FastText

The best configuration for Logistic Regression on FastText embeddings was -
C: 10000.0, class weight: None, dual: False, fit intercept: False, intercept scaling: 1, l1 ratio: 0.0, max iter: 100, multi class: auto, n jobs: None, penalty: l2, random state: None, solver: liblinear, tol: 0.0001, verbose: 0, warm start: False
It yielded the best score of 0.8 on validation set and further 0.765 on the test set.

TfidfVectorizer

The best configuration for Logistic Regression on TfidfVectorizer embeddings was -
C: 1.0, penalty: l2, solver: saga, max iter: 10000, class weight: None
It yielded the best accuracy score of 0.82 on validation set and further 0.815 on the test set.

miniLM and mpnet

For miniLM, the best accuracy score on validation set was 0.92, and for mpnet - 0.96. The score on test sets were 0.9 and 0.95 respectively. Both had the best configuration as -
C: 0.1, penalty: l2, solver: saga
I chose all-mpnet-base-v2 as it was the top in the list.

Stanza

Counting neutrals as negative result in accuracy of 0.97, and as positive - 0.98 using Stanza on validation set. Using that shift from neutral to positive, we get the accuracy of 0.92 on test set.

Some notes

- FastText has shown lower accuracy results and it is probably due to being pretrained on other set, whereas TF-IDF is based on the vocabulary of our own dataset.
- For the same reason as above FastText might have shown a more realistic evaluation or there was underfitting.
- Transformer based models have shown much better results, which is probably due to the ability to recognize more complex semantic relationships.