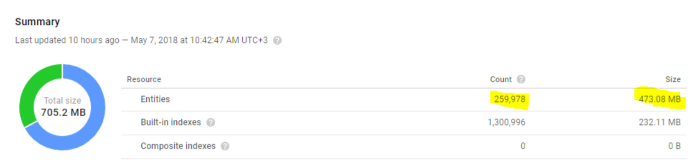
Turkish News Analytics

Selim Fırat Yılmaz
Berk Mandıracıoğlu

May 8, 2018

Bilkent University

Crawling Attempt 1



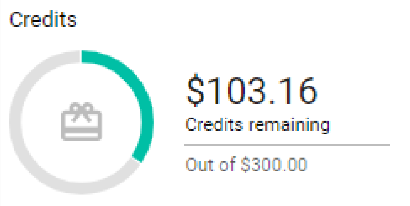
Summary

Last updated 10 hours ago — May 7, 2018 at 10:42:47 AM UTC+3

Total size 705.2 MB

Resource	Count	Size
Entities	259,978	473.08 MB
Built-in indexes	1,300,996	232.11 MB
Composite indexes	0	0 B

- Google Cloud **Free** Credits
- Start from milliyet.com and traverse the site via **BFS**
- Very technical problem (while scraping from Takvim & Milliyet)



Just in 4 days :(

Bloom Filters

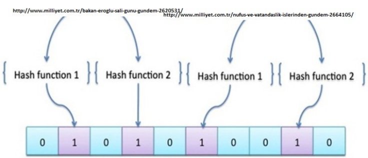
Idea: Do not crawl websites that are crawled before.

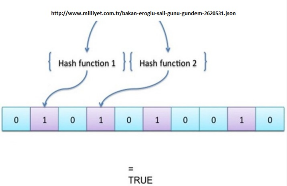
Solution 1: Store urls in a hashmap and check in O(1) time.

```
>>> str(sys.getsizeof("http://www.milliyet.com.tr/bakan-eroglu-sali-gunu-gundem-2620531.json")*200000/(1024*1024)) + " GB"
'19 GB'
```

200K urls are stored in 19 GB memory.

Solution 2: Bloom Filters: Probabilistic data structure to check membership in a set via hashing. Very useful in data streams.





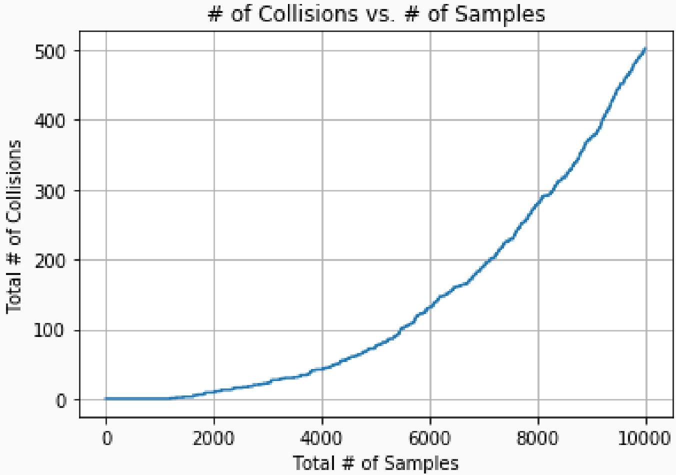
Procedure

1. Init the array of bits a, initially all 0.
2. Set 2 hash functions h1(x), h2(x).
3. ones(x) gives the the # of bits in the binary representation of x.
4. zeros(x) gives the the # of bits in the binary representation of x
5. For element el to be added to the set,
   - a[h1(zeros(el))] = 1
   - a[h2(ones(el)) = 1
6. To check whether an element might be in the set,
   - a[h1(zeros(el))] == 1 and a[h2(ones(el)) == 1
7. To check whether an element is not in the set,
   - not (a[h1(zeros(el))] == 1 and a[h2(ones(el)) == 1)

Bloom Filters Analysis

While appending 10K disting elements to 2GB bitstream.

# of collisions: False Positives

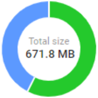


Crawling Real Time News

- Purpose: Obtain real time news and reduce complexity
- Visit **RSS** of 64 news websites **every 15 minutes** to obtain title, and description
- Continue to link of the news story to obtain content via **news-please** library
- Crawling since **Mar 13, 2018, 8:30:00 PM** using Google Cloud Compute Instance f1-micro (1 vCPU, 0.6 GB memory) [sudan ucuz]
- Crawled data is insterted into Google Cloud **Datastore** (NoSQL Document Database)

Summary

Last updated 10 hours ago – May 7, 2018 at 10:43:09 AM UTC+3



Resource	Count	Size
Entities	180,870	285.41 MB
Built-in indexes	1,946,596	386.37 MB
Composite indexes	0	0 B

Visualizing news Dataset with Kibana

- Data visualization plugin for ElasticSearch
- News data was converted to ElasticSearch indices
- These indices represent data
- Indices are used to visualise our data

General Properties of News Dataset

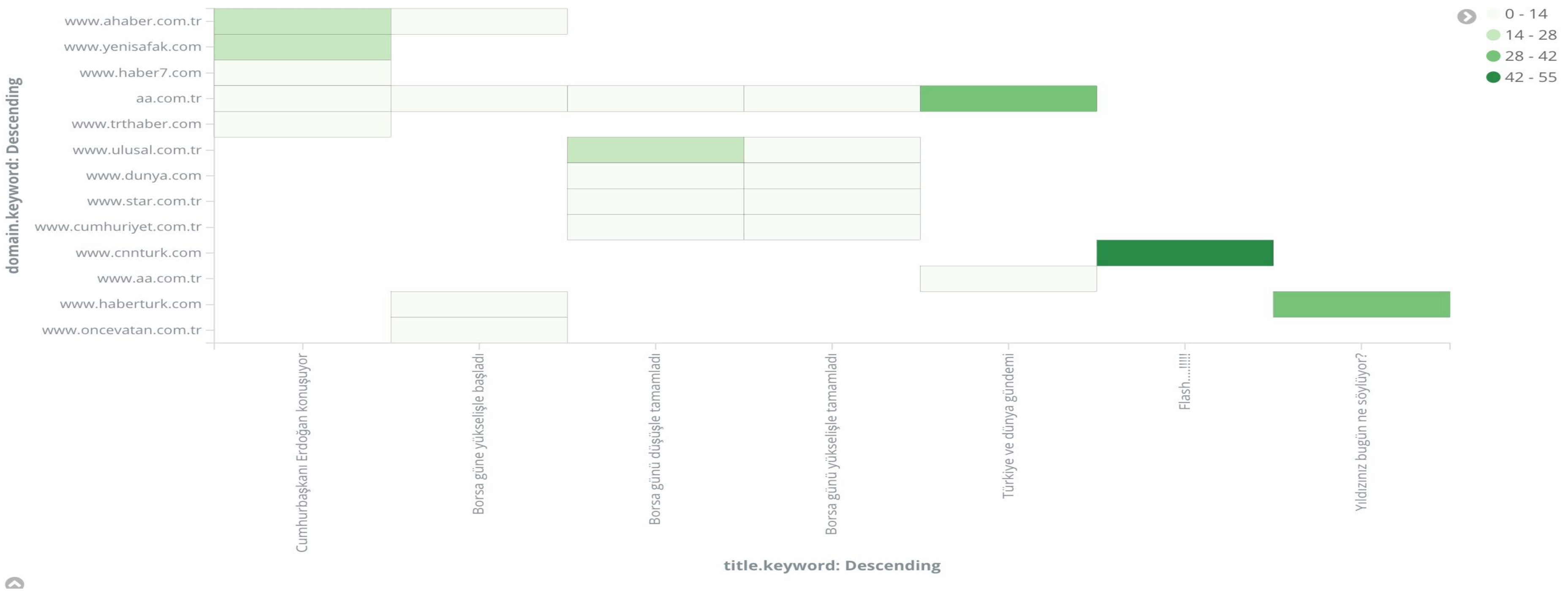
Figure: News counts over time

Figure: Domain Cloud

Figure: Word Cloud of Titles

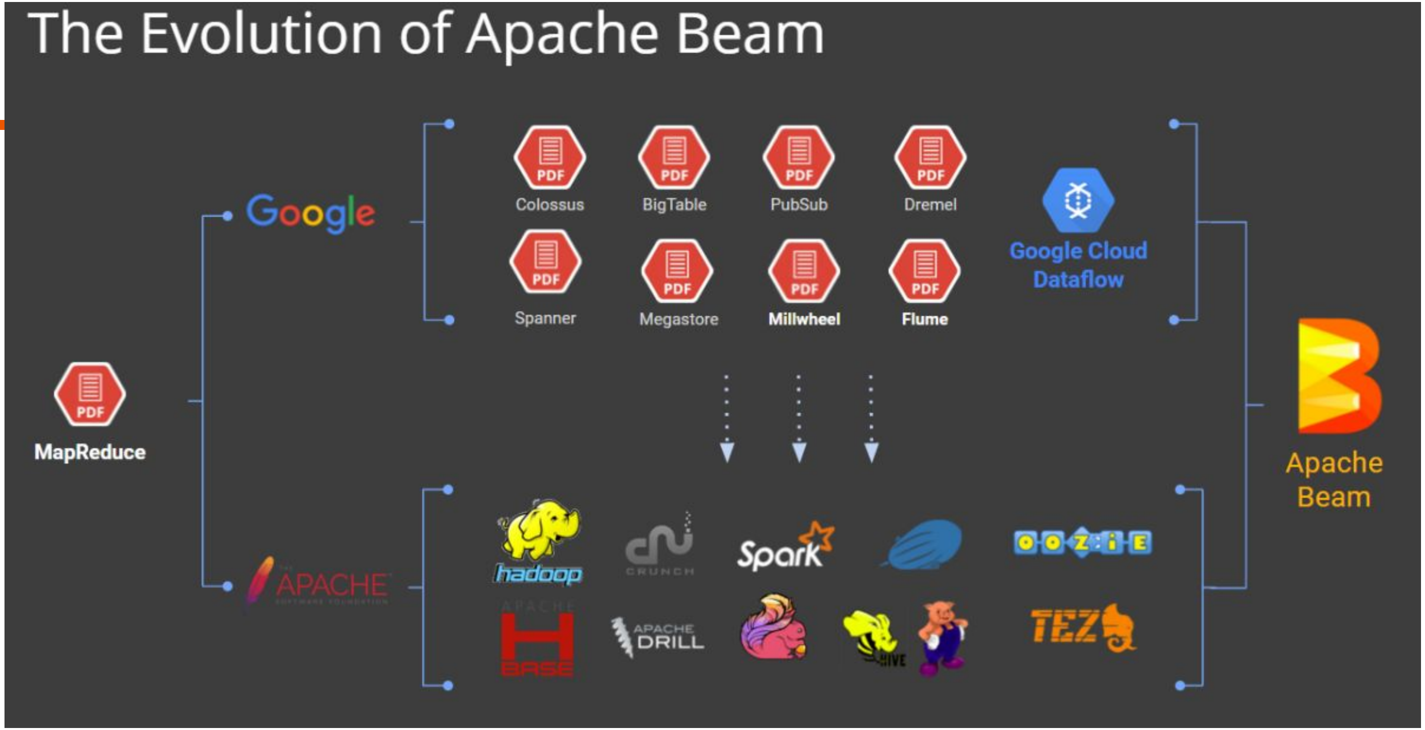
Figure: News Title Cloud

News Sites vs News Titles



- It is clear that ahaber.com.tr and yenisafak.com are biased towards Erdoğan

Apache Beam



- Bases on MapReduce paper published in 2004
- Combines features from DataFlow, Spark, Hadoop environments

General Structure of Apache Beam Pipeline

- Works with Pipelines that define dataflow
- Pipelines apply transforms(PTransform) to data
    - Similar to map and reduce stages of MapReduce
- Data are represented as PCollections which can be both input and output

Figure1: Example Pipeline

Implementing LSH with ApacheBeam - MapReduce

Mapper1

- **Input:** <key = documentId, value = line representation of news>
- **Output:** <key = documentId, value = 32-bit int shingle(of defined size) >

1) Mapper1 seperates news into shingles that are in size k, where k is given
2) Redundant shingles are discarded
3) Shinges are hashed to represent a 32-bit integer

GroupByKey

- The output of Mapper1 is grouped by keys and pipelined to Mapper2 transform

Mapper2

- **Input:** <key = documentId, value = list of shingles>
- **Output:** <key = band, value = documentId>

1) *h* number of hash functions are applied to each shingle, where *h* is given
2) For each hash function minimum value is calculated and signature row is produced
3) Signature row is separated into bands of given size *b*
4) For each band in signature row, emit(band, docId)

GroupByKey - Reduce

- The output of Mapper2 is grouped by keys and pipelined to Mapper3

Mapper3

- Mapper3 input: <key = band, value = list of documentId>
- Mapper3 output: <key = pair of documentId, value =1>

