Word2Vec – שיטה של “word embedding” – ייצוג של מילים/ביטויים באמצעות וקטור של מספרים ממשיים. מטרת השיטה היא לייצג את המילים בצורה בעלת משמעות תחבירית/סמנטית מה שלא קיים בשיטה של “one hot”. השיטה מסתמכת על תאוריה בבלשנות לפיה מילים המופיעות לעיתים תכופות באותו הקשר הן בעלות משמעות סמנטית משותפת. מדובר ברשת deep learning המבצעת embedding כאשר הכניסה לרשת היא וקטורי "one hot" של המילים, באמצע ישנה שכבה אחת של מטריצה בגודל של מספר המילים במילון x מימד הווקטור לייצוג – המטריצה היא lookup table ובפועל היא כוללת את המשקלים של וקטורי הייצוג הנלמדים (מאותחלת לפי פילוג רנדומלי ומשתנה בתהליך הלמידה), כך שכל שורה היא הווקטור המתאים למילה, ובסוף קיימת פונקציית מחיר (בה נדון בהמשך).

קודם כל יוצרים מתוך הטקסט dataset של מילים ומילות ההקשר בו מופיעות, כלומר מגדירים "חלון" סביב כל מילה ("מילת מטרה") המכיל את "מילות ההקשר" - המילים הסמוכות למילה (למשל מספר מילים לפני ו/או אחרי המילה). למשל עבור המשפט: “the quick brown fox jumped over the lazy dog” וחלון דו-צדדי בגודל 1 נקבל dataset לדוגמא:

(

לשיטה יש שתי גרסאות: (continuous bag of words) ו- Skip-Gram. הראשונה מנסה ללמוד מתוך מילות ההקשר את המילה (לפי הדוגמה לעיל מתוך את ), והשנייה מנסה למצוא את מילות ההקשר בחלון מתוך מילת המטרה (מתוך את ).

כעת מגדירים הסתברות של מילת המטרה מתוך מילות ההקשר (או להפך) על ידי הפעלת פונקציית softmax על המכפלה הסקלרית בין הווקטורים המייצגים של המילים. כך, פונקציית הסתברות הכוללת של ה-dataset מתקבלת על ידי מכפלת ההסתברויות הללו ואותה אנו מנסים להביא למקסימום (או באופן שקול את פונקציית ה-“log-likelihood”). כך מגדירים "פונקציית מחיר" על ידי average negative log-likelihood, כלומר:  
כאשר זה מספר המילים ב-dataset, הינו גודל החלון (כאן דו-צדדי). משתמשים באלגוריתם אופטימיזציה (למשל SGD – stochastic gradient descent) כדי למזער את פונקציית המחיר, וכך משתנים המשקלים של וקטורי הייצוג. לצורך חישוב , משתמשים בשיטה זו בשני וקטורי ייצוג עבור כל מילה. משתמשים בוקטור כאשר היא מילת וב- כאשר היא מילת . לדוגמא, אם משתמשים בשיטת Skip-Gram, אזי בהינתן ש- מילת ו- מילת , אזי חישוב ההסתברות נראה כך:

ניסיונית התברר כי האופטימיזציה של מזעור פונקציית המחיר תהיה קלה יותר כאשר משתמשים בשני ייצוגים לכל מילה – פעם אחת בתור מילת הקשר ופעם בתור מילת מטרה, ולבסוף ליצירת ה- הסופי לוקחים ממוצע חשבוני של הווקטורים עבור כל מילה .

חישוב הגרדיאנט של ה- לפי כל וקטורי ה- הינו יקר ולכן משתמשים בשיטה של כך שבמקום לחשב גרדיאנט לפי כל המילים במילון מנסים להבחין בין מילות הקשר אמתיות לבין מילות "רעש" רנדומליות, כלומר להגדיל את ההסתברות של מילת המטרה מתוך מילת הקשר אמתית ולהקטין את ההסתברות של מילת המטרה מתוך מילות רנדומליות.

בנוסף, בשיטה זו מילים שחוזרות על עצמן הרבה בלי לתרום מידע (למשל מילים כמו המילה “the”) עלולות להשפיע על החישוב. בשביל להימנע מכך, בודקים עבור כל מילה את התכיפות שלה בטקסט ועושים דגימה לפי התכיפות של המילה בכל קטע הטקסט, כלומר ככל שהתכיפות של המילה בטקסט גבוהה יותר כך ההסתברות להוריד מה-dataset צמדים של (context,target) בהם מופיעה המילה הינה גדולה יותר.

דוגמת ריצה של Word2Vec: