# PREDICTION MODELS BASED ON MAX-STEMS

## (or harnessing imbalanced data)

Episode One: One-Word Based

Ahmet Furkan EMREHAN

(matahmet@gmail.com)



## PREDICTION MODELS BASED ON MAX-STEMS

- Episode One: One-Word Based
- Episode Two: A Combinatorial Approach
- Episode Three: Effect of Hyperparameters
- Episode Four: Advanced Examinations

# INTRODUCTION

- As is seen, quantity of information is grown in a rampant manner. Correspondingly written information soars with social media apps day by day. Tweets, comments, tags give a great contribution to that bulk of written information.



# PROBLEM

- Labelling written information, sentences in practical sense, is a problem in Supervised Learning for Text Mining Literature.
- Moreover frequencies of labels are imbalanced in most cases. For example, most headlines of news in a news portal are labelled as «breaking news» or «news flash» in order to get attraction.

# MOTIVATION

- Documents (docs) in this context are sentences. Sentences are composed of ordered words. One computes frequency of a word in sentences with known label (in train set) by labels.
- Frequency of words can give an idea about label of sentences in which they are. My models in this study are based on that approach.
- A set of solutions for those problems (labelling and imbalanced data) is proposed in this study.
- This study is aimed to be a contribution to Supervised Learning Literature as a bunch of Prediction models for Text Mining.



# METHOD (Word to Stem)

- Using words for prediction of a sentence entails an approach based on structure of relevant language. This study focuses on the agglutinative language (ex. Turkish, Hungarian, Estonian, Basque, Japanese, Korean etc.)
- Naturally,in agglutinative language, stem of a word is core part to create «meaning». In most cases, word is in form of stem with derivational or/and inflectional affixes (morphemes).
- But to use word for computing frequencies may not be efficient on account of specific derivational and inflectional forms of word.
- For this reason, to use stem is more convenient than to use word because the stem involves meaning or concept which word bear in pure form (without fixes).

# METHOD (Stem to Max-Stem)

- As length of a stem decreases, its meaning scope of the stem expands semantically. Stem may involve broad which goes over the limit of scope of word.
- In such cases, to choose derivational form of the stem with maximum length but which the word includes fits for purpose in terms of reasonably marking off scope of meaning of the word.
- That approach is extended to whole cases in order to guarantee saving the meaning of the word. (for more discussion: Step1_turkish_stems_ReadMe.txt)



# COMPONENTS OF MODELS

- $p$: $index\ of\ categories\ (or\ labels)$
- $Label^p$: $category\ wit\quad p\ index$
- $n$: $counts\ of\ categories\ (or\ labels)$
- $doc_i$: $document, in\ test\ set, wit\quad index\ i\ as\ a\ sentences\ or\ just\ a\quad eadline$
- $stem_{ij}$: $stem\ wit\quad index\ j\ of\ doc_i$
  ($stem\ can\ be\ c\quad osen\ as\ max\quad stem\ mentioned\ previous\ slides.$)
- $m_i$: $counts\ of\ stem_{ij}$
- $\Sigma^p$: $counts\ of\ documents\ labelled\ wit\quad category\ wit\quad index\ p\ in\ train\ set$
- $\Sigma^p_{ij}$: $counts\ of\ documents,\quad w\quad ic\quad include\ stem_{ij},\ labelled\ wit\quad category\ wit\quad index\ p\ in\ train\ set$