1) For each pair in value_list(list of documentIds)
   a) If pair1 > pair2, then emit([pair2,pair1], 1)
   b) Else, emit([pair1,pair2], 1)

GroupByKey - Reduce

- The output of Mapper3 is grouped by keys and final output is written

Figure: Pipeline of LSH implementation in Apache Beam

Experiments on LSH

- Plagiarism Corpus Dataset was used to experiment with LSH implementation [1]
- Students were asked to :
  - Copy and Paste (cut) ,
  - Lightly Revise(light),
  - Heavily Revise(heavy) a source,
- For cut category, they did not specify the start and end positions to copy the source

EXPERIMENT 1: # of bands = 25 and # of hash functions = 100 so rows = 4

Aim: to find *cut*

True positive = 0.3157

False positive = 0.13157

True negative = 0.86842

False negative = 0.6842105263157895

- False negative is high because start and and places of cut is not specified so not all cuts may be highly similar

EXPERIMENT 2: # of bands = 20 and # of hash functions = 100 so rows = 5

Aim: to find *cut*

True positive = 0.210

False positive = 0.039

True negative = 0.9605

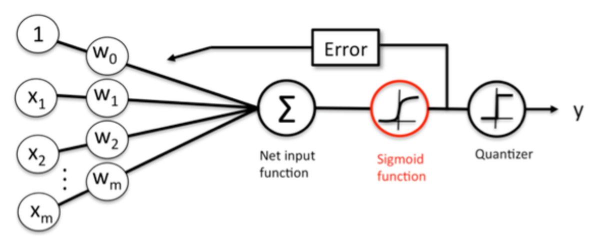
False negative = 0.7892

- False negative is high because start and and places of cut is not specified so not all cuts may be highly similar

Challenge: Turkish Language

- Morphologically very rich: İstanbullulaştıramadıklarımızdan
- High out-of-vocabulary rate (2009, Sarikaya et al.)
- Not enough data
- Not enough good tools for NLP
- **Preprocessing:** Normalization (seviyorummmm -> seviyorum)
- **Preprocessing:** Stemming (uzmanlığı/uzmanlar-> uzman)

Logistic Regression



Optimizer: Stochastic Gradient Descent

while *Not Converged* **do**
  Randomly shuffle examples in training set
  for $i = 1, \cdots, N$ **do**
    $w^+ = w - \gamma \nabla_w L(f_w(x_i, y_i))$

