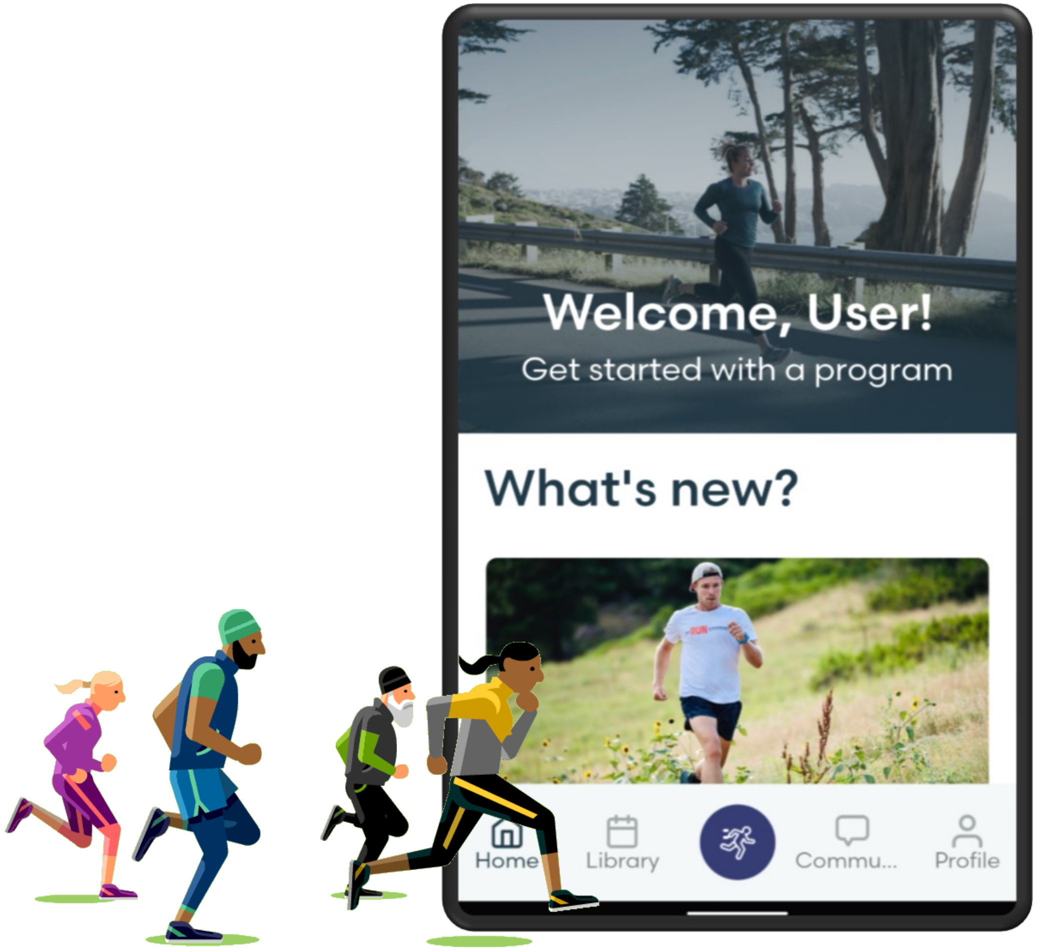
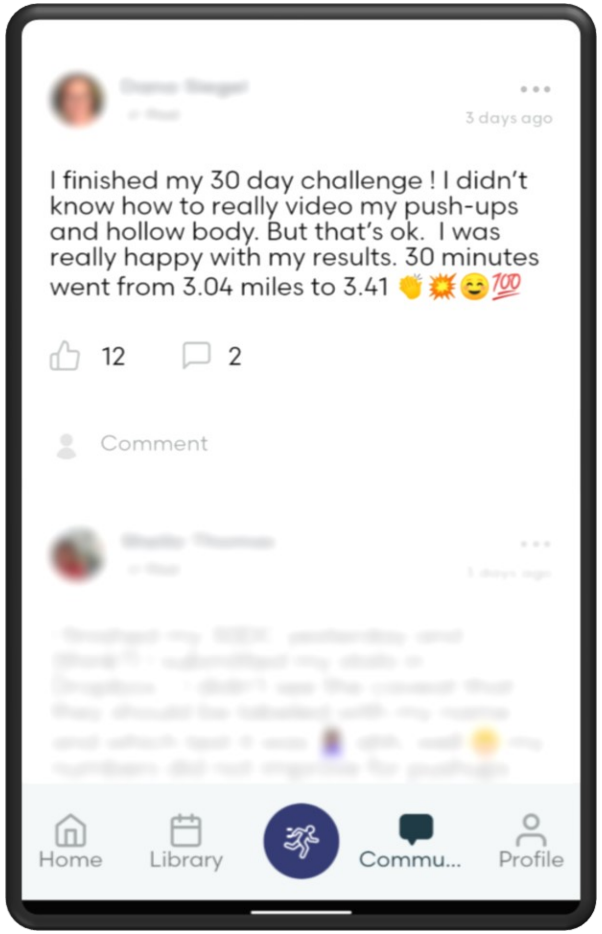
Beyond Words

predict user churn with text (meta)data

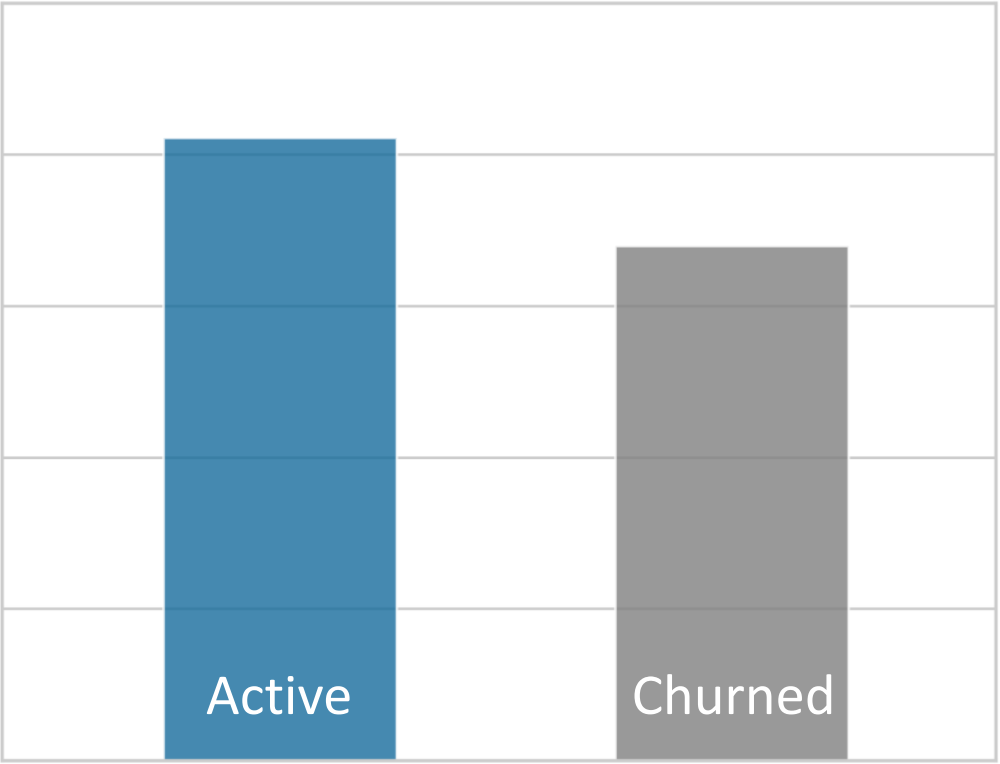
Eric Zhang

Text data from user in-app communication





User churn, big impact on revenue



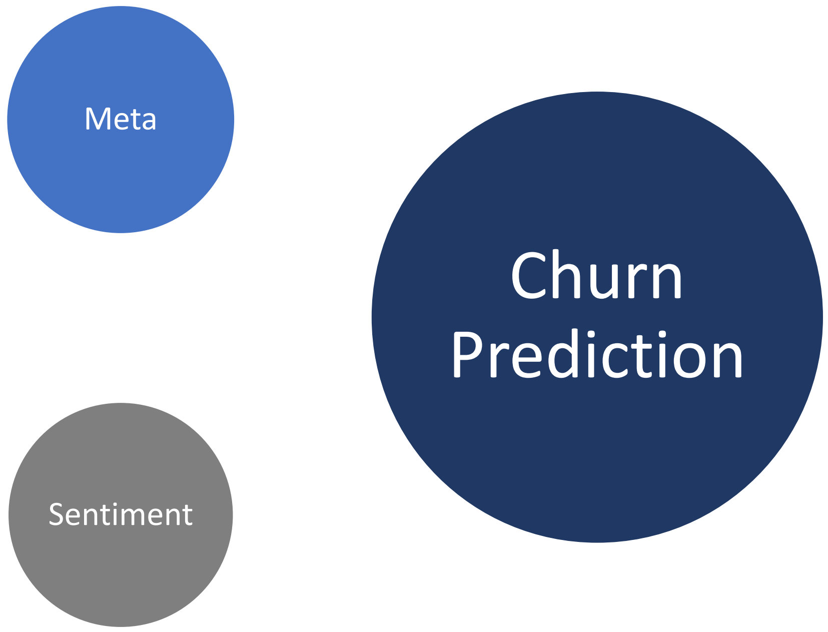
Subscribers

Note: data are not disclosed due to the confidentiality reasons.

Use text data to predict user churn

Meta

Sentiment

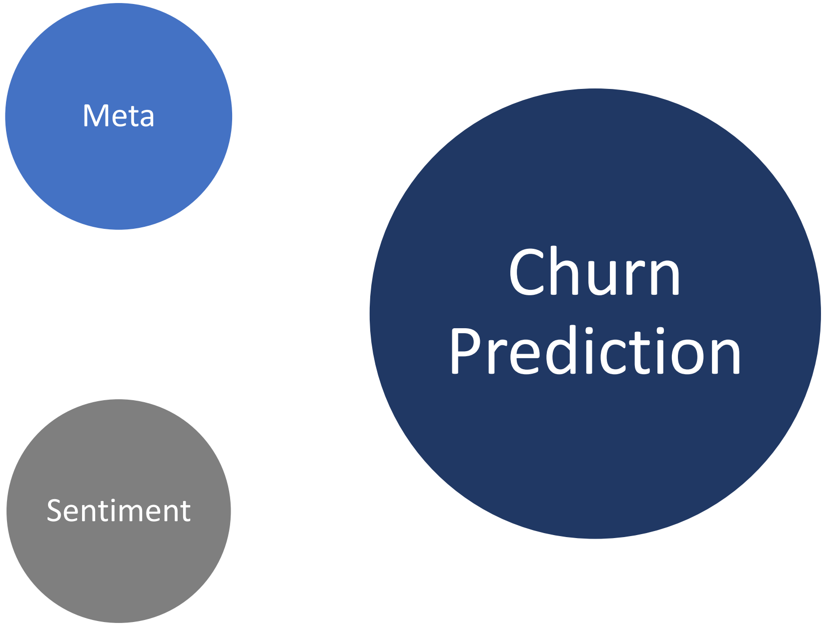


Use text data to predict user churn

Meta

- number of character/ text
- likes received
- timestamp

Sentiment



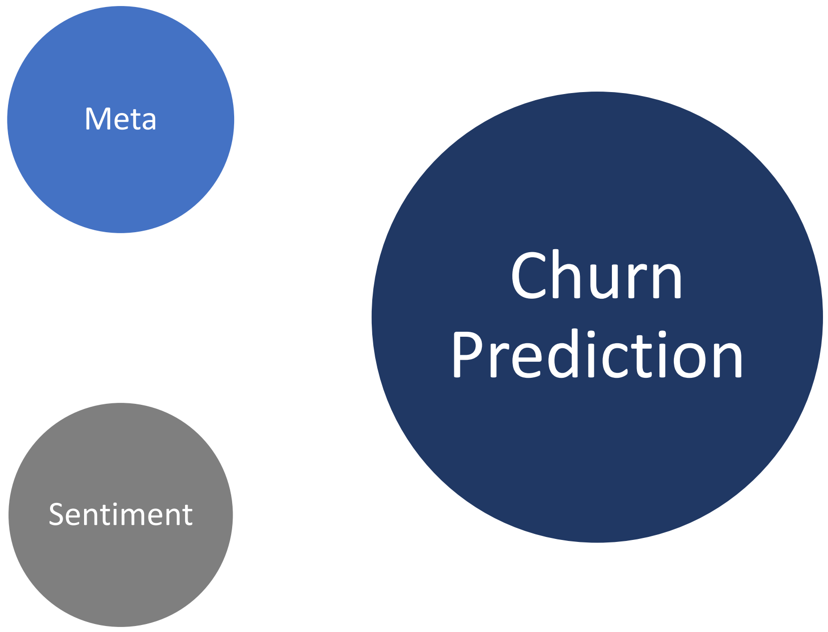
Use text data to predict user churn

Meta

- number of character/ text
- likes received
- timestamp

Sentiment

- happy or frustrated



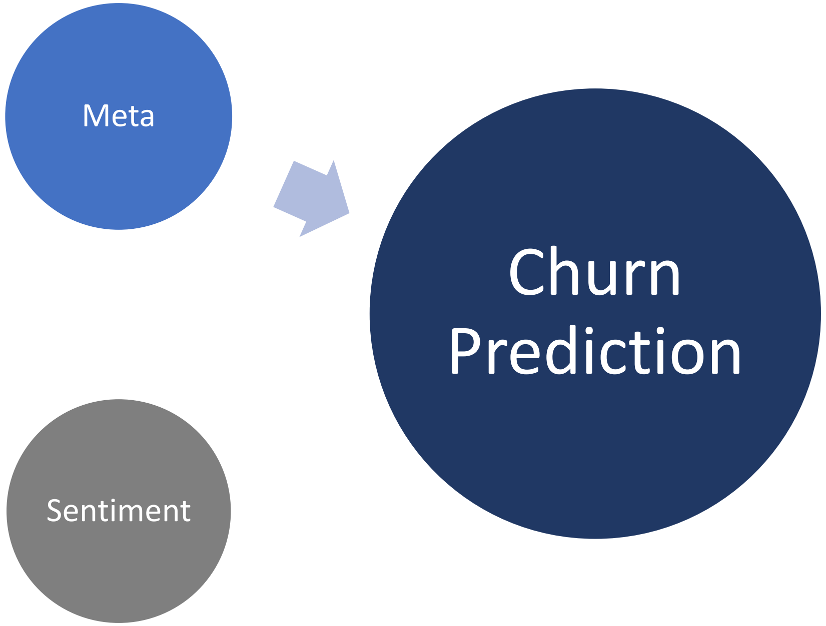
Use text data to predict user churn

Meta

- number of character/ text
- likes received
- timestamp

Sentiment

- happy or frustrated



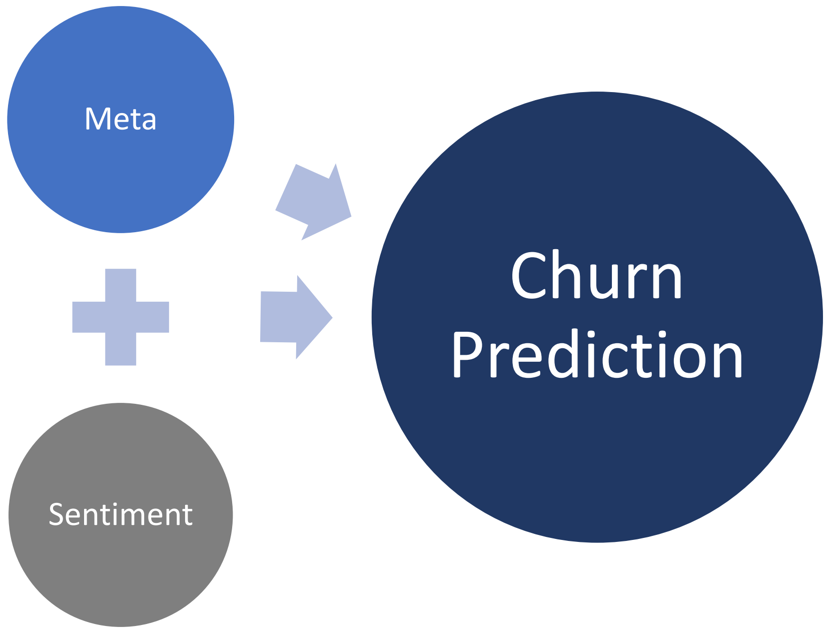
Use text data to predict user churn

Meta

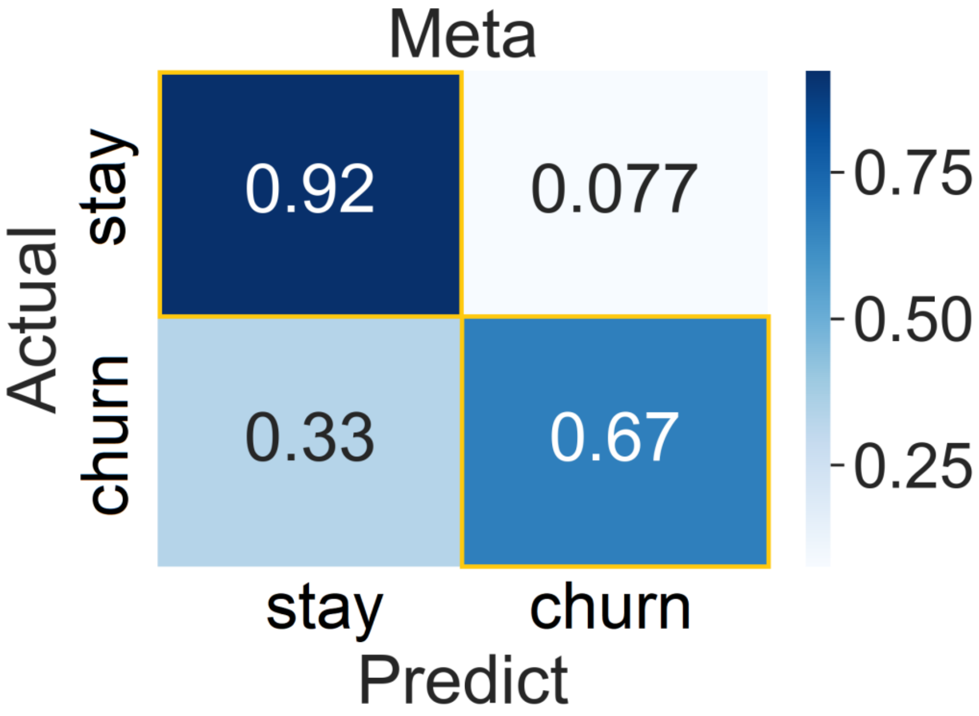
- number of character/ text
- likes received
- timestamp

Sentiment

- happy or frustrated



Text meta: 85% accuracy on churn prediction



Accuracy: 85.2%
Precision: 76.9%
Recall: 66.7%

Train 60% | Test 40%

Stratified, 5 folds CV

Sentiment analysis by NLP

- VADER
- Off-the-shelf BERT

Sentiment analysis by NLP

- ~~VADER~~
- ~~Off-the-shelf BERT~~

I wonder why we torture you guys so badly 😂

VADER: -0.8356
Off-the-shelf BERT: negative 97%

Overview of user texts

"Thanks Coach" "Good Job" "Great" "Happy"

Active Users

Churned Users

Overview of user texts

“Thanks Coach” “Good Job” “Great” “Happy”

- Similar high frequency keywords
- Positive & Supportive

Active Users

Churned Users

Overview of user texts

“Thanks Coach” “Good Job” “Great” “Happy”

- Similar high frequency keywords
- Positive & Supportive
- “Plug-and-play” not working

Active Users

Churned Users

Fine-tuned BERT for sentiment analysis

BERT fine-tuned

Tone (positivity)

Content (subjectivity)

Fine-tuned BERT for sentiment analysis

BERT fine-tuned

Tone (positivity)

- positive, neutral, ~~negative~~

Content (subjectivity)

Fine-tuned BERT for sentiment analysis

BERT fine-tuned

Tone (positivity)

- positive, neutral, ~~negative~~

Content (subjectivity)

- rich, partial, none

Fine-tuned BERT for sentiment analysis

BERT fine-tuned

Tone (positivity)

- positive, neutral, ~~negative~~

Content (subjectivity)

- rich, partial, none

Positive

"Congrats! Good job!"

- Content score: **0 (none)**

"Congrats! Good job on your first 10 miles!"

- Content score: **0.5 (partial)**

Fine-tuned BERT for sentiment analysis

BERT fine-tuned

Tone (positivity)

- positive, neutral, ~~negative~~

Content (subjectivity)

- rich, partial, none

Positive

"Congrats! Good job!"

- Content score: **0 (none)**

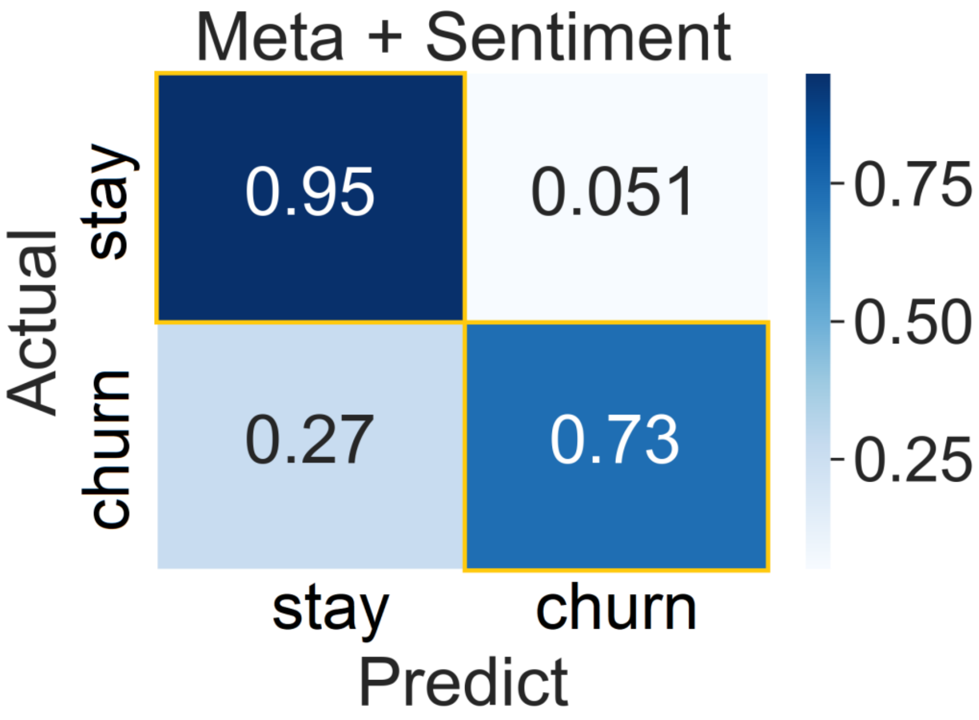
"Congrats! Good job on your first 10 miles!"

- Content score: **0.5 (partial)**

"Congrats! Good job on your first 10 miles! I had my first 10 miles this week too. It was BRUTAL cuz I had to do it in the full sun at the hottest part of the day. But I think it was REALLY good for me!"

- Content score: **1 (rich)**

Meta + Sentiment: 89% accuracy (Meta: 85%)

