NLP with Disaster Tweets - Model Comparison

Michael Swindle

1 Overview

Kaggle provides 10 000 labelled tweets and asks: *Is this tweet about a real disaster?* Three pipelines scale from sparse TF-IDF counts to dense Word2Vec embeddings.

---

2 Pipelines

Version	Feature set	Key steps
v7	TF-IDF	Lower-case, punctuation strip, stop-word removal, XGBoost
v8	Clean TF-IDF	Adds UK→US spelling map, extra noise removal
v9	Word2Vec	100-dim CBOW vectors averaged per tweet, XGBoost

---

3 Results – Internal vs Kaggle

```
# internal split
internal <- tibble(
  Pipeline = c("v7 TF-IDF", "v8 Clean TF-IDF", "v9 Word2Vec"),
  Internal = c(run_v7(train_master, test_master),
               run_v8(train_master, test_master),
```

```r
                run_v9(train_master, test_master))
)

# hard-coded leaderboard scores
kaggle <- tibble(
  Pipeline = c("v7 TF-IDF", "v8 Clean TF-IDF", "v9 Word2Vec"),
  Kaggle   = c(0.56114, 0.57370, 0.76953)
)

results <- internal %>% left_join(kaggle, by = "Pipeline")

# table
knitr::kable(results, digits = 3,
             col.names = c("Pipeline", "Internal F1", "Kaggle F1"))
```

Pipeline	Internal F1	Kaggle F1
v7 TF-IDF	0.822	0.561
v8 Clean TF-IDF	0.824	0.574
v9 Word2Vec	0.786	0.770

Internal 20 % hold-out vs Kaggle leaderboard

```r
# grouped bars
plot_df <- results %>% pivot_longer(-Pipeline, names_to = "Source", values_to = "F1")
ggplot(plot_df, aes(Pipeline, F1, fill = Source)) +
  geom_col(position = position_dodge(0.7), width = 0.6, colour = "black") +
  scale_fill_brewer(palette = "Blues") +
  labs(title = "Internal vs Kaggle F1 scores",
       y = "F1 score",
       fill = "Score source") +
  theme_minimal(base_size = 12) +
  theme(
    plot.background  = element_rect(fill = "white", colour = "white"),
    panel.background = element_rect(fill = "white", colour = "white")
  )
```

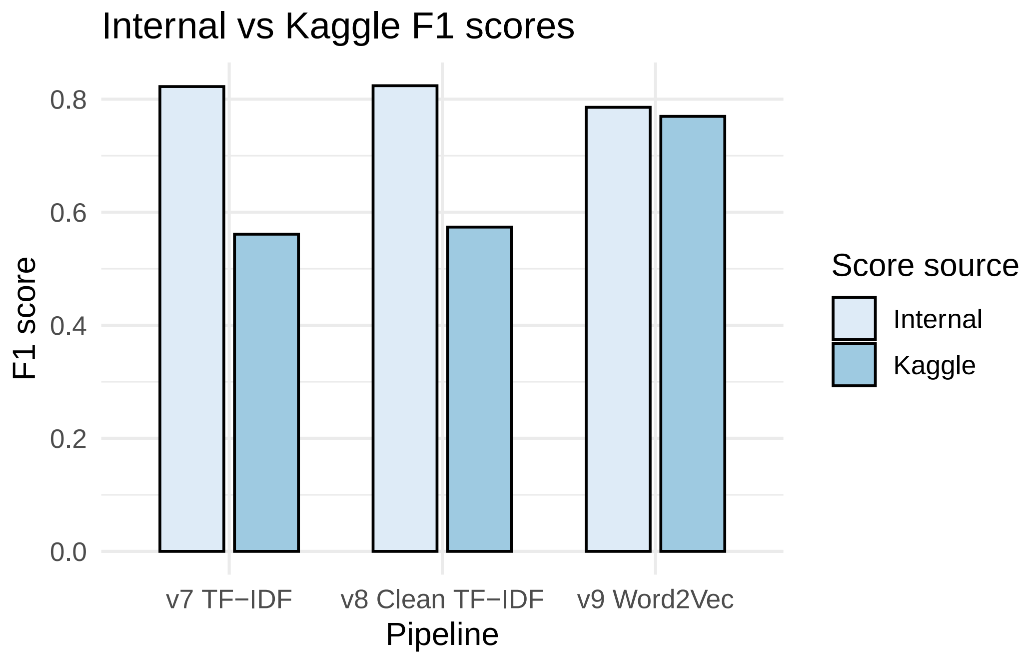

Figure 1: Internal 20 % hold-out vs Kaggle leaderboard

Why Kaggle is lower

Random splits leak tweets from the same event into both train and test, inflating F1. Kaggle's hidden set keeps events together and has a different class balance, so scores drop.