Some mathematics

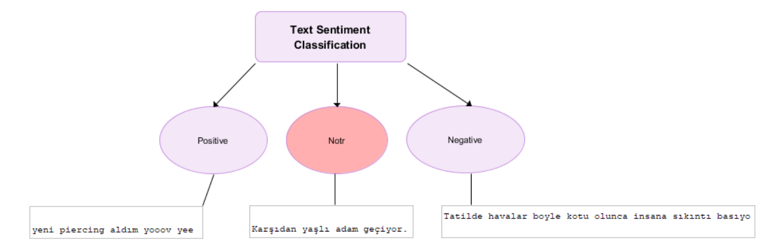
Classification Threshold Function (Quantizer)

$$f(x) = \begin{cases} NEGATIVE & 0 \leq x \leq 0.5 \\ POSITIVE & 0.5 < x \leq 1 \end{cases}$$

Sigmoid Function Derivation

$$\begin{aligned} g'_{\text{logistic}}(z) &= \frac{\partial}{\partial z}\left(\frac{1}{1+e^{-z}}\right) \\ &= \frac{e^{-z}}{(1+e^{-z})^2} (\text{chain rule}) \\ &= \frac{1+e^{-z}-1}{(1+e^{-z})^2} \\ &= \frac{1+e^{-z}}{(1+e^{-z})^2} - \left(\frac{1}{1+e^{-z}}\right)^2 \\ &= \frac{1}{(1+e^{-z})} - \left(\frac{1}{1+e^{-z}}\right)^2 \\ &= g_{\text{logistic}}(z) - g_{\text{logistic}}(z)^2 \\ &= g_{\text{logistic}}(z)(1 - g_{\text{logistic}}(z)) \end{aligned}$$

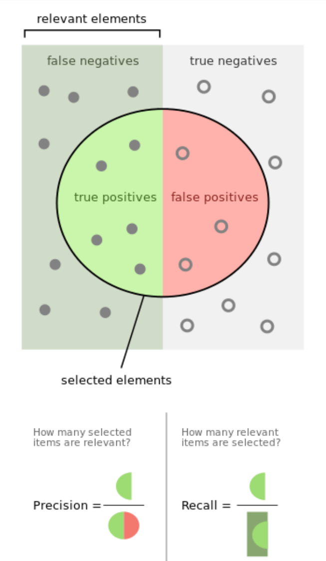
Sentiment Analysis



- Trained & validated & tested on 6k tweets dataset consisting of 3000 negative, 1552 positive, 1448 notr tweets.
- Test data is 20% of dataset.
- Validation splits are chosen as %20 of training set.
- Decision tree, logistic regression, nearest neighbor, SVM classifiers are evaluated on this dataset using **SKLearn** library.
- We chose **Logistic Regression Classifier** with SGD to implement in **numpy** matrix library.

Evaluation Metrics

$$F_1 = \frac{2}{\frac{1}{\text{recall}} + \frac{1}{\text{precision}}} = 2 \cdot \frac{\text{precision} \cdot \text{recall}}{\text{precision} + \text{recall}}.$$



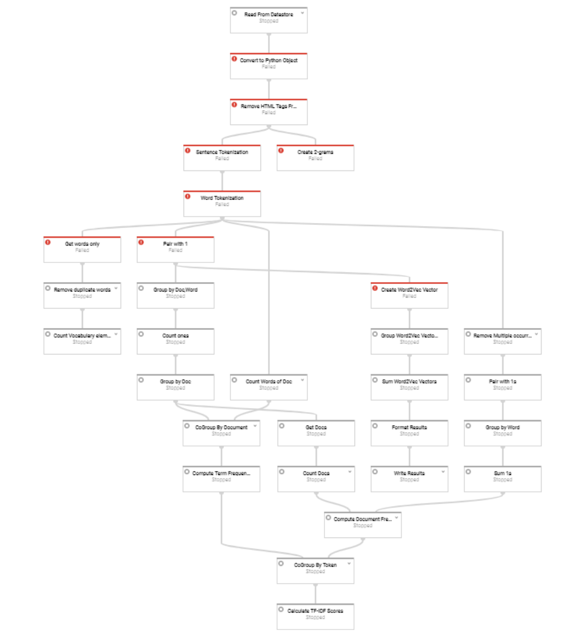
3 Class Experiments

Table 1: Some experiments on 3 class classification

Classifier	Features	Word Shingles	Normalized	Precision	Recall	**F1 Score**
Decision Tree	Bag-of-words	1	Yes	0.65	0.59	0.61
Naive Bayes	TF-IDF	1	Yes	0.46	0.46	0.45
Nearest Centroid	TF-IDF	1	No	0.62	0.56	0.58
SVM	Bag-of-words	1	Yes	0.99	0.51	0.67
Logistic Regression	Bag-of-words	1	Yes	0.71	0.63	0.66
Logistic Regression	TF-IDF	1	Yes	0.82	0.61	0.68
Logistic Regression	TF-IDF	1	No	0.82	0.62	**0.69**
Logistic Regression	Bag-of-words	2	No	0.80	0.62	**0.69**

- Notr tweets data are not so good.
- Bias on negative tweets (overfit)