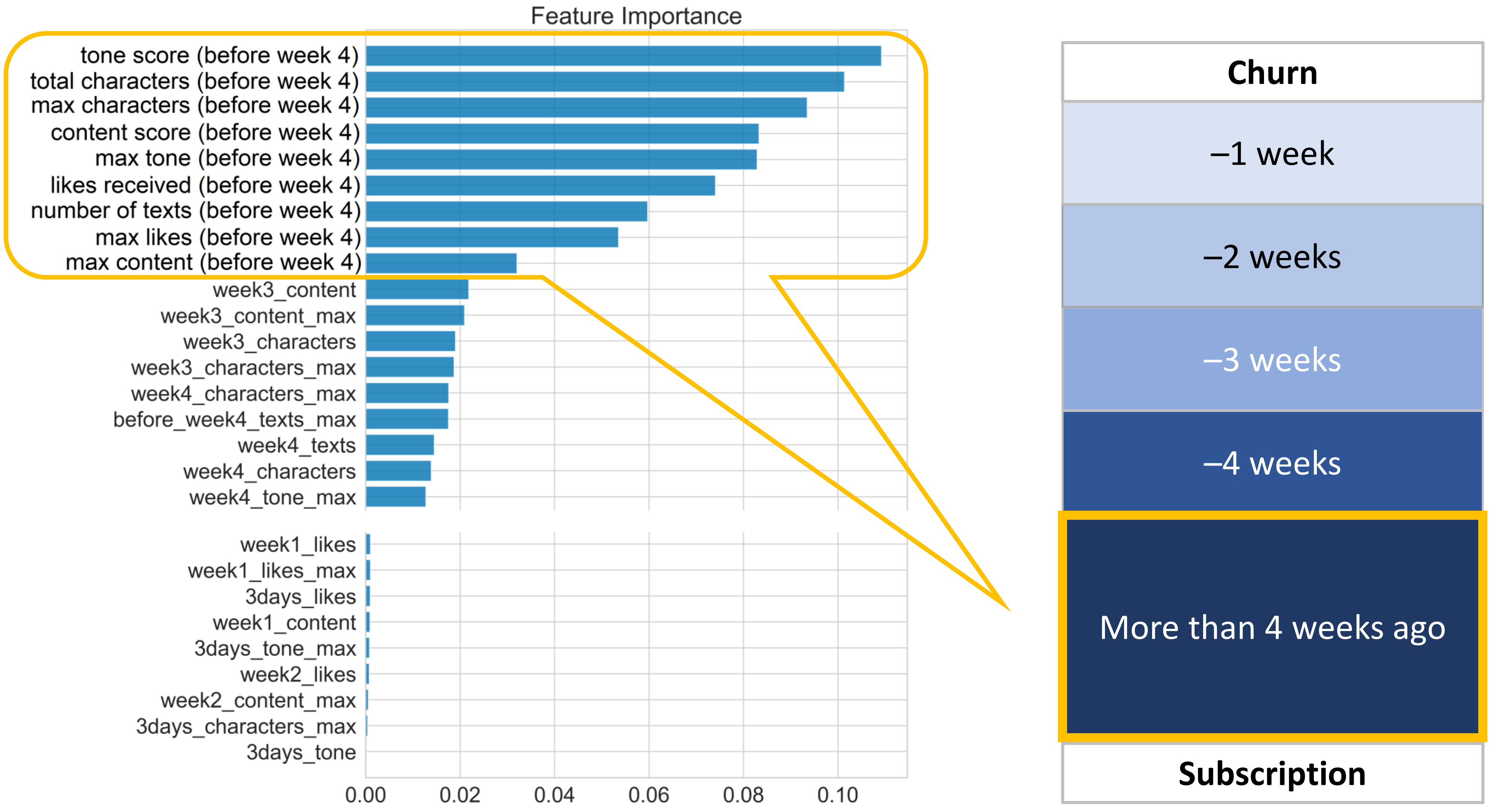


Accuracy: 88.9%
Precision: 84.6%
Recall: 73.3%

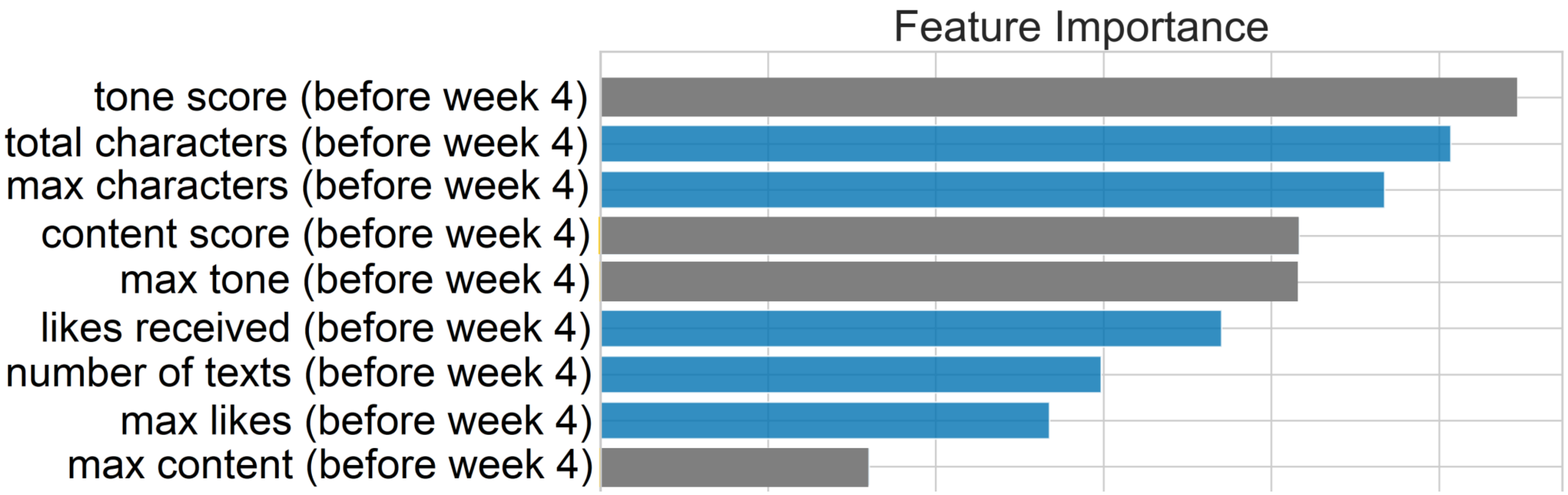
Train 60%	Test 40%

Stratified, 5 folds CV

Top features, 4 weeks before churn



Meta and sentiment features, comparable



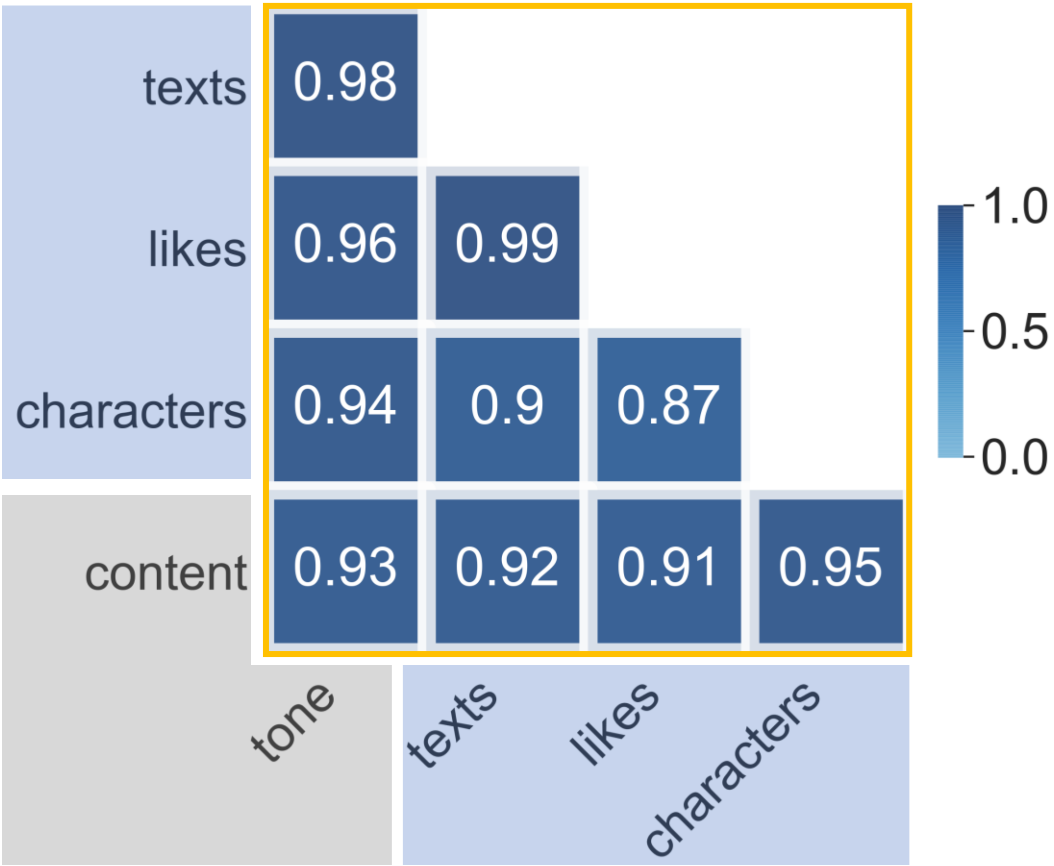
Sentiment features

Text meta features

Strong correlation

Meta features

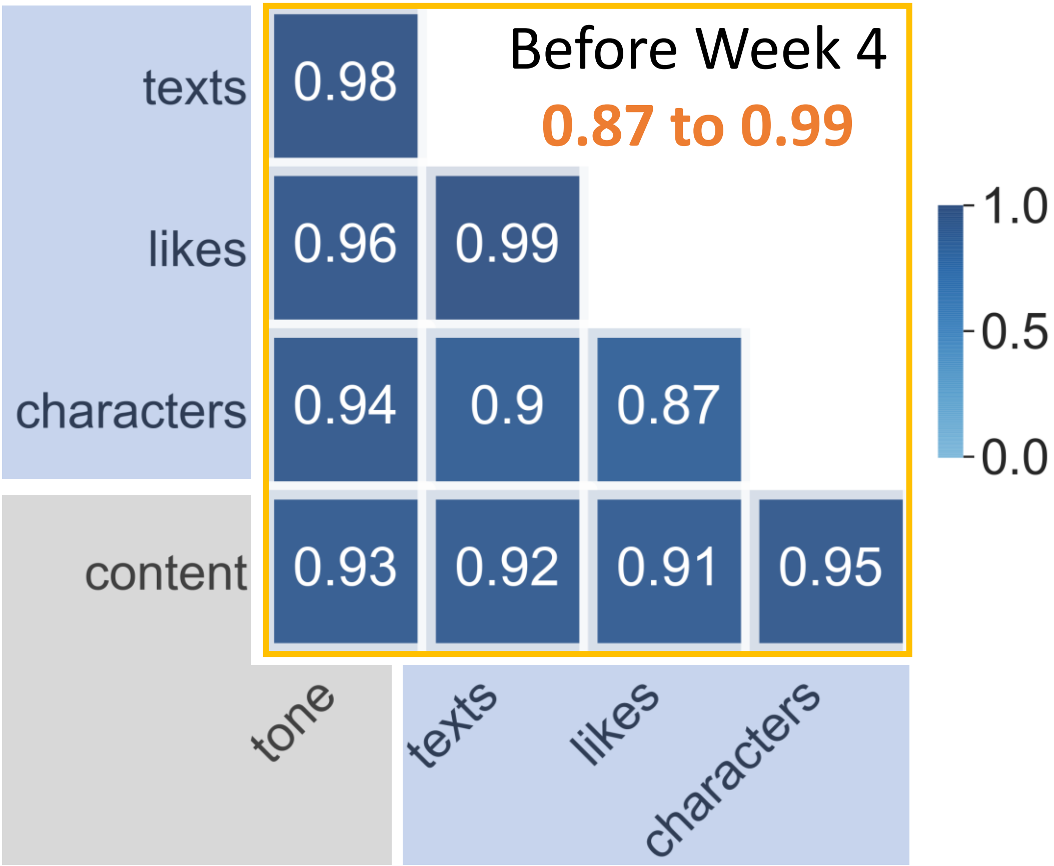
Sentiment features



Strong correlation

Meta features

Sentiment features

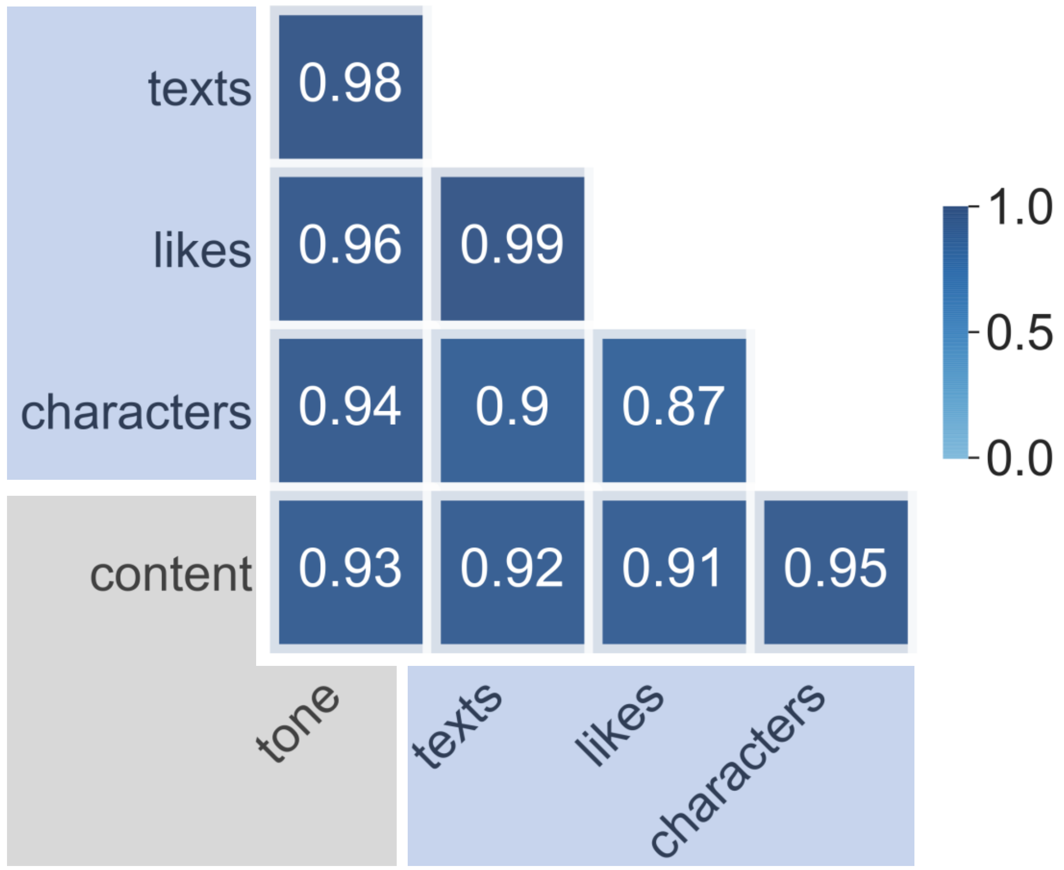


Strong correlation

Meta features

Sentiment features

- Why?



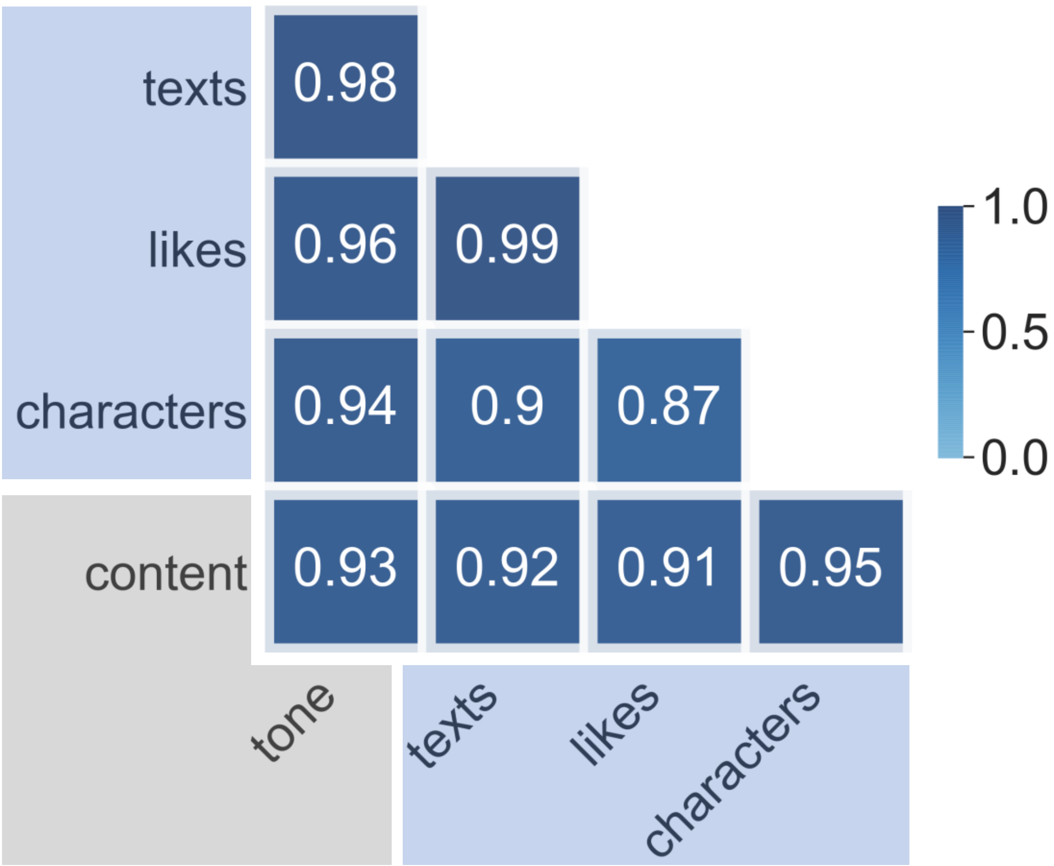
Strong correlation

Meta features

Sentiment features

- Why?

  Text quality: similar

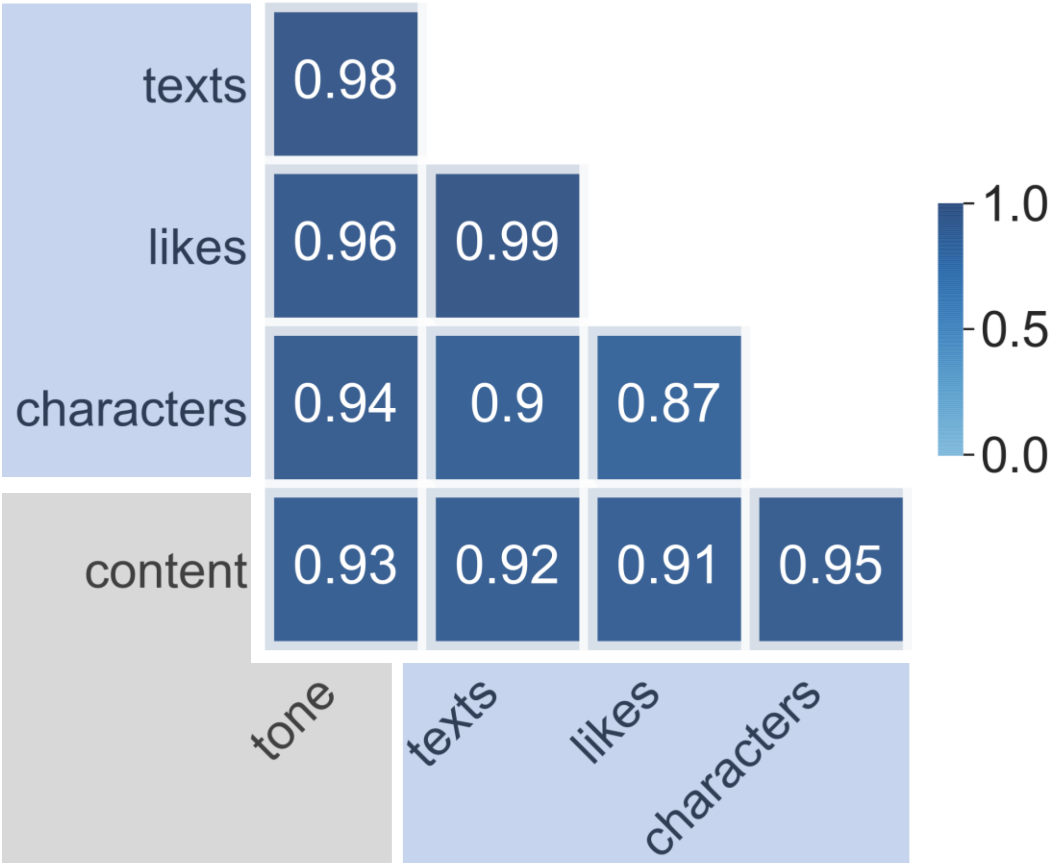


Strong correlation

Meta features

Sentiment features

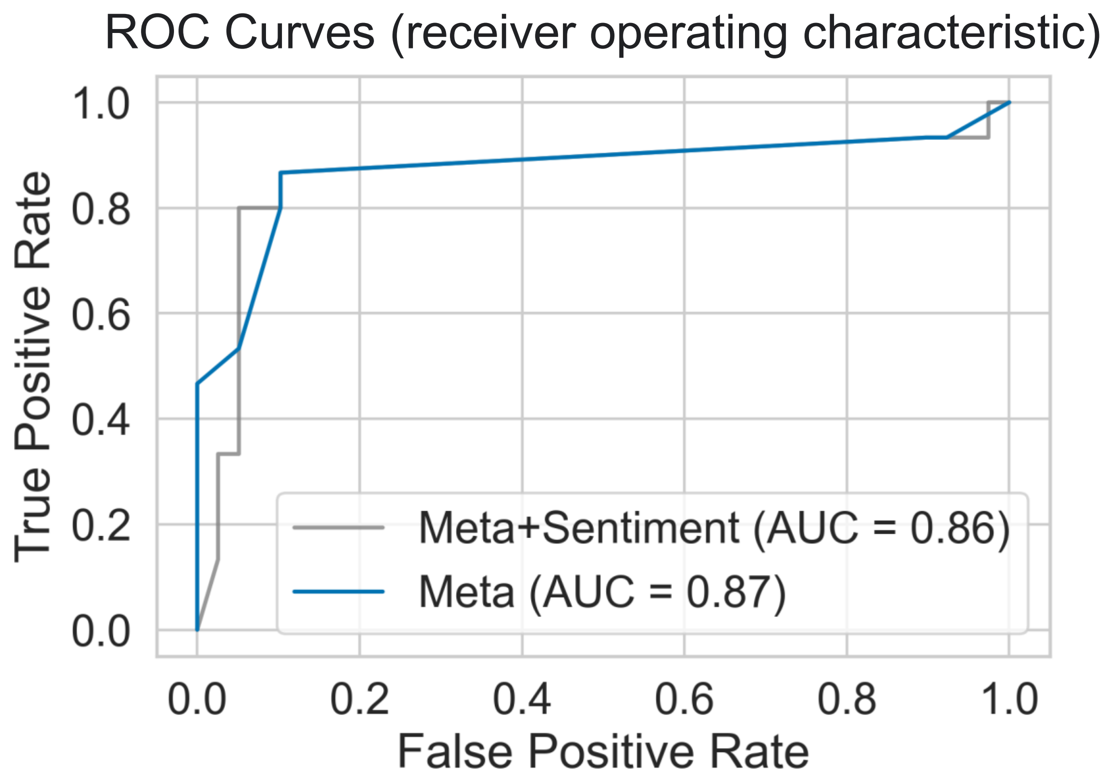
- Why?
  - Text quality: similar
  - Sentiment score: text quantity



Text meta can predict user churn

Text data

- AUC = 0.87 (meta)
- AUC = 0.86 (meta + sentiment)
- comparable



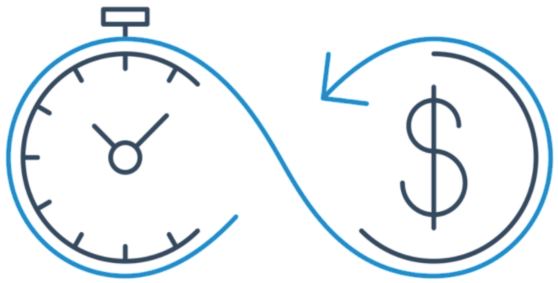
Text meta can predict user churn

Text data

- AUC = 0.87 (meta)
- AUC = 0.86 (meta + sentiment)
- comparable

Text meta good enough

- save time 50% (~~NLP labeling~~, ~~transfer learning~~)
- easy to scale up



Deliverable

Time to act

- **4 weeks** before user churn
- take actions:

  e.g. targeted survey, in-app perks, coach match-up, etc.