---

4 Lessons Learned

- Word2Vec gains about 0.20 F1 over sparse counts on both evaluations.
- Extra cleaning offers negligible benefit.
- Robust validation is essential – naïve random splits overstate performance.

---

5 Next Steps

1. Hyper-parameter search.
2. BERT baseline via HuggingFace.
3. Ensemble TF-IDF and embeddings.

Appendix

Full session info (click to expand)

```r
sessionInfo()
```

```
R version 4.5.1 (2025-06-13 ucrt)
Platform: x86_64-w64-mingw32/x64
Running under: Windows 11 x64 (build 26100)

Matrix products: default
  LAPACK version 3.12.1

locale:
[1] LC_COLLATE=English_United States.utf8
[2] LC_CTYPE=English_United States.utf8
[3] LC_MONETARY=English_United States.utf8
[4] LC_NUMERIC=C
[5] LC_TIME=English_United States.utf8

time zone: America/New_York
tzcode source: internal

attached base packages:
[1] stats     graphics  grDevices datasets  utils     methods   base

other attached packages:
 [1] tokenizers_0.3.0     tidytext_0.4.2       stopwords_2.3
 [4] caTools_1.18.3       tm_0.7-16            NLP_0.3-2
 [7] textstem_0.1.4       koRpus.lang.en_0.1-4 koRpus_0.13-8
[10] sylly_0.1-6          word2vec_0.4.0       xgboost_1.7.11.1
```

```
[13] caret_7.0-1          lattice_0.22-7       lubridate_1.9.4
[16] forcats_1.0.0        stringr_1.5.1        dplyr_1.1.4
[19] purrr_1.1.0          readr_2.1.5          tidyr_1.3.1
[22] tibble_3.3.0         ggplot2_3.5.2        tidyverse_2.0.0

loaded via a namespace (and not attached):
 [1] qdapRegex_0.7.10     bitops_1.0-9         pROC_1.18.5
 [4] rlang_1.1.6          magrittr_2.0.3       e1071_1.7-16
 [7] compiler_4.5.1       vctrs_0.6.5          reshape2_1.4.4
[10] pkgconfig_2.0.3      crayon_1.5.3         fastmap_1.2.0
[13] labeling_0.4.3       rmarkdown_2.29       prodlim_2025.04.28
[16] tzdb_0.5.0           bit_4.6.0            xfun_0.52
[19] textshape_1.7.5      jsonlite_2.0.0       recipes_1.3.1
[22] SnowballC_0.7.1      syuzhet_1.0.7        parallel_4.5.1
[25] R6_2.6.1             stringi_1.8.7        RColorBrewer_1.1-3
[28] textclean_0.9.3      parallelly_1.45.0    rpart_4.1.24
[31] Rcpp_1.1.0           iterators_1.0.14     knitr_1.50
[34] future.apply_1.20.0  Matrix_1.7-3         splines_4.5.1
[37] nnet_7.3-20          timechange_0.3.0     tidyselect_1.2.1
[40] rstudioapi_0.17.1    yaml_2.3.10          timeDate_4041.110
[43] codetools_0.2-20     listenv_0.9.1        plyr_1.8.9
[46] withr_3.0.2          evaluate_1.0.4       future_1.58.0
[49] survival_3.8-3       proxy_0.4-27         xml2_1.3.8
[52] pillar_1.11.0        lexicon_1.2.1        janeaustenr_1.0.0
[55] renv_1.1.4           foreach_1.5.2        stats4_4.5.1
[58] generics_0.1.4       vroom_1.6.5          hms_1.1.3
[61] scales_1.4.0         globals_0.18.0       class_7.3-23
[64] glue_1.8.0           slam_0.1-55          tools_4.5.1
[67] data.table_1.17.8    ModelMetrics_1.2.2.2 gower_1.0.2
[70] grid_4.5.1           ipred_0.9-15         nlme_3.1-168
[73] cli_3.6.5            lava_1.8.1           gtable_0.3.6
[76] digest_0.6.37        farver_2.1.2         htmltools_0.5.8.1
[79] lifecycle_1.0.4      hardhat_1.4.1        bit64_4.6.0-1
[82] sylly.en_0.1-3       MASS_7.3-65
```