TF-IDF MapReduce

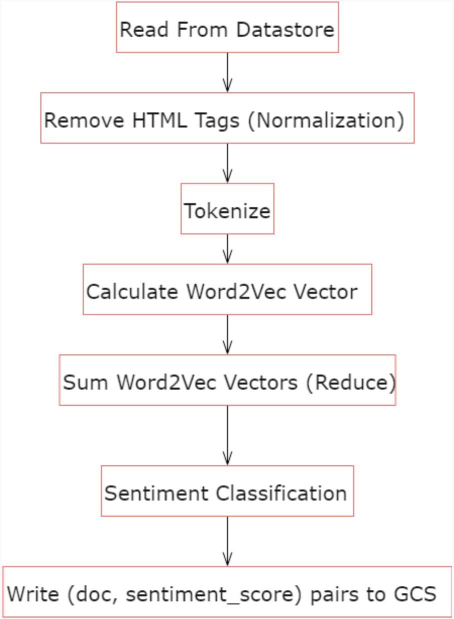


2 Class Experiments

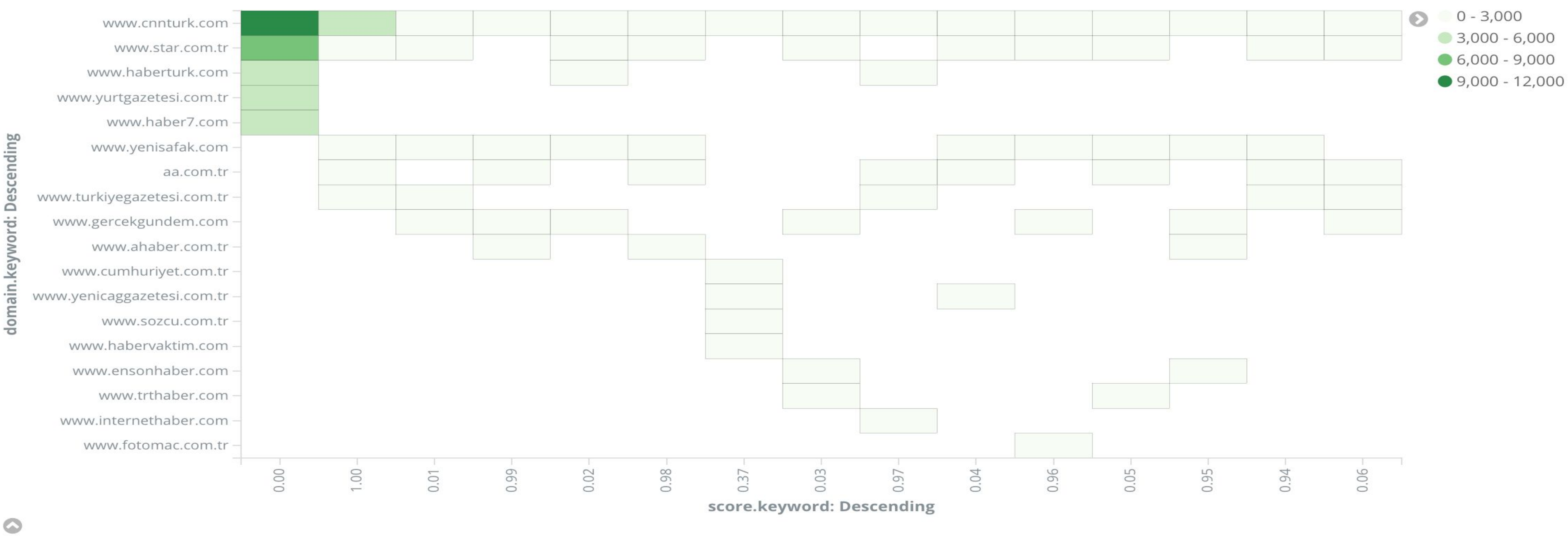
Table 2: Some experiments on 2 class classification

Classifier	Features	Normalized	Stemmed	Precision	Recall	F1
Perceptron	Word2Vec	Yes	Yes	0.79	0.77	0.78
Perceptron	Word2Vec	No	Yes	0.76	0.75	0.75
Perceptron	Word2Vec	No	No	0.76	0.75	0.75
Perceptron	TF-IDF	Yes	Yes	0.85	0.80	0.82
Perceptron	Bag-of-Words	Yes	Yes	0.85	0.83	0.83

Sentiment Classification - Apache Beam



News Sites vs Their Sentiment Scores



- It is clear that cnn.com and star.com.tr have more negative news published

Named Entity Recognition

LOCATION PERSON ORGANIZATION

Siyasetin gündeminde bir süredir erken seçim var.

Cumhurbaşkanı Tayyip Erdoğan “Erken seçim yok” dese de kulisler hareketli.

Bu iddiaları sahayı iyi tanıyan isimlerden ANAR Genel Müdürü İbrahim Uslu ile konuştum.

ANAR’ın son anketine göre Ak Parti, 1 Kasım 2015 seçiminden 3 puan geride, yüzde 46-47 seviyesinde.

Aynı seçimde yüzde 11.9 oy alan MHP yüzde 7’ye, yüzde 25 oy alan CHP yüzde 22’ye gerilemiş durumda.