Deliverable

Time to act

- **4 weeks** before user churn
- take actions:

  e.g. targeted survey, in-app perks, coach match-up, etc.

Evaluate

- which strategy works the best:

  e.g. 1 coaching session vs 1 month membership
- multi-armed bandit testing on high-risk users (e.g. top 20%)

Takeaways

Text meta

- good enough for churn prediction, save time

Takeaways

Text meta

- good enough for churn prediction, save time

Actionable insight

- predict, intervene, and evaluate

Takeaways

Text meta

- good enough for churn prediction, save time

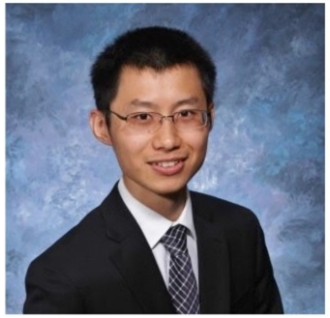
Actionable insight

- predict, intervene, and evaluate

With more data

- real-time prediction and evaluation (sliding window)

Zelong (Eric) Zhang

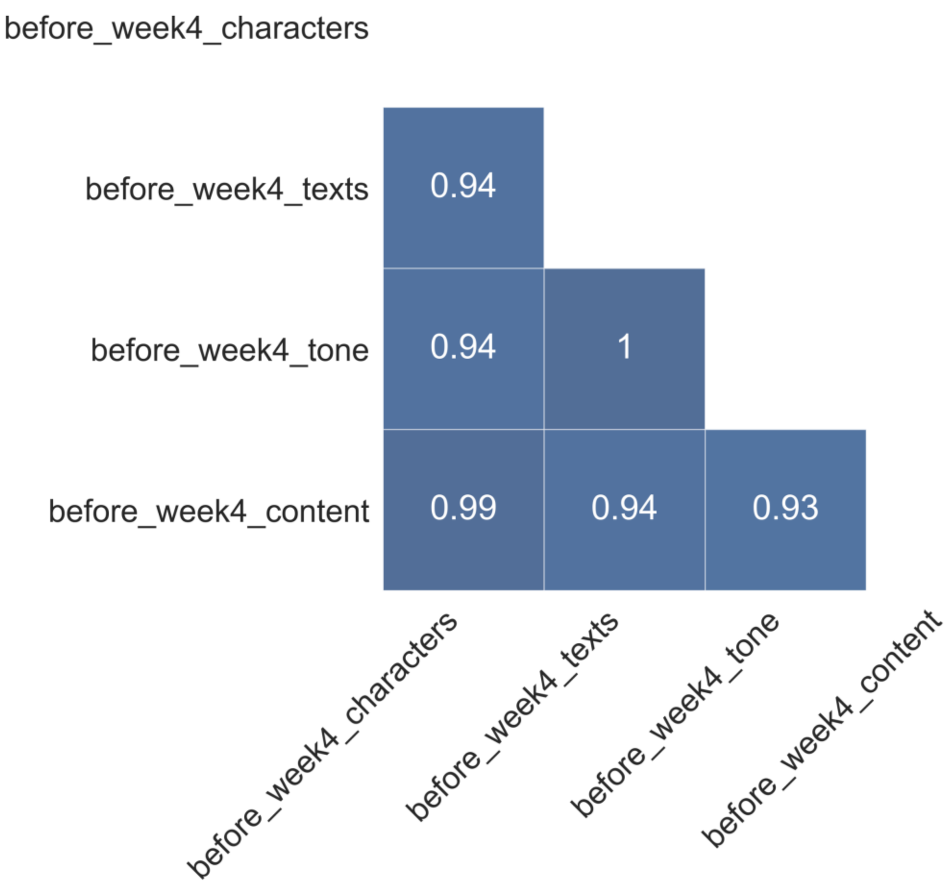


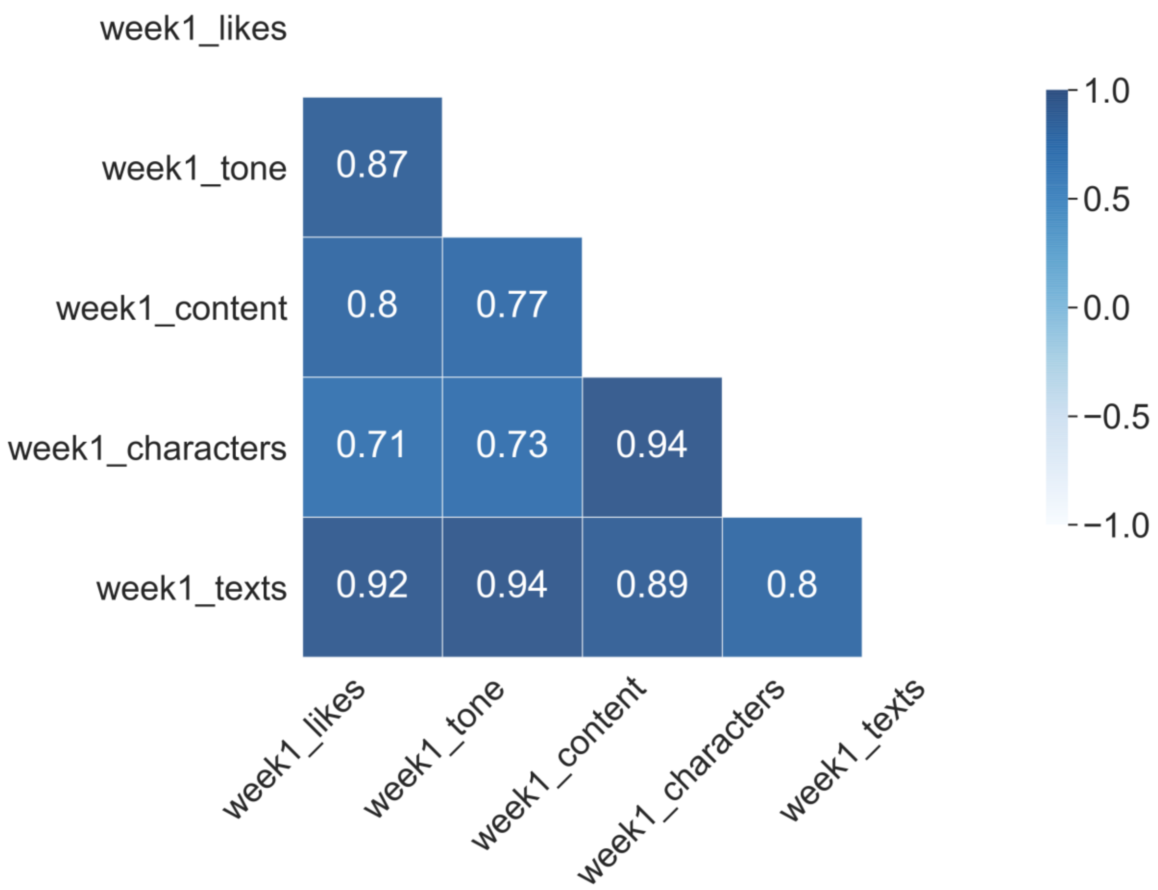
- PhD in Computational Chemistry
- Award-winning film (US DOE), photography
- User Experience and Decision-Making

Stony Brook University

LSU
LOUISIANA STATE UNIVERSITY

Extra Slides



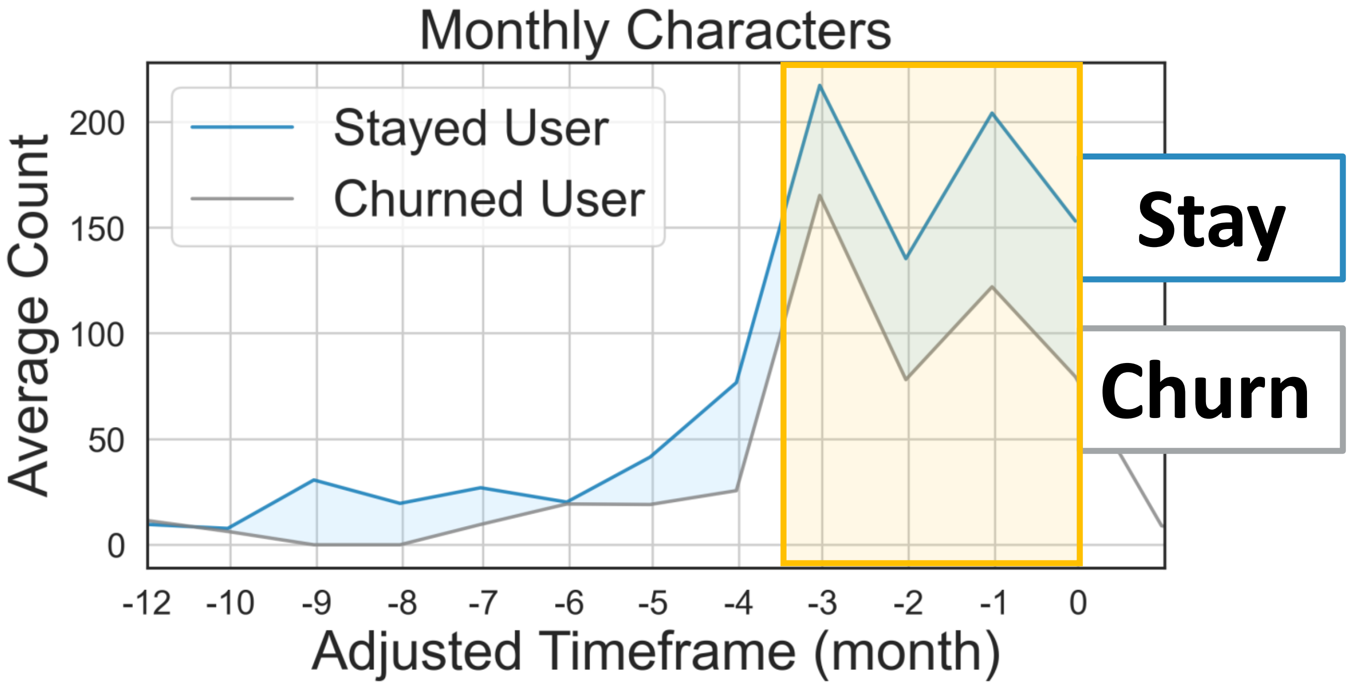


Text meta-data can predict user churn

Strong correlation
- text meta
- sentiment

Text meta features
- Good enough
- Easy to scale up

User in-app communication
- Strong indicator of user churn
- Customer life time 3 to 4 months



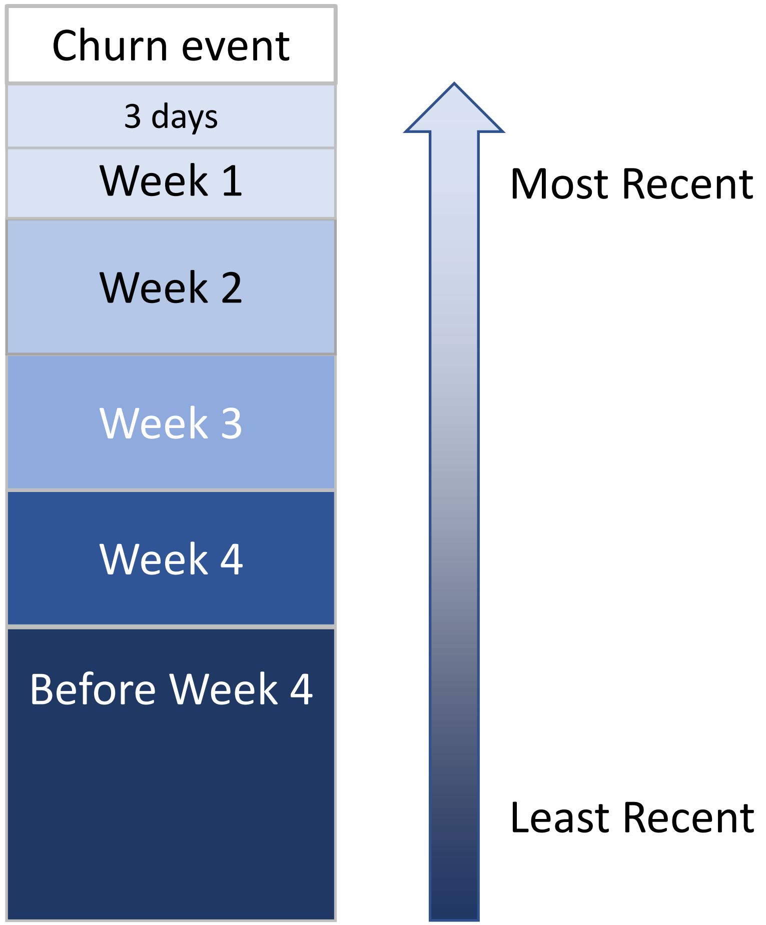
Send surveys to users at high risk of churn

What is the issue?	Actions
This app keeps crashing	Address the technical issue causing the frequent crash
I cannot export my data. This app doesn’t support that.	Evaluate the cost of adding the excise data export function
I am not happy with my coach, s/he is too pushy.	Offer the client with a different coach with the personality the client likes.
The coach doesn’t answer my questions timely	Ask the coach to respond this client’s messages in a timely manner, or add count-down timer.
General dissatisfaction	Offer the client in-app perks - a free month membership, a free one-on-one coach session, priority access to certain special features

Text meta (texts, likes, characters)

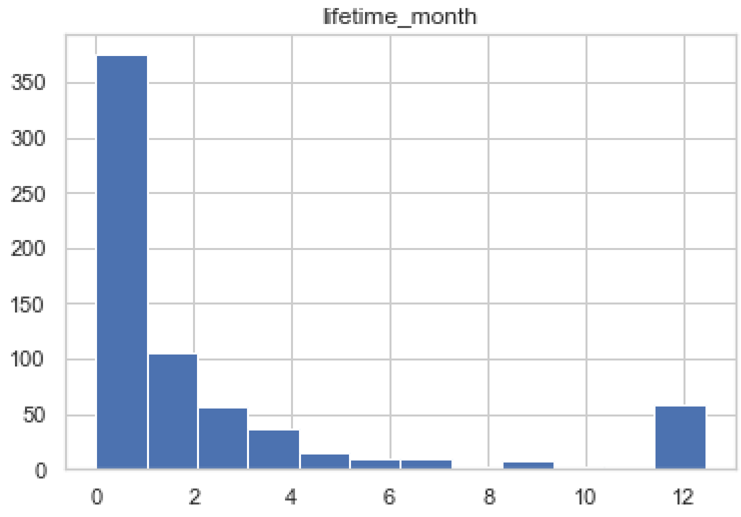
Before decision (churn or stay)

- 3 days
- Week 1
- Week 2
- Week 3
- Week 4
- Before Week 4 (everything)

