse dna2vec models (Ng, 2017). Human genome results were compared with results from mouse genome in order to show that our findings hold true for any organism. Also, mouse genome has comprehensive mutation data available.

2.2. Counting mutation per k-mer

We collected sliding window 8-mer around each mutation for nearly 35 million mutations in human and 73 million mutations in mouse. Then we counted occurrence of all k-mers and mutation pairs. Each pair contain coordinate and type of mutation in the k-mer.

2.3. Integration mutation and dna2vec data

For all 65,536 8-mers we generated 24 neighbors with hamming one distance for each coordinate of the k-mer. For a particular k-mer and its neighbor, we extracted mutation count and dna2vec cosine similarity.

2.4. Calculating correlation between mutations and cosine similarity

Using the data generated in previous step for each mutation in every coordinate for all the words we compared number of mutations and cosine similarity and prepared the confusion matrix.

3. Results

3.1. dna2vec Properties

As previously shown addition and subtraction vector embeddings correspond to concatenation and subtraction events in k-mers. Concatenation of a k-mer to another k-mer from right or from left generates two strings with comparable embeddings. (Figure 1A). Similarly, subtracting and then concatenating is applicable to embeddings (Figure 1B).

3.2 Mutation counts

We applied sliding window technique to the genome and we counted each mutation in every coordinate for all 8-mers. Figure 2 illustrates technique and genome-wide results for k-mers of interest. Mutation counts for both human and mouse genomes were calculated.

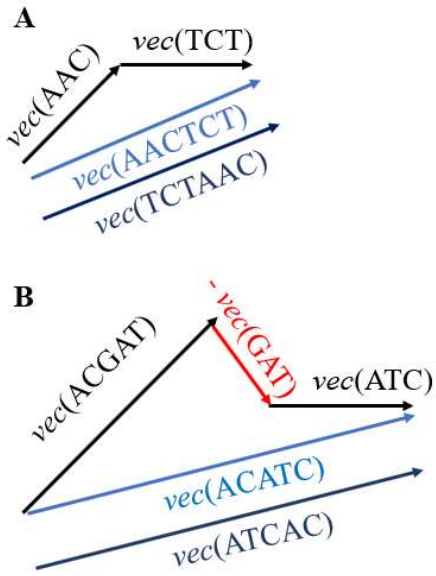
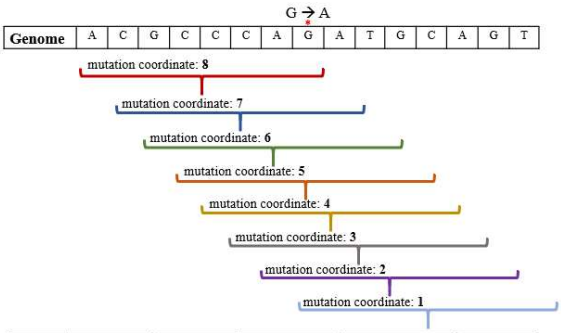


Figure 1. Schematic representation of concatenation and subtraction event in context of dna2vec embeddings.



Results	Genome	Coordinate	Variation Genome	Nucleotide Exchange	Mutation Count
	ACGCCCAG	8	ACGCCCAA	G → A	236
	CGCCCAGA	7	CGCCCAAA	G → A	45
	GCCCAGAT	6	GCCCAAAT	G → A	202
	CCCAGATG	5	CCCAAATG	G → A	304
	CCAGATGC	4	CCAAATGC	G → A	174
	CAGATGCA	3	CACATGCA	G → A	280
	AGATGCAG	2	CAAATGCA	G → A	318
	GATGCAGT	1	CAAATGCA	G → A	329

Figure 2: Mutation count for each sliding window k-mer overlapping a mutation. Top part depicts k-mers overlapping a mutation and coordinate of mutation in each k-mer. Bottom is generated after extracting and counting k-mers over all mutations.

3.3 Mutation counts vs. vector similarity

In order to quantify relationship between vector embeddings and mutation counts we used ranking of mutation counts and cosine similarity per coordinate. So, for a k-mer, all possible mutations at a particular coordinate were ranked according to mutation counts (rank 1 is the highest count). Similarly, for a particular

k-mer cosine similarity of all possible words generated by changing one letter (i.e. mutation) has been calculated and ranked (highest similarity is ranked 1). Only after, we calculated correlation between mutation occurrence and cosine similarity. We found that 80% of human mutations and 70% of mouse mutations have overlapping top rank with cosine similarity. Table 1 lists k-mers carrying mutations with highest mutation count and highest cosine similarity.

Table 1: Mutation counts and cosine similarity values. For the k-mer TGAGCACT, each possible neighbor with hamming distance one has been listed and for each neighbor, mutation count and cosine similarity has been calculated. The numbers in parenthesis indicate ranking of mutation within given coordinate.