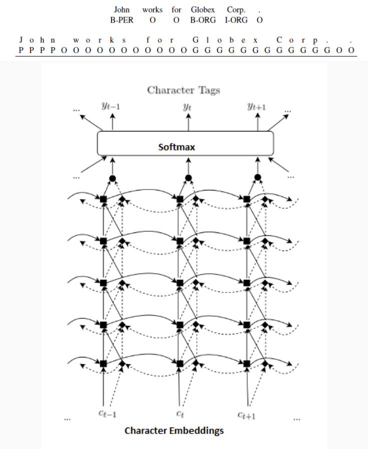
Uslu, kayıpların ağırlıklı olarak İYİ Parti’ye gittiğini düşünüyor.

Ankara bu ihtimali daha yüksek sesle konuşuyor.

- Named Entity
- Named Entity Recognition task
- Mustafa Kemal, Mustafa Kemal Caddesi
- İpek(Person), ipek(Product)

Named Entity Model

- Implemented similar model to Character Based BiLSTM (Kuru et al., 2016) via **Keras**
- Sentence -> Sequence of named entities

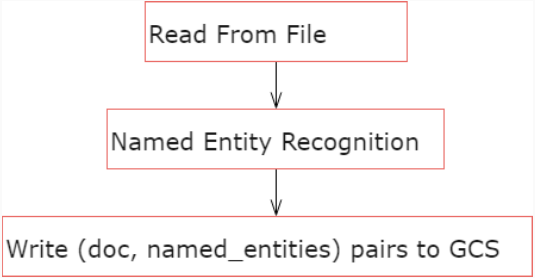


- State-of-art without gazetteers (Kuru et al., 2016)

Named Entity Dataset & Experiments

- Dataset by (Tür et al., 2003)
- 35000 sentences labeled with PERSON, ORGANIZATION, LOCATION
- 30000 training sentences
- 2237 validation sentences
- 3336 test sentences
- Training is stopped at 89.61 F1 score on validation set
- Test F1 is 82.37
- Much lower than original implementation (91.30)

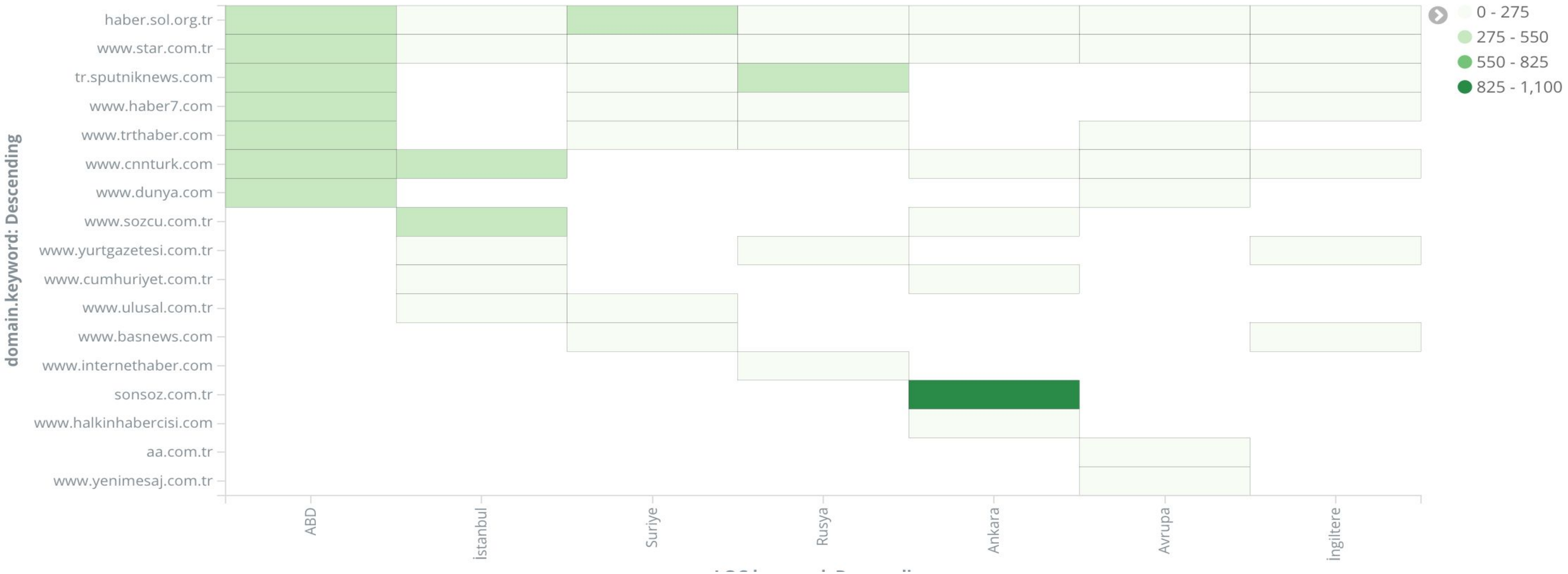
Named Entity Recognition - Apache Beam (MapReduce)