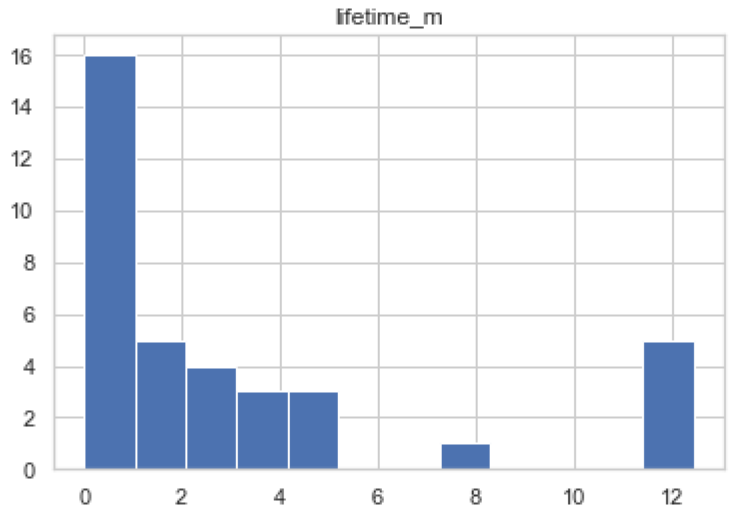


Churn User Lifetime Stats

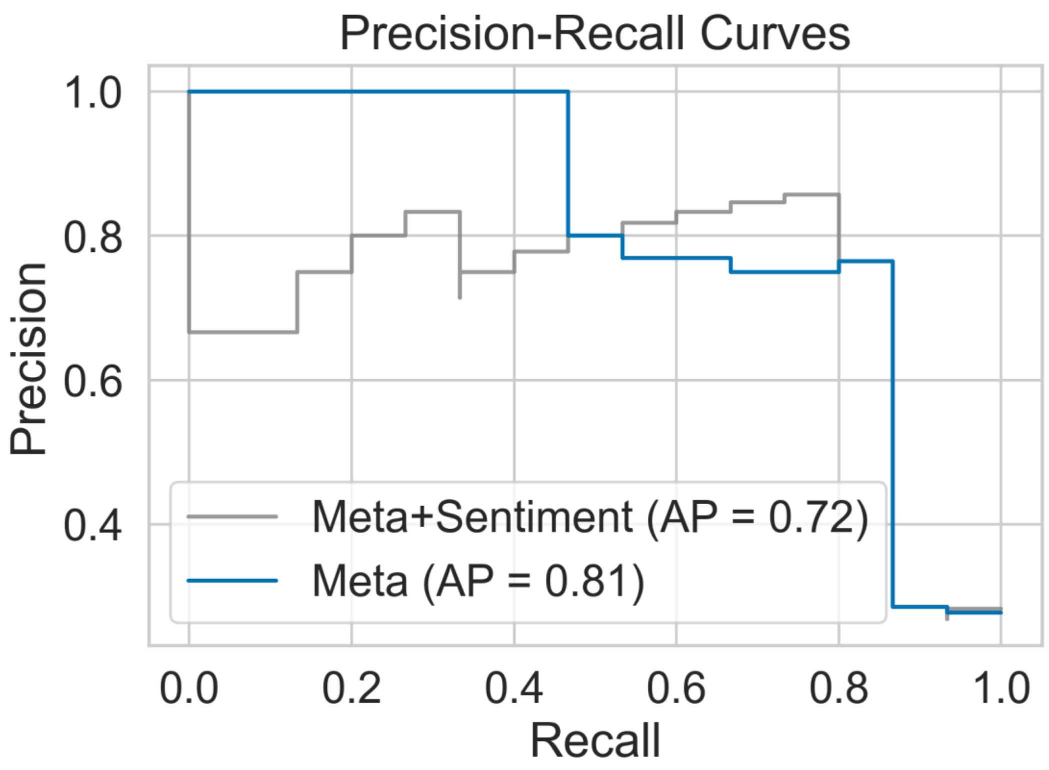
User Subscription	Month
Max	12.4
Min	0
Mean	2.8
Median	1.0
Unbiased variance	10.9
Standard deviation	3.3

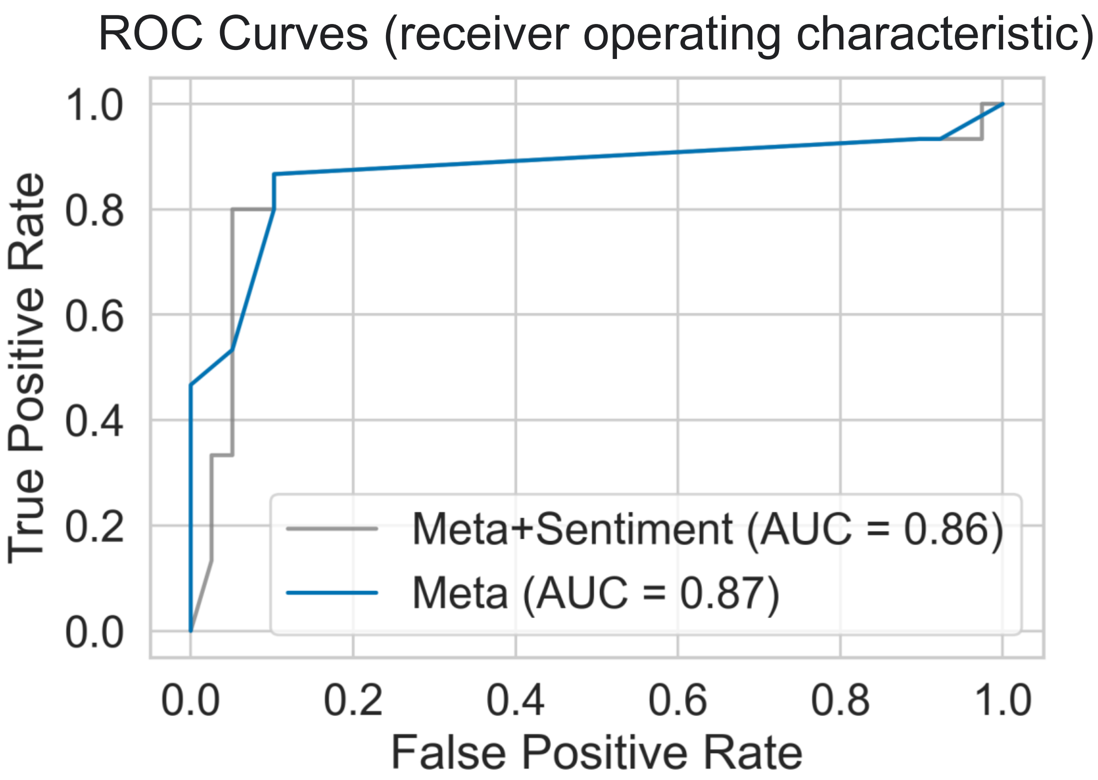


User in Text Data	Month
Max	12.4
Min	0
Mean	3.4
Median	2.0
Unbiased variance	15.2
Standard deviation	3.9

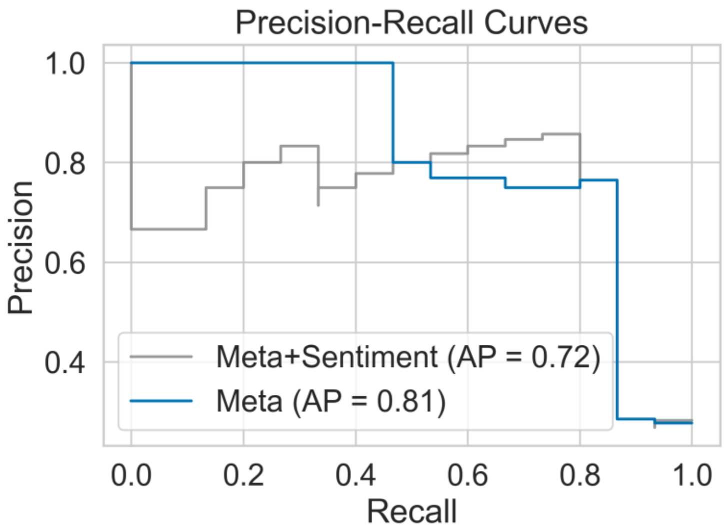


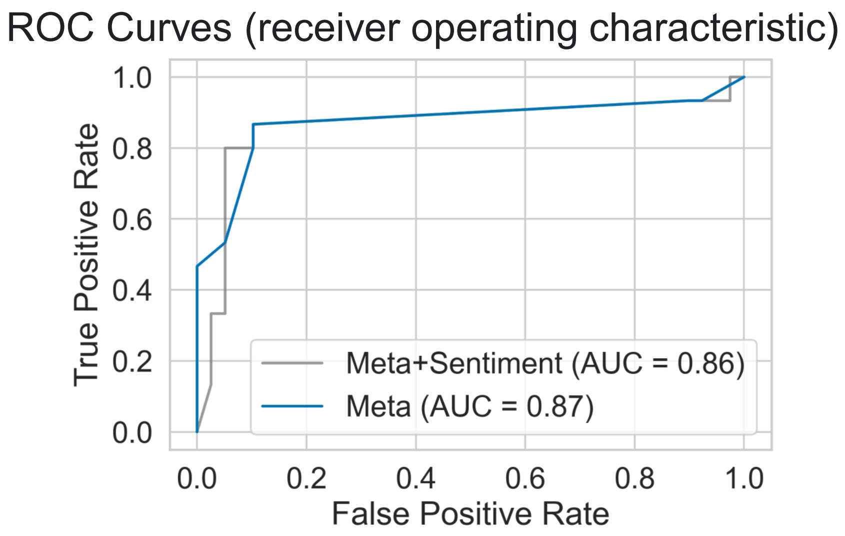
Precision-Recall curves and ROC curves

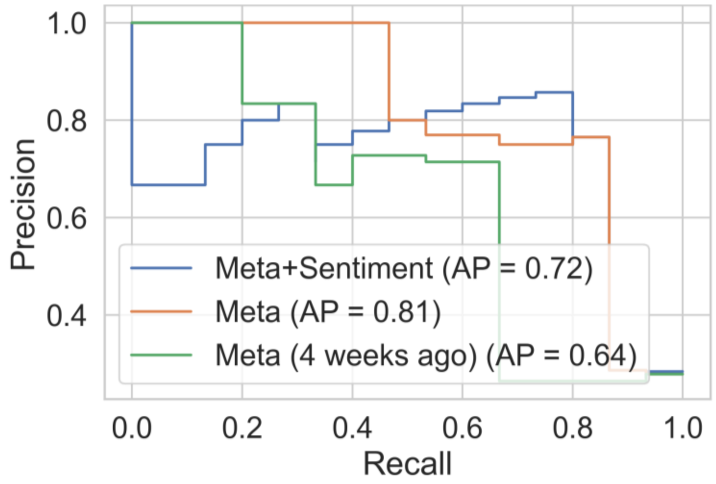


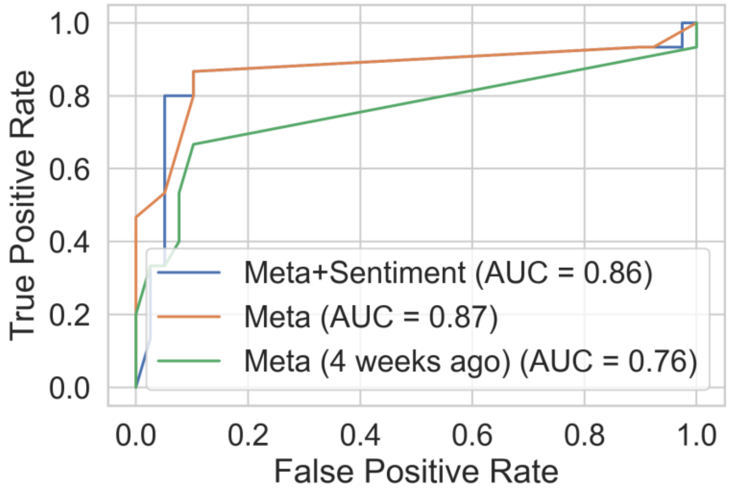


Precision-Recall curves and ROC curves

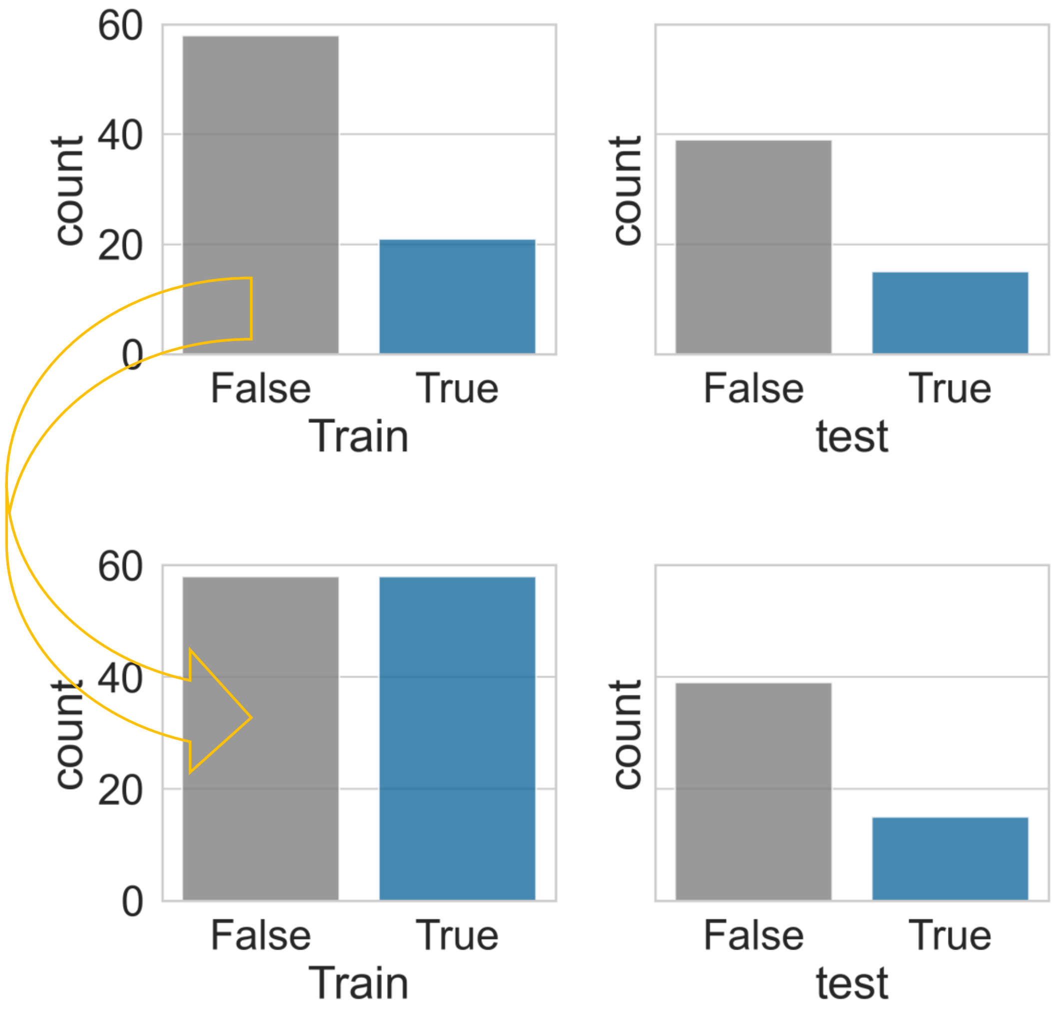








Train and test datasets

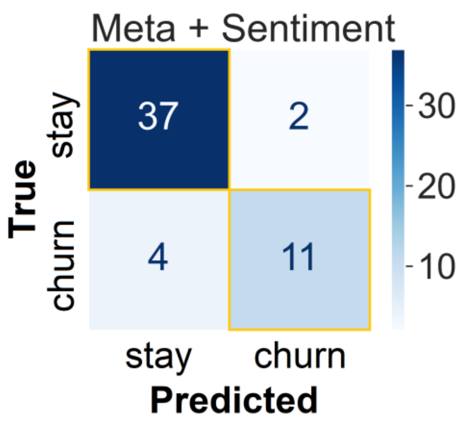


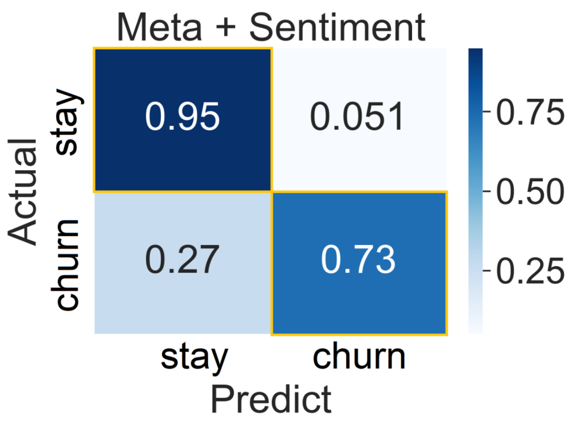
Adaptive Synthetic (ADASYN)

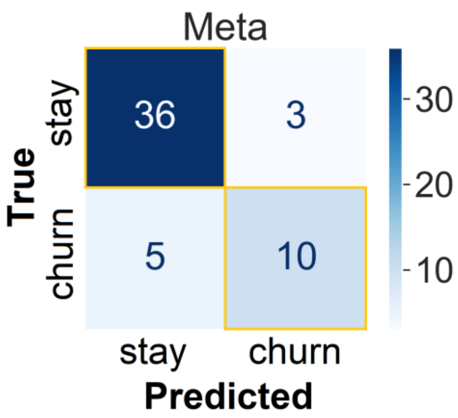
ADASYN is marginally better than SMOTE for this dataset (CV has slightly better recall).

