Intro:

In natural language processing, there is a high demand for word representations that can accurately categorize the complex syntactic structures and semantic definitions of words or phrases. Word2Vec builds off of the work of the effective Skip-gram model and introduces several optimizations and improvements that makes it one of the most competitive word vector representations today. GloVe takes a different approach, instead constructing a ____ based on the properties that enable linear arithmetic on word vectors in skip-grams. However, both ultimately provide state of the art word vector representations that maintain the property of providing meaningful arithmetic operations.

Word2Vec:

The way Word2Vec works is that it essentially improves the existing skip-gram model by modifying the algorithm to train significantly faster, improve non-frequent word accuracy, and support phrases. The skip-gram model essentially seeks to optimize the sum of the logarithm of softmax of the training word and each of the words that appear in the same training sample. In other words, words that appear in similar contexts will be closer together in the vector space. This model is very powerful because of its accuracy and ability to perform rational arithmetic operations. For example, the operation Germany + Capitol might give you Berlin. The problem with this algorithm, however, is that it must iterate over the entire vocabulary many times while training. This is not feasible if the user wishes to construct a model with a relatively comprehensive coverage of any language's vocabulary.

In order to combat this, Word2Vec proposed 3 potential solutions: the Hierarchical Softmax, Negative Sampling, and Subsampling of Frequent Words. Hierarchical Softmax approximates the softmax function by taking a subset of relevant words from the vocabulary based off of a tree. Negative Sampling essentially compares words to randomly selected non-relevant samples and tries to maximize words that achieve higher relevancy than the noise rather than the entire vocabulary. Frequent Word Subsampling heuristically eliminates words with occurrence higher than a certain threshold, as they are usually stopwords that encode very little value and incur high computational costs. Results suggest that a combination of Negative Sampling and Frequent Word Subsampling are the fastest and have the highest semantic accuracy. Finally, Word2Vec detects n-gram phrases by looking for word pairs that have significantly higher appearance rates together as that phrase than they do otherwise. The results suggest that phrase detection works but requires a very large dataset to have good accuracy.

Overall, Word2Vec maintains the accuracy and nice linear arithmetic properties of skip-grams while providing significant performance improvements and extending support to n-gram phrases. The resulting technique is a very powerful tool for natural language processing.

GloVe:

GloVe similarly attempts to preserve the arithmetic operations that skip-gram provides but takes an entirely different approach to training. It first boils down the properties that enable reasonable arithmetic operations on word vectors, then constructs a model that they believe to be optimal based on these requirements.

The GloVe paper posits that the main property that enables linear arithmetic is the ratio of co-occurrences of words. This fundamental analysis leads GloVe to the log-bilinear model. In essence, they try to set the value of the dot product between two words to the logarithm of the co-occurrence of those two words. In order to derive these values, GloVe models the problem as a weighted least squares regression problem and attempts to minimize the difference of the dot product between 2 words and the logarithm of that word’s co-occurrence.

Conclusion:

Ultimately, both models provide high-accuracy, efficient and intuitive vector models that support arithmetic operations. Performance wise, GloVe achieves a slightly more accurate result at 42 billion words than a 100 billion word Word2Vec, and generally outperforms it when both are trained on 6 billion words. However, GloVe notably does not support phrase detection, as it is a unigram model, and may suffer in accuracy when users are looking for a specific phrase. The results suggest that the 2 models have very similar accuracies when provided large corpuses, so the decision to use which model likely depends on the context. If more computing power is available, it may be more beneficial to use Word2Vec, whereas GloVe is likely better for more lightweight applications where phrase detection capabilities should be sacrificed for significantly better performance with smaller datasets.

Sources:

Word2Vec paper:

https://proceedings.neurips.cc/paper/2013/file/9aa42b31882ec039965f3c4923ce901b-Paper.pdf

GloVe paper: https://nlp.stanford.edu/pubs/glove.pdf

GloVe website: https://nlp.stanford.edu/projects/glove/