Future Works

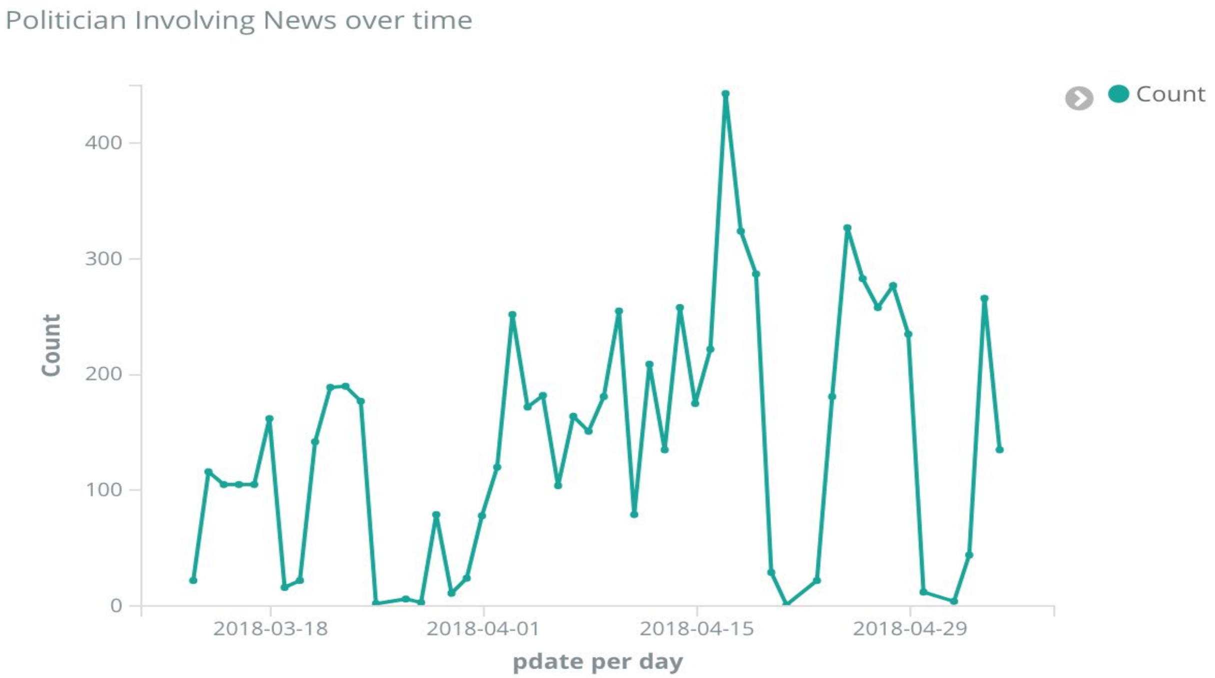
- Build architecture on Google Cloud Pub/Sub
- since both Apache Beam and Pub/Sub supports windowing for data streams
- To make it real time

News Sites vs Location Named Entity



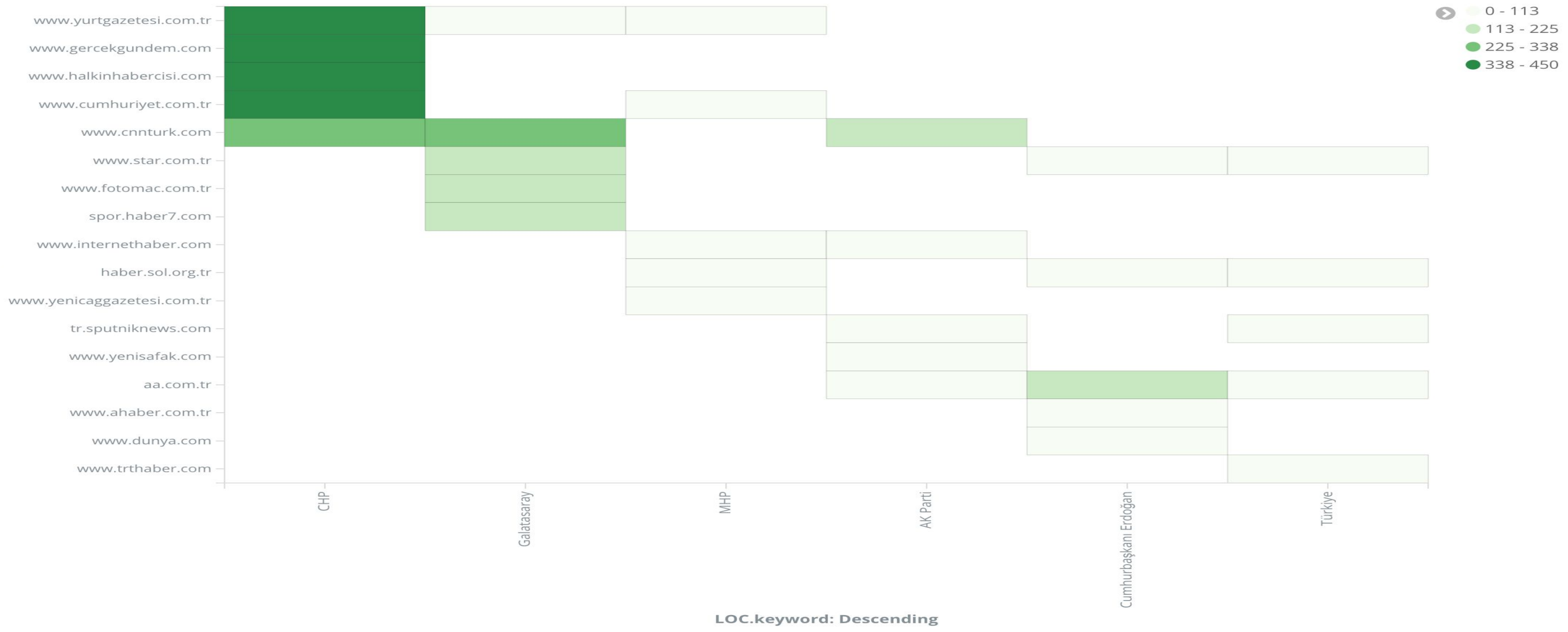
- Sonsoz.com seems to be publishing news related with Ankara location very much
- ABD dominates other locations regarding news count