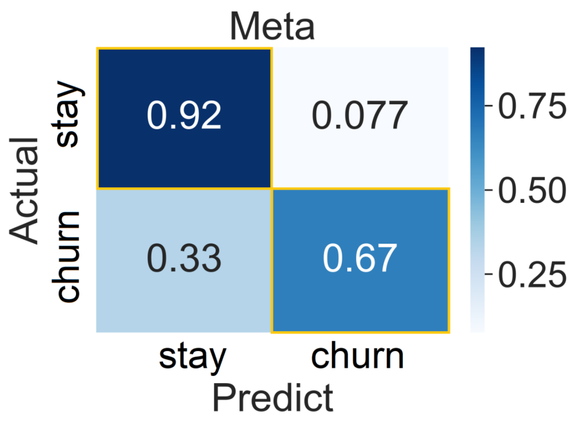
"ADASYN focuses on generating samples next to the original samples which are wrongly classified using a k-Nearest Neighbors classifier while the basic implementation of SMOTE will not make any distinction between easy and hard samples to be classified using the nearest neighbors rule."

Validation metrics







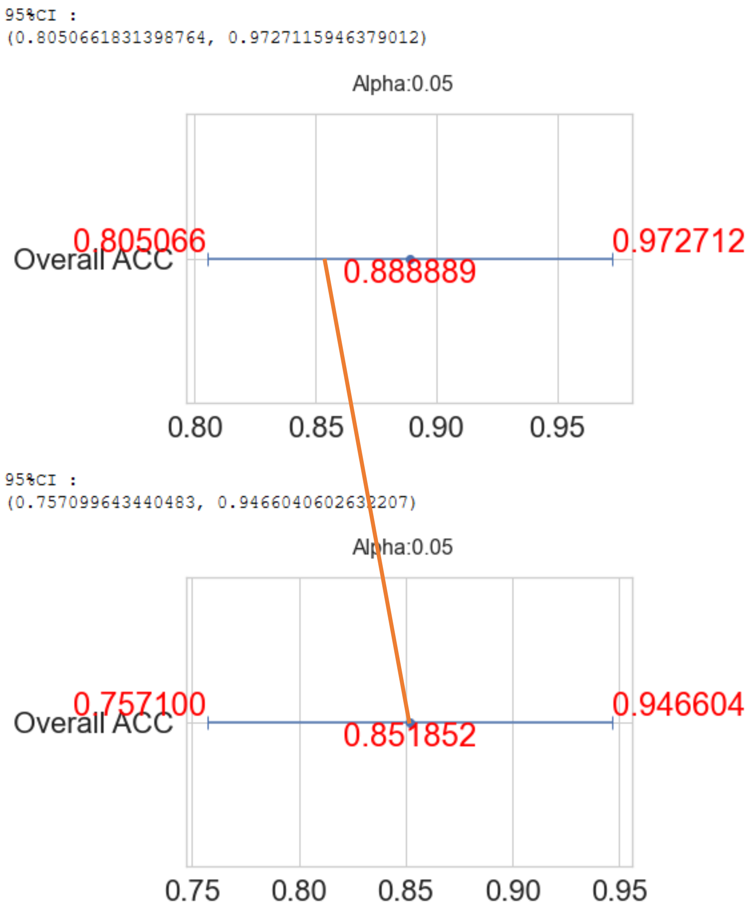


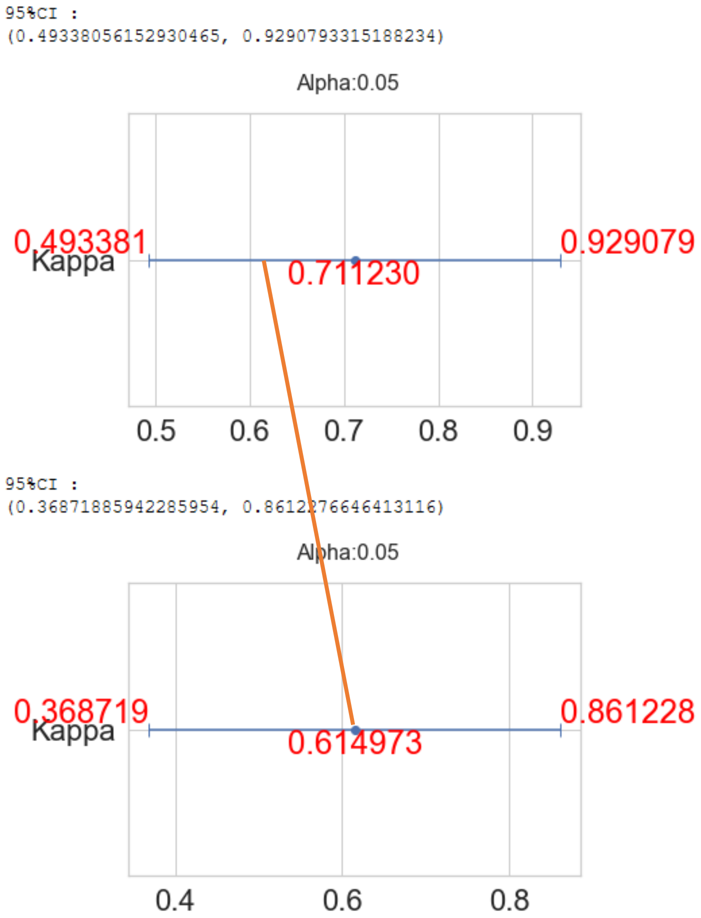
	Meta + Sentiment	CV*		Meta	CV*
Accuracy	0.89	0.93 (0.035)	Accuracy	0.85	0.91 (0.049)
Precision	0.84	0.95 (0.040)	Precision	0.77	0.92 (0.047)
Recall	0.73	0.92 (0.074)	Recall	0.67	0.90 (0.097)
F1	0.79	0.93 (0.036)	F1	0.71	0.91 (0.054)

95% Confidence Interval of Accuracy (0.805, 0.973) vs (0.757, 0.947)

*Cross-validation: stratified, 5 K-folds

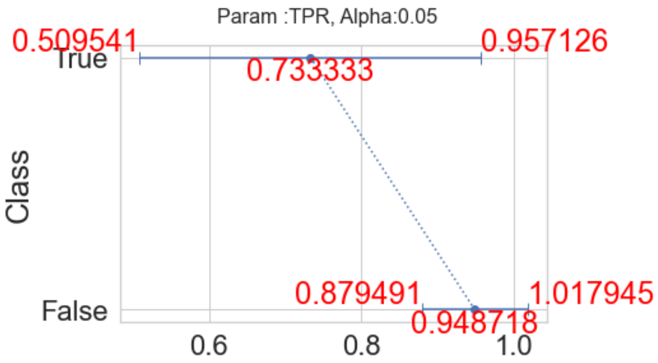
95% Confidence Interval of accuracy, Kappa



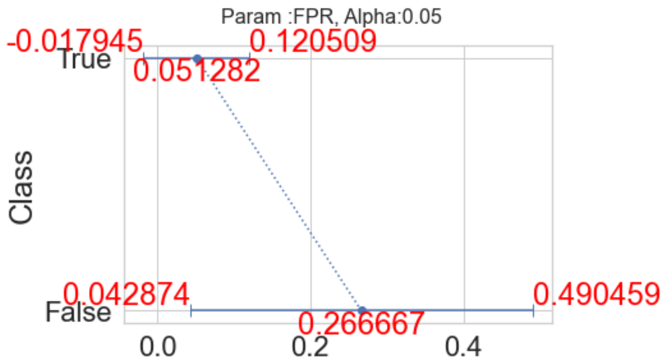


95% Confidence Interval of TPR, FPR

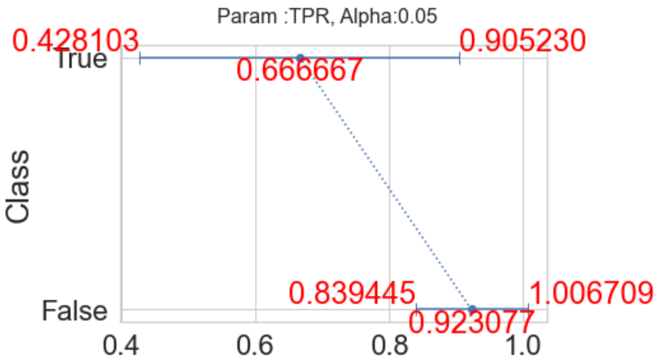
```
95%CI :
False : (0.8794910269654593, 1.017944870470438)
True : (0.5095408368517008, 0.9571258298149657)
```



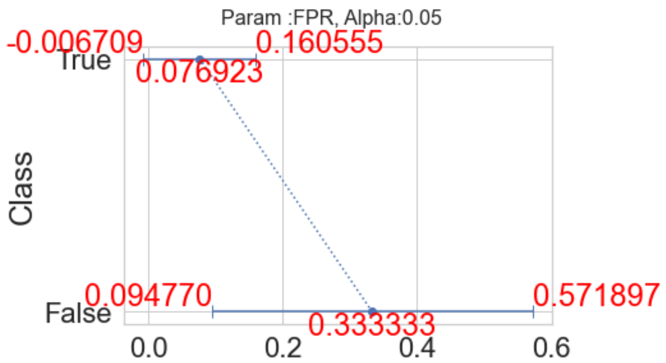

```
95%CI :
False : (0.04287417018503423, 0.4904591631482992)
True : (-0.017944870470437996, 0.12050897303454064)
```



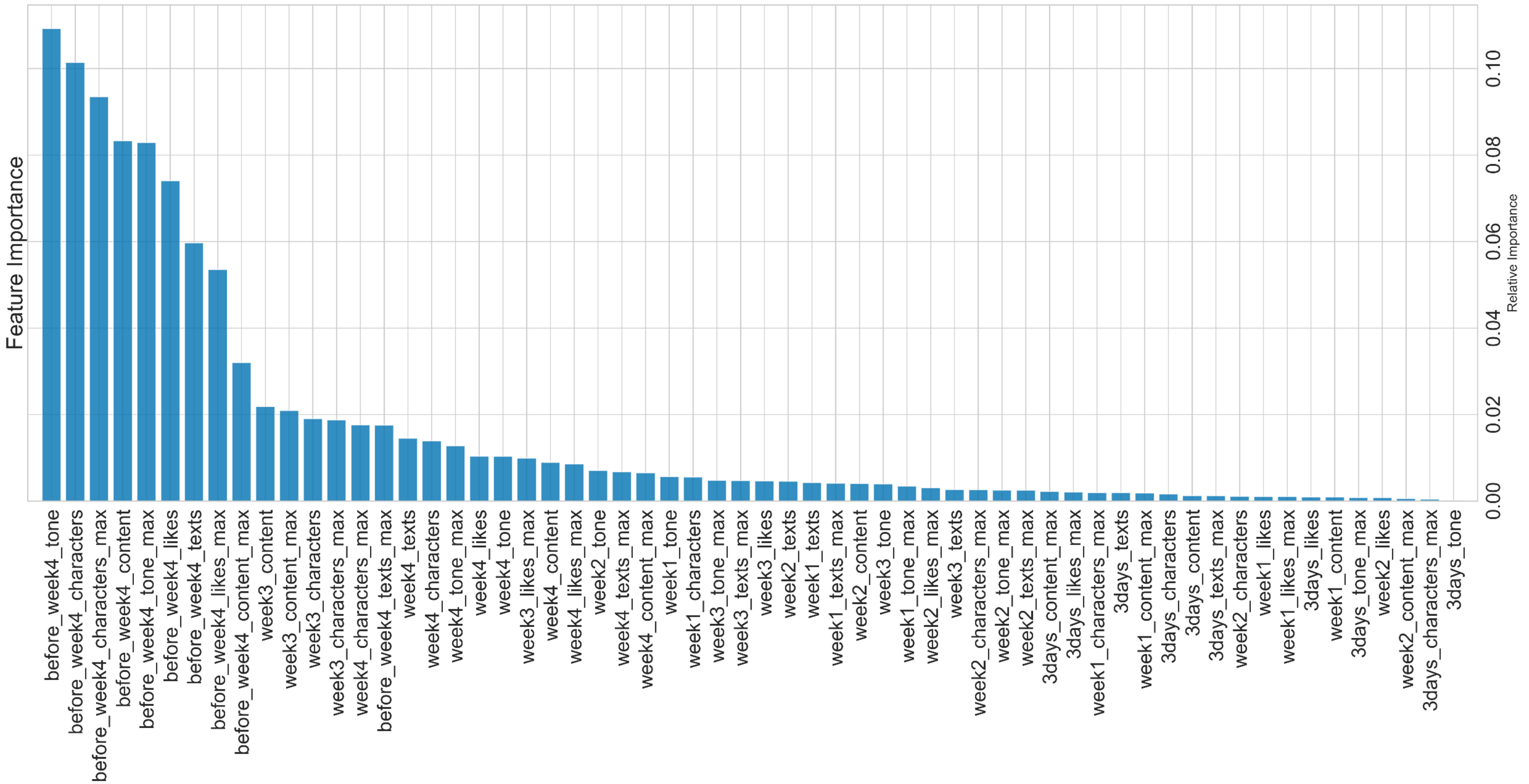

```
95%CI :
False : (0.8394452011845039, 1.0067086449693423)
True : (0.4281030638421943, 0.9052302694911389)
```




```
95%CI :
False : (0.09476973050886103, 0.5718969361578057)
True : (-0.006708644969342259, 0.160554798815496)
```



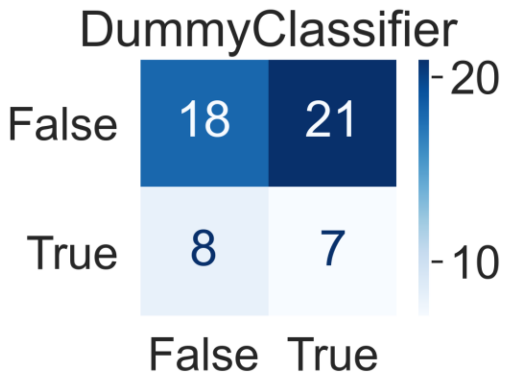
Feature importance

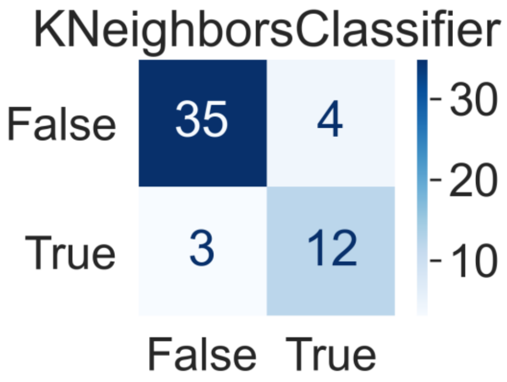


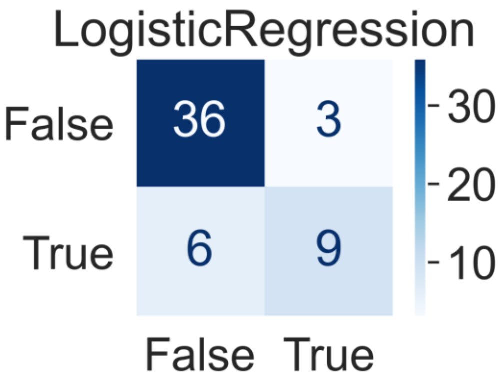
Classification models for churn

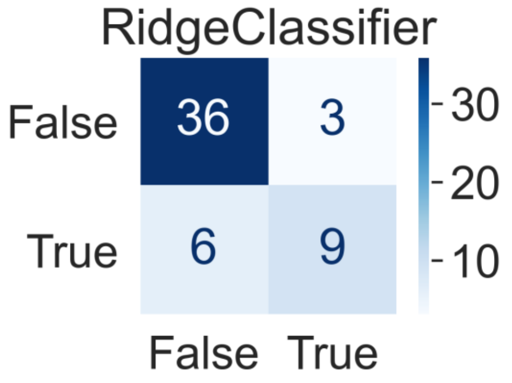
Random Forest	Test	CV	Std. dev.
Accuracy	0.89	0.924	0.042
Precision	0.846	0.951	0.040
Recall	0.733	0.917	0.074
F1	0.786	0.923	0.044