# COMPONENTS OF MODELS

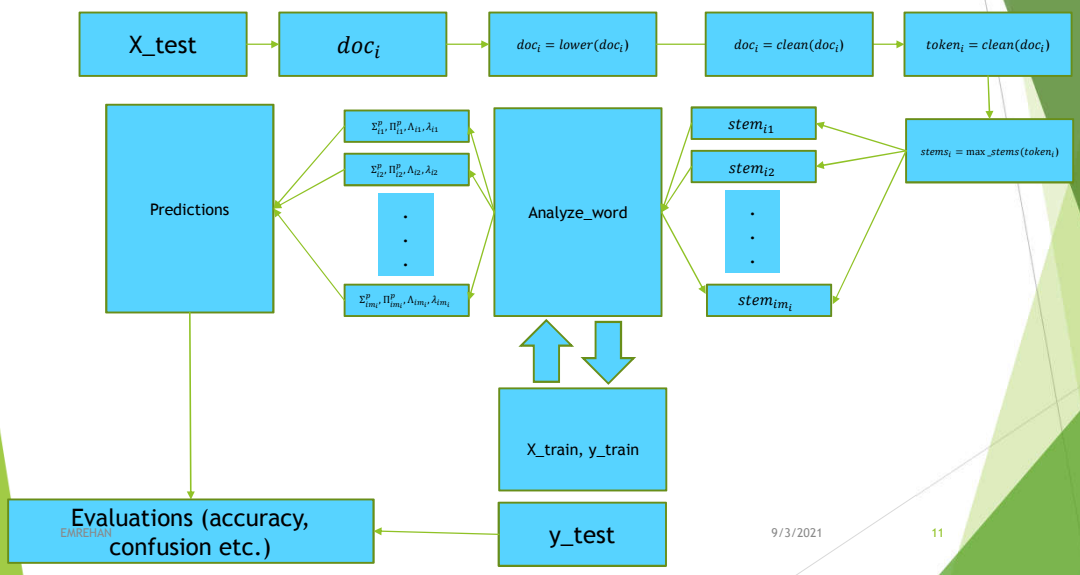
- $\Lambda_{ij} := Label^{q}\ w\ \ ere\ q = \arg\max_{p} \Sigma_{ij}^{p}$
- $\Lambda_{i}^{p}$: counts of $\Lambda_{ij}\ w\ \ ic\ \ equals\ to\ Label^{p}$
- $\lambda_{ij}: lengt\ \ of stem_{ij}$
- $\rho_{i}^{p}\quad \frac{\sum_{j=1}^{m_i} \Sigma_{ij}^{p}}{\Sigma^{p}}$ *
- $in\ case\ t\ \ at\ \Sigma^{p}=0, \rho_{i}^{p} := 0$
- $\Pi_{ij}^{p}\quad \frac{\Sigma_{ij}^{p}}{\sum_{q=1}^{n} \Sigma_{ij}^{q}}$
  ($it\ can\ be\ considered\ as\ probability\ of\ stem_{ij}\ labelled\ wit\ \ category\ wit\ \ p\ index$)



# COMPONENTS OF MODELS

- $\overline{\Pi_{i}^{p}} := average_{j}\ \left(\Pi_{ij}^{p}\right) suc\ \ t\ \ at\ all$ "j "s $meet\ t\ \ e\ condition\ \Pi_{ij}^{p} > 0$
  in case that $\Sigma_{ij}^{p} = 0\ for\ all\ p = 1,2, \dots n, \overline{\Pi_{i}^{p}} = 0$
- $\widehat{\Pi_{i}^{p}}\quad \max_{j}(\Pi_{ij}^{p})$

# General Scheme for Prediction Models






# Model 1

- $predict_1(doc_i) = \begin{cases} Label^q & q\ suc\ \ t\ \ at\ \ if\ \Sigma^p_{ij} = 0 < \Sigma^q_{ij}\ for\ all\ p \neq q\ \ and\ for\ all\ \ j = 1, \dots, m_i \\ q = argmax_{c(r,j)}\ \sum_j \Sigma^r_{ij} & ot\ \ erwise \end{cases}$
- $^*c(r,j) \quad (r = \arg 2nd\ max_p\ \Lambda^p_i)\ and\ (\Lambda_{ij} = Label^r)$ (Because $\arg 2nd\ max_p\ \Lambda^p_i$ may not be unique.)