Politician Related Counts and Time



- There is a peak after the decision for early election

News Sites vs Organizations



- It is clear that cumhuriyet, yurtgazetesi, gercekgundem are biased towards CHP

Named Entity Clouds

Location Cloud

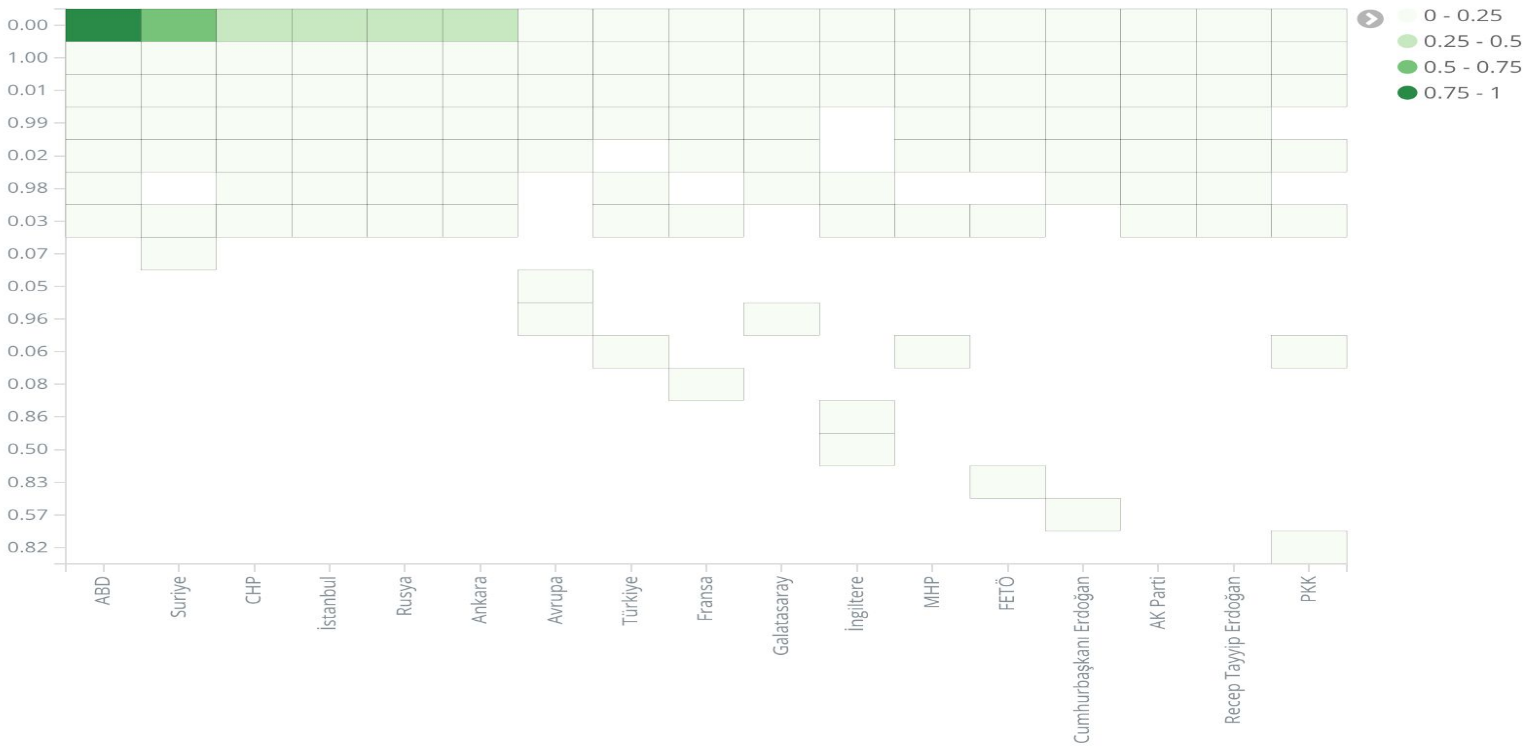
Organization Cloud

- Clear that ABD and İstanbul are popular in news
- CHP is popular among organizations

Person Cloud of News

- It is clear that Recep Tayyip Erdoğan is dominating other people among the news

Named Entity Analysis



- It is clear that ABD and Suriye are mostly in news that are negative