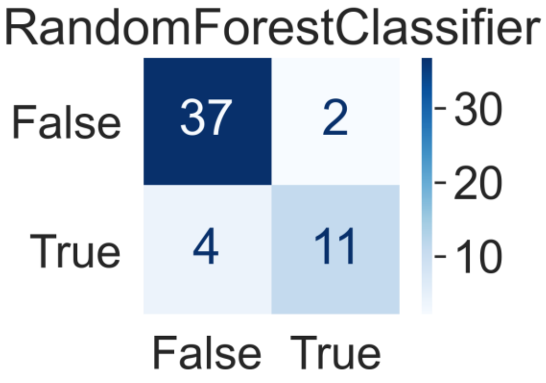
CV: Stratified KFold, 5 splits

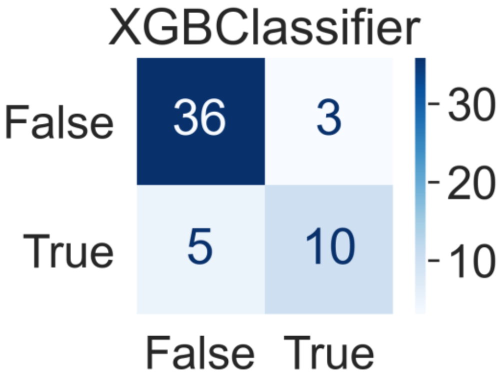


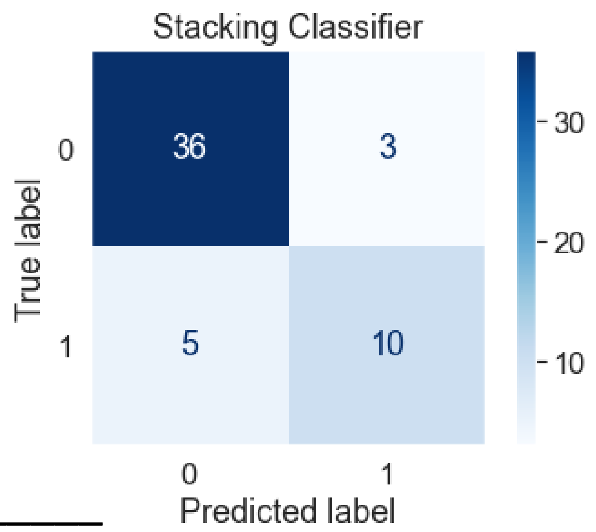






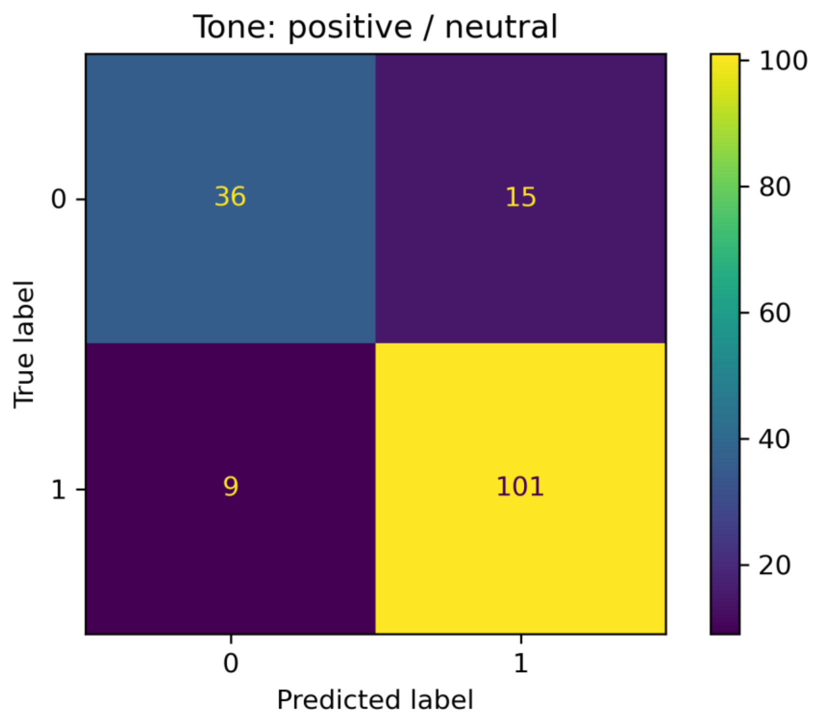


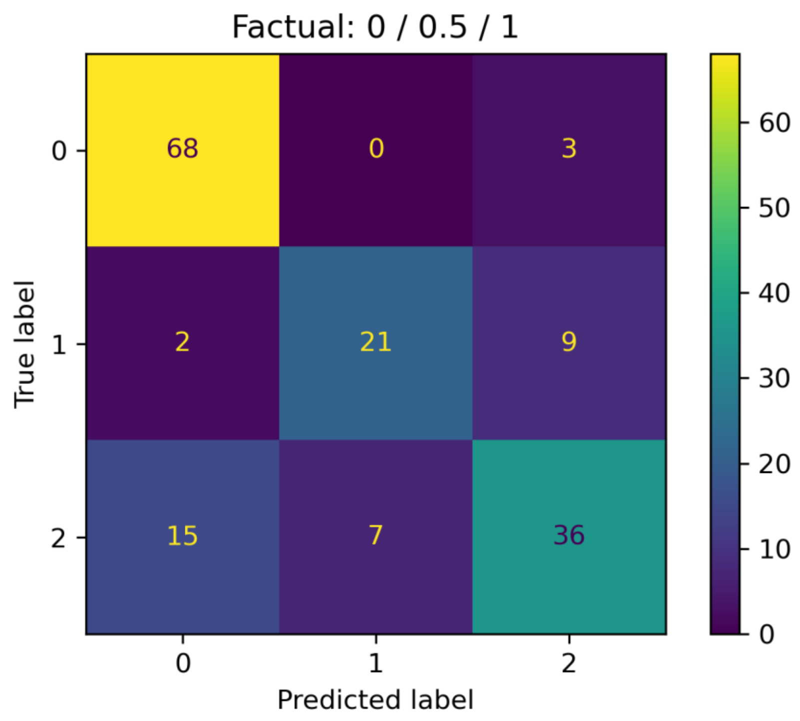




Combined by Logistic Regression

NLP BERT validation metrics





Tone:
accuracy 0.850932

Content:
accuracy 0.776398

Sanity check of sentiment analysis

Predicted by the BERT model I fine-tuned!

	text	Tone	Content	VADER_Score	OTS_BERT_label	OTS_BERT_score
1387	Way to go, Michael!	positive	0.0	0.0000	positive	0.9998
1388	I like to torture myself!!!! 🤪🤪🤪	positive	0.0	-0.5526	negative	0.9992
1389	yes!!!	positive	0.0	0.5538	positive	0.9997
1390	Oh dear that is swollen. Is ice helping?	neutral	0.5	0.5859	negative	0.9983
1391	Night run	neutral	0.5	0.0000	positive	0.5209
1392	Dabbling in swimming and biking. When my fitne...	positive	0.5	0.3382	negative	0.9583

VADER: a lexicon and rule-based sentiment analysis tool that is specifically attuned to sentiments expressed in social media.
OTS BERT: Off-the-shelf version of “distilbert-base-uncased-finetuned-sst-2-english"

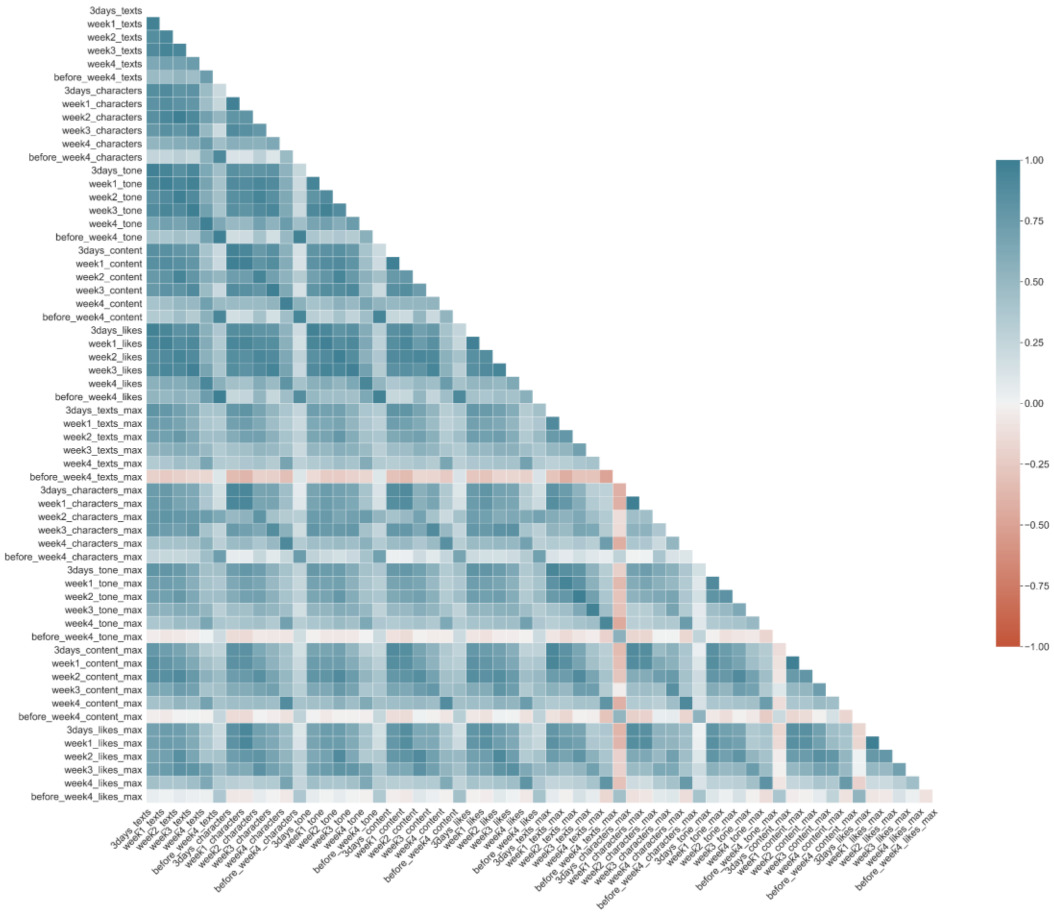
In-app user record

- Registered User: 56603
- User with text data: 773
- User with both registration and text data: 623

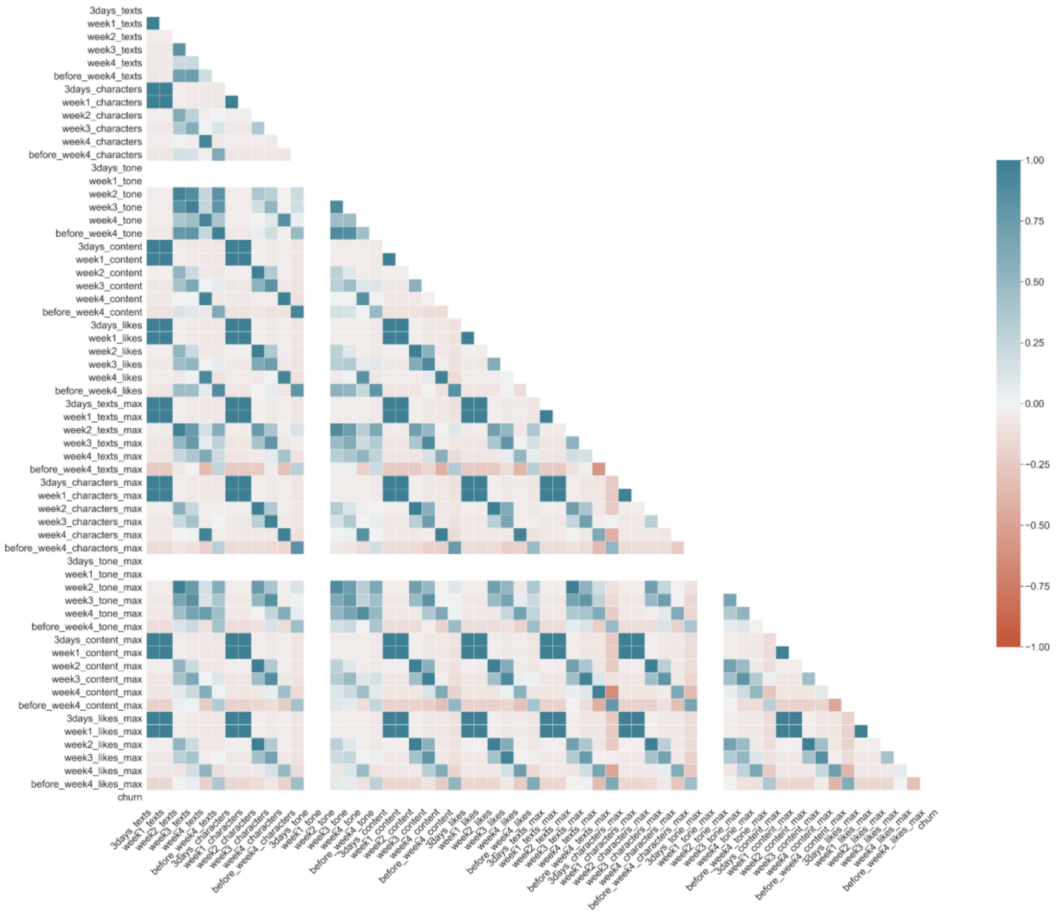
- User with subscription history: 2476
- User with both subscription history and text data: 159

- User with survey record: 490
- User with both subscription and survey: 124
- User with subscription, survey, and text: 34