	Conversion	Mutation Count	Cosine Similarity
T	T -> A	50 (3)	0.6481 (1)
	T -> G	90 (2)	0.6410 (2)
	T -> C	277 (1)	0.5936 (3)
G	G -> T	52 (3)	0.6190 (1)
	G -> A	343 (1)	0.5888 (2)
	G -> C	106 (2)	0.5191 (3)
A	A -> G	162 (1)	0.5550 (1)
	A -> C	45 (2)	0.5400 (2)
	A -> T	33 (3)	0.5215 (3)
G	G -> A	452 (1)	0.5696 (1)
	G -> T	69 (3)	0.5490 (2)
	G -> C	82 (2)	0.5370 (3)
C	C -> T	301 (1)	0.6268 (1)
	C -> G	66 (3)	0.5402 (2)
	C -> A	109 (2)	0.5014 (3)
A	A -> G	293 (1)	0.6137 (1)
	A -> T	69 (3)	0.6101 (2)
	A -> C	127 (2)	0.5726 (3)
C	C -> T	379 (1)	0.6706 (1)
	C -> G	135 (2)	0.6256 (2)
	C -> A	75 (3)	0.6208 (3)
T	C -> T	283 (1)	0.7887 (1)
	C -> G	68 (2)	0.7461 (2)
	C -> A	50 (3)	0.7352 (3)

Figure 3 illustrates overlap of mutation count and cosine similarity rankings per coordinate for a particular 8-mer. For the first coordinate of the k-mer TGAGCACT, T→ C mutation has the highest ranking compared to T→ G and T→ A mutations. On the other hand, word generated by T→A change has the highest cosine similarity ranking when compared to words generated by T→ G and T→ C changes. When all coordinates are examined, at 6 out of 8 coordinates, top ranks for mutation and cosine similarity overlaps.

Figure 3: Relation between the mutation count and cosine similarity. Letters above and below indicate ranking of mutation counts and cosine similarity, respectively, in decreasing order.

3.4 Confusion Matrix

We prepared the confusion matrix for both human and mouse (Table 2, Table 3) in order to assess predictive power of cosine similarity to predict experimental mutations. Ranking order per coordinate has been considered as features to be compared. The confusion matrix show that accuracy values are 0.7974 and 0.8322 for mouse and human, respectively.

Table 2: Confusion matrix for mouse

	Reference (Mutation Rank)		
Prediction (Cosine Similarity)	**1**	**2**	**3**
1	7174	123	2
2	1182	328	33
3	428	165	105

Summary of Statistics			
	Class: 1	**Class: 2**	**Class: 3**
Sensitivity	0.8167	0.5325	0.7500
Specificity	0.8347	0.8639	0.9369
Accuracy	0.7974		

4. Conclusions

dna2vec has been previously shown to exhibit concatenation and similarity properties for k-mer vectors. Our study showed that dna2vec can also reveal mutation susceptibility of k-mers. Our findings suggest that by using dna2vec, mutation susceptibility of k-mers can be predicted by processing the genome sequence alone even if no experimental result is available.

Table 3: Confusion matrix for human

	Reference (Mutation Rank)		
Prediction (Cosine Similarity)	**1**	**2**	**3**
1	2073	16	0
2	307	33	4
3	64	35	6

Summary of Statistics			
	Class: 1	**Class: 2**	**Class: 3**
Sensitivity	0.8482	0.39286	0.600000
Specificity	0.8298	0.87327	0.960839
Accuracy	0.8322		

This indicates that mutation susceptibility information is inherent and embedded within genome sequence and dna2vec can unlock it. With our approach, genomes of numerous organisms can be analyzed for mutation susceptibility, paving the way to comparison mutation dynamics or mechanisms among different organism without experimental mutation data. Moreover, dna2vec has potential to unveil the traces of yet undiscovered biological mechanisms effecting k-mers in genome. Finally, dna2vec opens up the genome to various applications of artificial intelligence by converting DNA sequences into fixed length, transformed embeddings.

Acknowledgment

Code and data used in this study are available at https://github.com/alperyilmaz/dna2vec_snp

References

Abdul-Mageed, M., & Ungar, L. (2017, July). Emonet: Fine-grained emotion detection with gated recurrent neural networks. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)* (pp. 718-728).

Akın, A. A., & Akın, M. D. (2007). Zemberek, an open source nlp framework for turkic languages. *Structure*, *10*, 1-5.

Arora, S., Li, Y., Liang, Y., Ma, T., & Risteski, A. (2018). Linear algebraic structure of word senses, with applications to polysemy. *Transactions of the Association for Computational Linguistics*, *6*, 483-495.

Bolukbasi, T., Chang, K. W., Zou, J. Y., Saligrama, V., & Kalai, A. T. (2016). Man is to computer programmer as woman is to homemaker? debiasing word embeddings. In *Advances in neural information processing systems* (pp. 4349-4357).

Chen, M. (2017). Efficient vector representation for documents through corruption. *arXiv preprint arXiv:1707.02377*.

Chen, X., & Lawrence Zitnick, C. (2015). Mind's eye: A recurrent visual representation for image caption generation. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 2422-2431).