# Model 1

- $predict_1(doc_i) = \begin{Bmatrix} Label^q & if\ \Sigma_{ij}^p = 0 < \Sigma_{ij}^q\ for\ all\ p \neq q \\ Label^q\ w\ ere\ q = \arg 2nd\ max_p\ \Lambda_i^p ot\ erwise \end{Bmatrix}$

*in case t at q is not uniqe, q is c osen as t e minimum index meeting t e condition*

# Model 2

- $predict_2(doc_i) = Label^q\ w\ ere\ q = \arg max_p\ \rho_i^p$

# Model 3

- $predict_3(doc_i) = Label^q \; w \;\; ere \; q = \arg max_p \overline{\Pi_i^p}$



# Model 4

- $predict_4(doc_i) = \begin{Bmatrix} Label^q & w \;\; ere \;\; q = \arg max_p \widehat{\Pi_i^p} \;\; if \; q \; is \; unique \\ q = \arg max_r \widehat{\Pi_i^r} \;\; and \; r = \arg max_p \Sigma_{ij}^p \; ot \;\; erwise \end{Bmatrix}$

# Model 5

- $predict_5(doc_i) = Label^q \; w \;\; ere \; q = \arg max_p \left(max_j \left(\lambda_{ij} \;\; \frac{\Sigma_{ij}^p}{\Sigma^p}\right)\right)$



# Case «No Prediction»

- No stem of a document may not be included by any document in train set, in some cases. Trivially prediction functions generate label as «No Prediction». This probability is nearly zero if size of train set is sufficiently large.
- However there is a higher probability of label «No Prediction» in model having Combinatorial Approach in the study. Because probability of that all elements of a combination (a bunch of stems in a document in test set) are in same document (in train set) is obviously lower than probability of a stem (involved by document in test set) in a document (in train set) .
- Some examples of that case is observed in Episode two.

# Case «Not Unique»

- In some cases, values generating predictions, like "$\arg max_p \widehat{\Pi_i^p}$" $and$ "$\arg 2nd\ max_p\ \Lambda_i^p$" , may not be unique because of equal values. Then models choose label indexed with minimum argument as a prediction corresponding list structure in Python.
- I use extra parameters (figuratively considered as tiebreaker), $\Sigma_{ij}^p$ and $\sum_j \Sigma_{ij}^r$ , on the purpose of avoiding that case.
- Moreover as train set gets large, probability of exitence of equal values is expected to diminish.



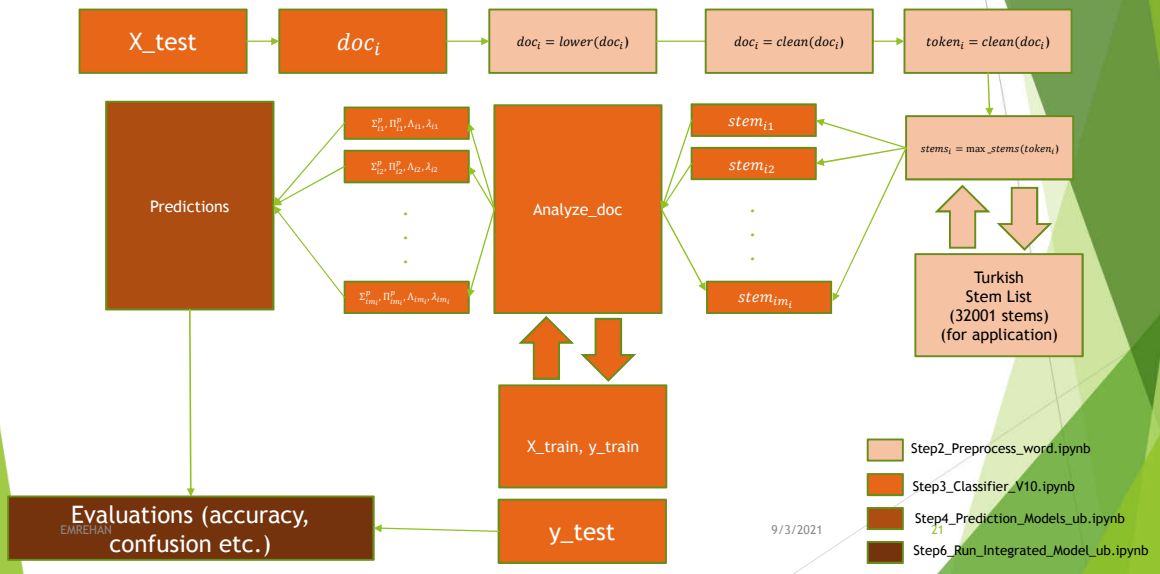
# Application (introduction)