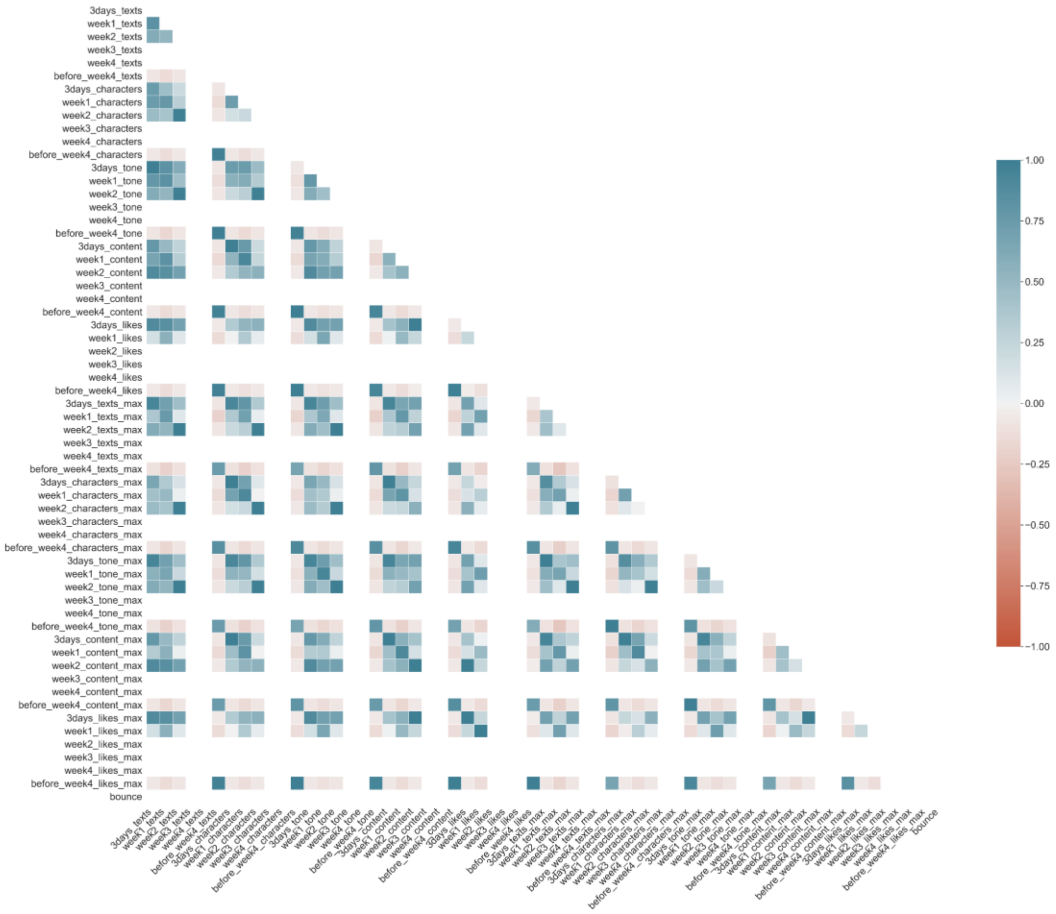
Feature correlation for active subscriber data



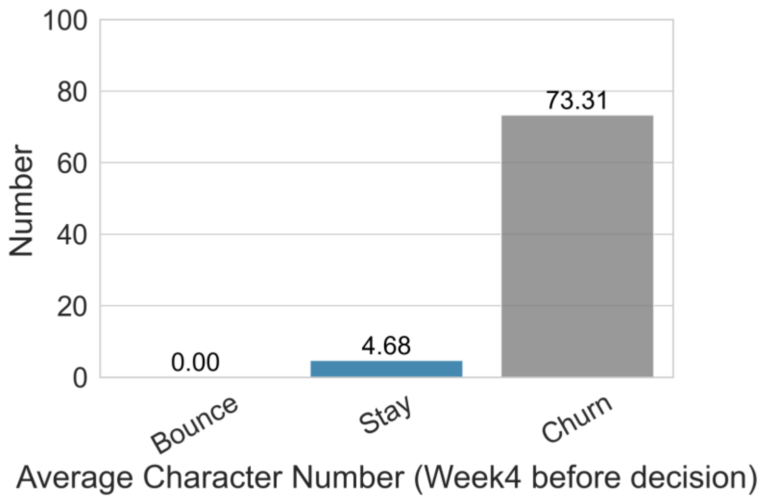
Feature correlation for churned user data

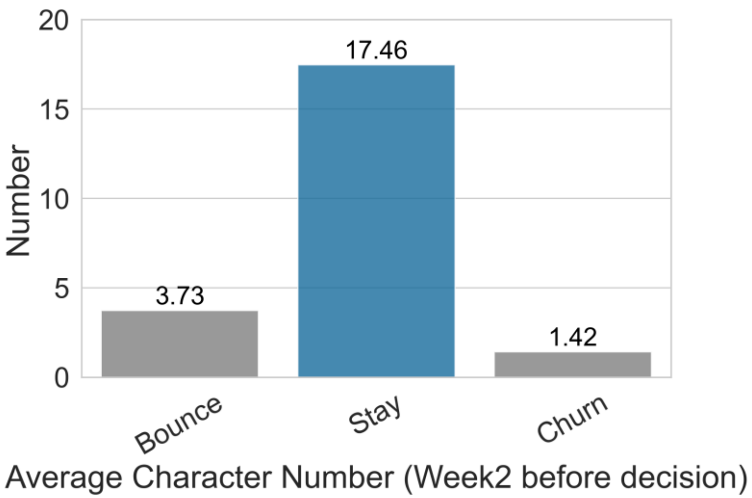


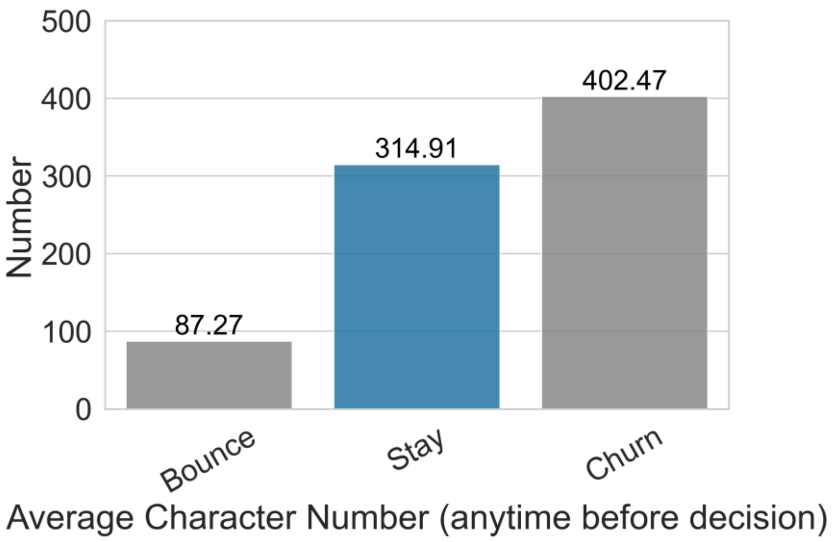
Feature correlation for bounced user data



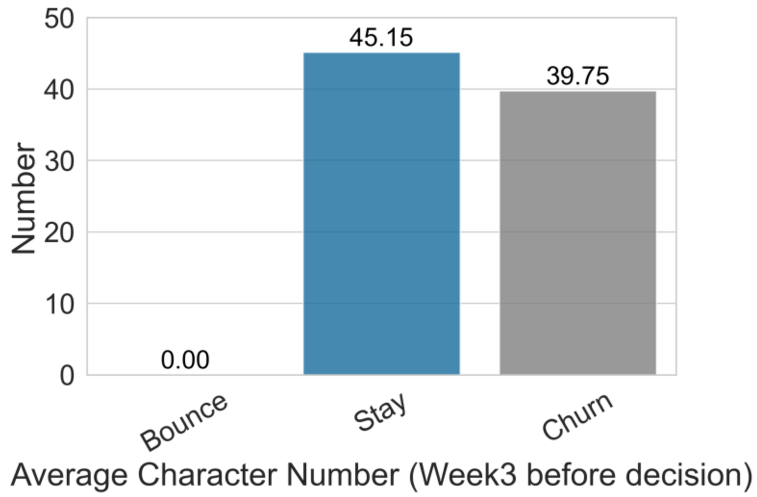
Exploratory Data Analysis

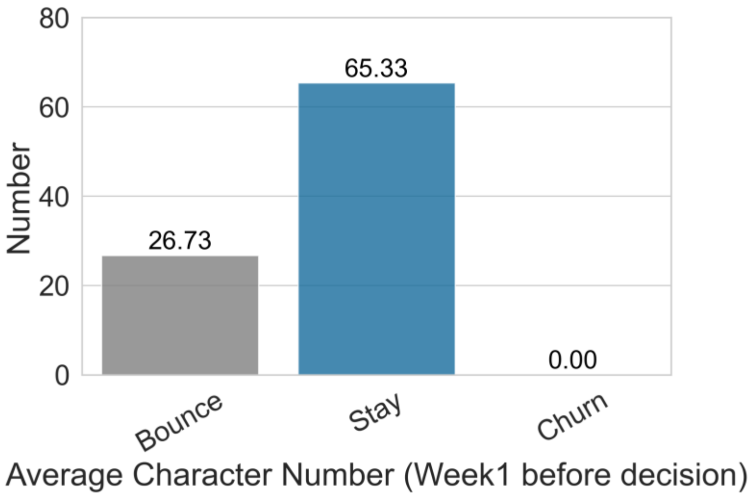




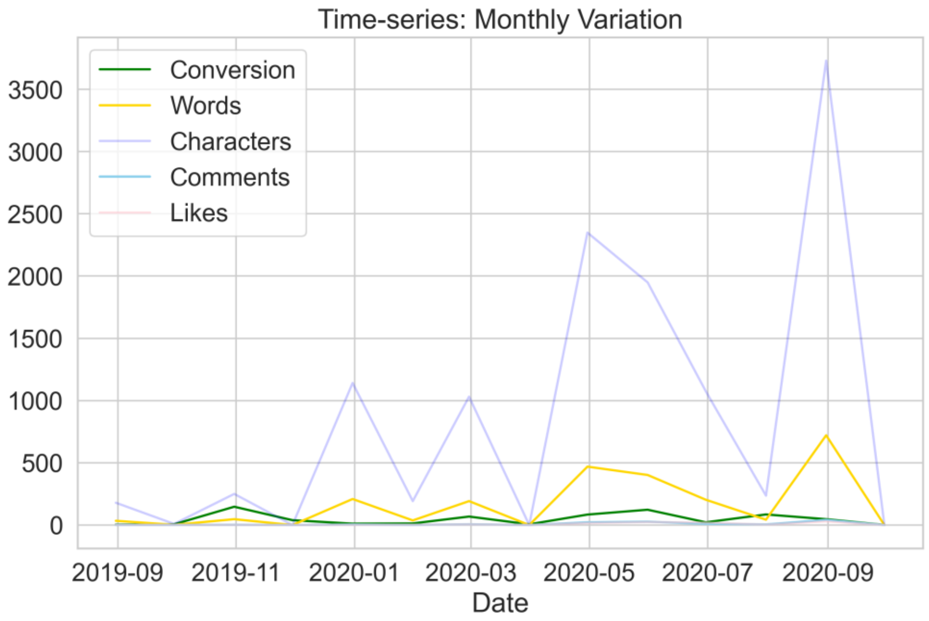


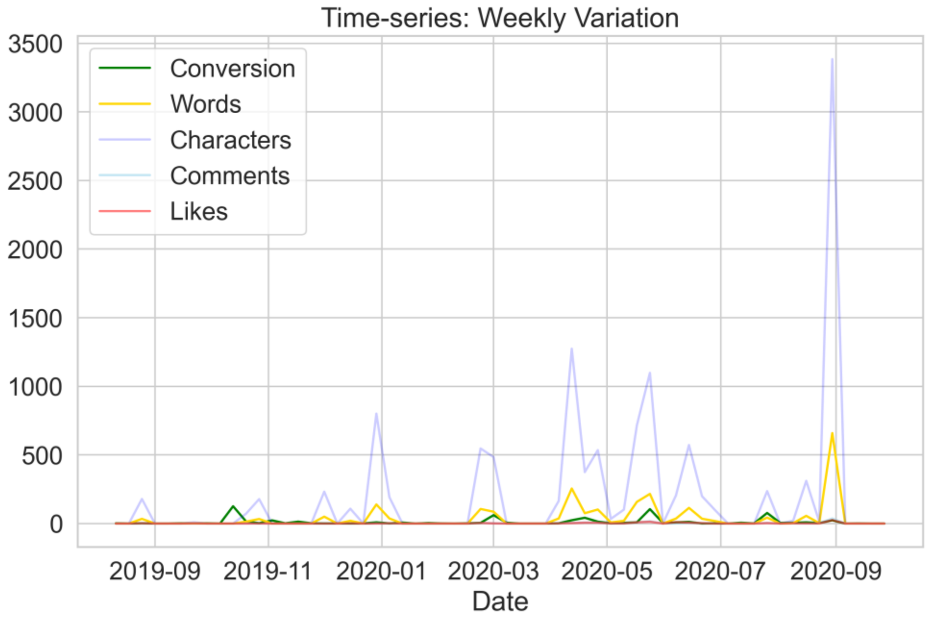
Total character number averaged by user number

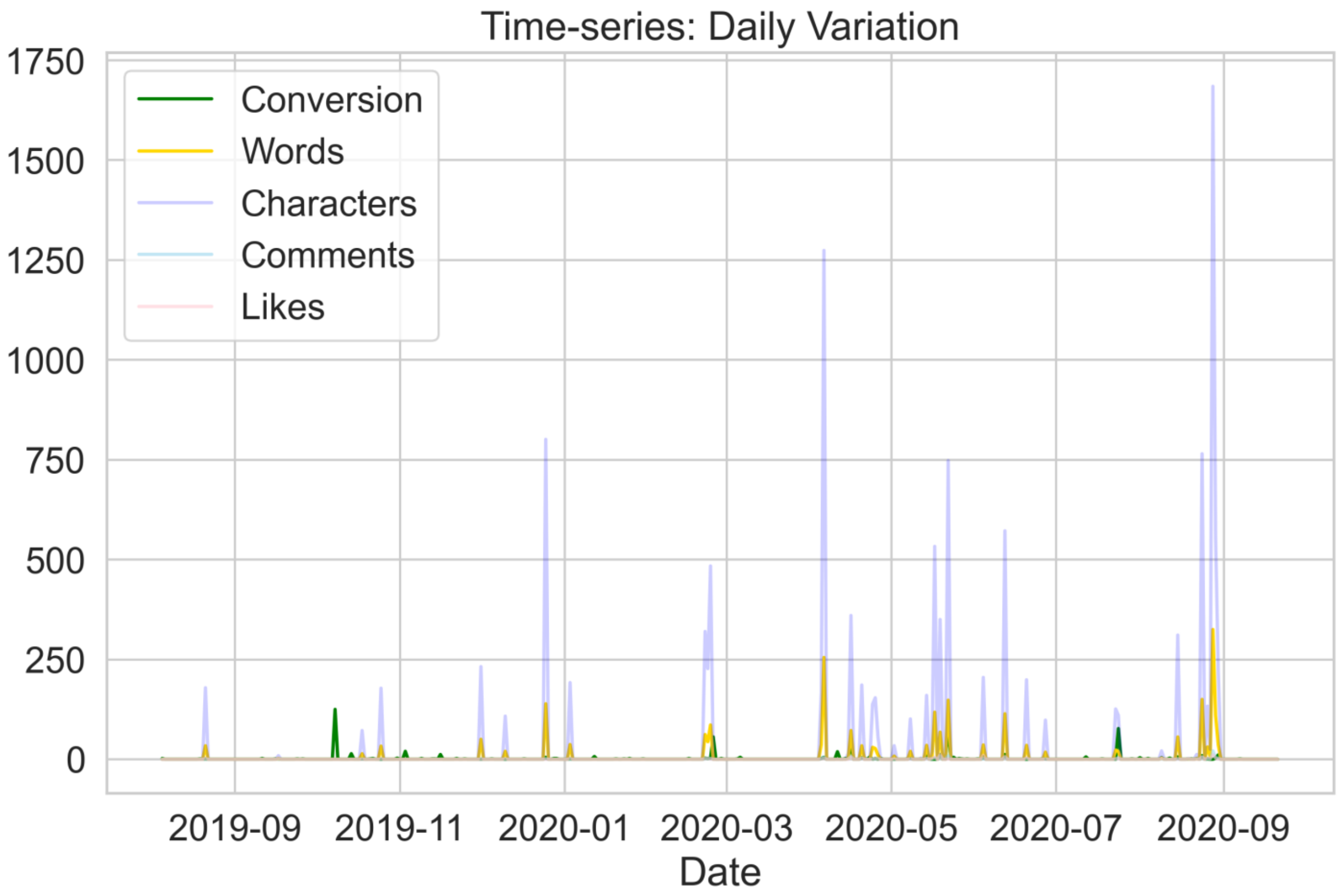




Exploratory Data Analysis







Credits

python™

TensorFlow

dmlc XGBoost

scikit learn

INSIGHT

Google AI

ktrain

GIMP

NLTK

flaticon

pycm