De Boom, C., Van Canneyt, S., Bohez, S., Demeester, T., & Dhoedt, B. (2015, November). Learning semantic similarity for very short texts. In *2015 ieee international conference on data mining workshop (icdmw)* (pp. 1229-1234). IEEE.

Dos Santos, C., & Gatti, M. (2014, August). Deep convolutional neural networks for sentiment analysis of short texts. In *Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers* (pp. 69-78).

Eisner, B., Rocktäschel, T., Augenstein, I., Bošnjak, M., & Riedel, S. (2016). emoji2vec: Learning emoji representations from their description. *arXiv preprint arXiv:1609.08359*.

Faruqui, M., Tsvetkov, Y., Rastogi, P., & Dyer, C. (2016). Problems with evaluation of word embeddings using word similarity tasks. *arXiv preprint arXiv:1605.02276*.

Gladkova, A., Drozd, A., & Matsuoka, S. (2016, June). Analogy-based detection of morphological and semantic relations with word embeddings: what works and what doesn't. In *Proceedings of the NAACL Student Research Workshop* (pp. 8-15).

Jauhar, S. K., Dyer, C., & Hovy, E. (2015). 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* (pp. 683-693).

Kiros, R., Salakhutdinov, R., & Zemel, R. S. (2014). Unifying visual-semantic embeddings with multimodal neural language models. *arXiv preprint arXiv:1411.2539*.

Kusner, M., Sun, Y., Kolkin, N., & Weinberger, K. (2015, June). From word embeddings to document distances. In *International conference on machine learning* (pp. 957-966).

Le, Q., & Mikolov, T. (2014, January). Distributed representations of sentences and documents. In *International conference on machine learning* (pp. 1188-1196).

Levy, O., & Goldberg, Y. (2014, June). Linguistic regularities in sparse and explicit word representations. In Proceedings of the eighteenth conference on computational natural language learning (pp. 171-180).

Mikolov, T., Chen, K., Corrado, G., & Dean, J. (2013). Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*.

Mikolov, T., Yih, W. T., & Zweig, G. (2013, June). Linguistic regularities in continuous space word representations. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies* (pp. 746-751).

Needleman, S. B., & Wunsch, C. D. (1970). *A general method applicable to the search for similarities in the amino acid sequence of two proteins. Journal of Molecular Biology, 48(3), 443–453.* doi:10.1016/0022-2836(70)90057-4

Ng, P. (2017). dna2vec: Consistent vector representations of variable-length k-mers. arXiv preprint arXiv:1701.06279.
45

Pedersoli, M., Lucas, T., Schmid, C., & Verbeek, J. (2017). Areas of attention for image captioning. In *Proceedings of the IEEE International Conference on Computer Vision* (pp. 1242-1250).

Perozzi, B., Al-Rfou, R., & Skiena, S. (2014, August). Deepwalk: Online learning of social representations. In *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining* (pp. 701-710). ACM.

Preoţiuc-Pietro, D., Lampos, V., & Aletras, N. (2015, July). An analysis of the user occupational class through Twitter content. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)* (pp. 1754-1764).

Preoţiuc-Pietro, D., Liu, Y., Hopkins, D., & Ungar, L. (2017, July). Beyond binary labels: political ideology prediction of twitter users. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)* (pp. 729-740).

Schwartz, R., Reichart, R., & Rappoport, A. (2015, July). Symmetric pattern based word embeddings for improved word similarity prediction. In *Proceedings of the nineteenth conference on computational natural language learning* (pp. 258-267).

Sienčnik, S. K. (2015, May). Adapting word2vec to named entity recognition. In *Proceedings of the 20th nordic conference of computational linguistics, nodalida 2015, may 11-13, 2015, vilnius, lithuania* (No. 109, pp. 239-243). Linköping University Electronic Press.

Uricchio, T., Ballan, L., Seidenari, L., & Del Bimbo, A. (2017). Automatic image annotation via label transfer in the semantic space. *Pattern Recognition*, *71*, 144-157.

Wang, Y., Liu, S., Afzal, N., Rastegar-Mojarad, M., Wang, L., Shen, F., ... & Liu, H. (2018). A comparison of word embeddings for the biomedical natural language processing. *Journal of biomedical informatics*, *87*, 12-20.

Yang, X., Macdonald, C., & Ounis, I. (2018). Using word embeddings in twitter election classification. *Information Retrieval Journal*, *21*(2-3), 183-207.

Yao, Y., Li, X., Liu, X., Liu, P., Liang, Z., Zhang, J., & Mai, K. (2017). Sensing spatial distribution of urban land use by integrating points-of-interest and Google Word2Vec model. *International Journal of Geographical Information Science*, *31*(4), 825-848.

Zhao, Z., Yang, Z., Luo, L., Lin, H., & Wang, J. (2016). Drug drug interaction extraction from biomedical literature using syntax convolutional neural network. *Bioinformatics*, *32*(22), 3444-3453.