- We use data of «nayn.co» a news portal in Turkish Language. Data is imported by url «"https://raw.githubusercontent.com/naynco/nayn.data/master/classification_clean.csv"».
- Head of data is presented below

| | Title | Categories |
|---|---|---|
| 12006 | 58 Saniyede Katar Meselesi? Katar krizi nedir?... | DÜNYA |
| 12496 | 58 Saniyede Türkiye - Almanya Gerginliği | DÜNYA |
| 12877 | Adriana Lima, Bomba Aşkla İlgili İlk Kez Konuş... | DÜNYA |
| 12878 | Galatasaraylı Taraftarlar Patladı: İstifa Edin | SPOR |
| 12880 | Galatasaray'dan Ayrılan Sabri, Neredeyse Bedav... | SPOR |

There are 11622 documents («Title» column) with label («DÜNYA» (World), «SPOR» (Sports), «SANAT»(Art) and «Teknoloji»(Technology)). But data is imbalanced in favor of category «DÜNYA» such that the counts [and percentages] of categories 9226[%79] ,1967 [%17],285 [%2] and 144 [%1] respectively.

# General Scheme for Application of Prediction Models






# Application (computations)

- Pandas and Sklearn libraries in Python is used for application of methods. Test size is chosen as 0.2 and random_state parameter for partition as 57.
- Values of parameters in model are computed below
- Counts of categories: $n = 4$
- Indexes and name of categories: $p = 1,2,3\ and\ 4\ ,\ Label^p =$ "DÜNYA","SPOR","SANAT and "Teknoloji", $respectively$
- Counts of categories in train set : $\Sigma^1 = 7384, \Sigma^2 = 1568, \Sigma^3 = 229\ and\ \Sigma^4 = 116$

# Application (computations)

- Now Let's show an example and compute its parameters (or compounds of models). We deal with document with index number $i = 38296$ , $rank\ (in\ test\ set) = 1356$ (index number may not be related to rank)
- $doc_i$: 2 *kedi 2 yıldır sanat müzesine girmeye çalışıyor*
  ($en$: 2 *cats try to enter art museum for 2 years*)
- $label_i$: *DÜNYA* ($en$: *world*)
- $stems$: ['*kedi*','*yıldır*','*sanat*','*müze*','*gir*','*çalı*']
- Output of analyze_doc(doc,X_train,y_train): $j = 1, \dots, m_i = 6$

```
[['kedi', 4, [15, 2, 0, 0], [0.88, 0.12, 0.0, 0.0], 'DÜNYA', 'SPOR'],
 ['yıldır', 6, [15, 55, 1, 0], [0.21, 0.77, 0.01, 0.0], 'SPOR', 'DÜNYA'],
 ['sanat', 5, [6, 0, 8, 0], [0.43, 0.0, 0.57, 0.0], 'SANAT', 'DÜNYA'],
 ['müze', 4, [9, 1, 7, 0], [0.53, 0.06, 0.41, 0.0], 'DÜNYA', 'SANAT'],
 ['gir', 3, [110, 16, 6, 4], [0.81, 0.12, 0.04, 0.03], 'DÜNYA', 'SPOR'],
 ['çalı', 4, [156, 25, 7, 4], [0.81, 0.13, 0.04, 0.02], 'DÜNYA', 'SPOR']]
```

$stem_{ij}$ | $\lambda_{ij}$ | $\Sigma_{ij}^{p}$ | $\Pi_{ij}^{p}$ | $\Lambda_{ij}$ | Category of 2nd max $\Sigma_{ij}^{p}$ (not used)



# Application (computations)

- $i = 38296$
- $stems$: $stem_{i1}$ = "*kedi*", $stem_{i2}$ = "yıldır" , $stem_{i3}$ = "sanat" , $stem_{i4}$ = "müze", $stem_{i5}$ = "gir" , $stem_{i6}$ = "çalı"
- $lengt\ \ of\ stems : \lambda_{i1} = 4, \lambda_{i2} = 6, \lambda_{i3} = 5, \lambda_{i4} = 4, \lambda_{i5} = 3, \lambda_{i6} = 4$
- $\Sigma_{ij}^{p}$: *counts of documents, w ic include* $stem_{ij}$, *labelled wit category wit index p in train set*
- $for\ j = 1\ and\ j = 6 : \Sigma_{i1}^{1} = 15, \Sigma_{i1}^{2} = 2, \Sigma_{i1}^{3} = 0, \Sigma_{i1}^{4} = 0 , \Sigma_{i6}^{1} = 156, \Sigma_{i6}^{2} = 25, \Sigma_{i6}^{3} = 7, \Sigma_{i6}^{4} = 4$
- $\Lambda_{ij} := Label^{q}\ w\ \ ere\ q = \arg\max_{p} \Sigma_{ij}^{p}$ : $\Lambda_{i1}$ = "DÜNYA", $\Lambda_{i2}$ = "SPOR", $\Lambda_{i3}$ = "SANAT"
  $\Lambda_{i4}$="DÜNYA", $\Lambda_{i5}$ = "DÜNYA", $\Lambda_{i6}$="DÜNYA"
- $\Lambda_{i}^{p}$: counts of $\Lambda_{ij}$ *w ic equals to* $Label^{p}$, $\Lambda_{i}^{1} = 4, \Lambda_{i}^{2} = 1, \Lambda_{i}^{3} = 1, \Lambda_{i}^{4} = 0$
- $\Pi_{ij}^{p}$ for j = 2 and j = 4 : $\Pi_{i2}^{1} = 0.21, \Pi_{i2}^{2} = 0.77, \Pi_{i2}^{3} = 0.01, \Pi_{i2}^{4} = 0, \Pi_{i4}^{1} = 0.53, \Pi_{i4}^{2} = 0.06, \Pi_{i4}^{3} = 0.41, \Pi_{i4}^{4} = 0$

# Application (computations)

- $\rho_i^p$: $\rho_i^1 = \frac{311}{7384} = 0.042$, $\rho_i^2 = \frac{99}{1568} = 0.063, \rho_i^3 = \frac{29}{229} = 0.127$, $\rho_i^4 = \frac{8}{116} = 0.069$
- $\overline{\Pi_i^p}$: $\overline{\Pi_i^1} = \frac{0.88+0.21+0.43+0.53+0.81+0.81}{6} = 0.612$, $\overline{\Pi_i^2} = \frac{0.12+0.77+0.0+0.12+0.13}{5} = 0.24$

$$\overline{\Pi_i^3} = \frac{0.01 + 0.57 + 0.41 + 0.04 + 0.04}{5} = 0.214, \overline{\Pi_i^4} = \frac{0.03 + 0.02}{2} = 0.0.025$$

- $\widehat{\Pi_i^p}$: $\widehat{\Pi_i^1} = 0.88$, $\widehat{\Pi_i^2} = 0.77$, $\widehat{\Pi_i^3} = 0.57$, $\widehat{\Pi_i^4} = 0.03$



# Application (computations)

- Some Notes:

Algorithm to find stem of word is not be said to work perfectly due to morphological nature of Turkish language:

word: yıldır[….for a year] → stem: yıl[year] but algorithm gives: yıldır(mak)[(to)discourage]

word: çalışıyor [(They) try to ] → stem: çalış(mak)[(to) try (to do something)] **but algorithm gives: çalı [bush]**

But it is reasonably well:

word: müzesine [to museum] → stem: müze [museum]

word: girmeye [for the purpose of entering] → stem: gir(mek) [(to) enter]

The reason of imperfect cases is turkish stem list which algorithm uses. Because excluding derivational forms in turkish stem list may give rise to losing of true stem:

for example çalışıyor → çalış(mak) (true stem but in derivational form then excluded) → çal(mak) (original stem but not related modern meaning of çalış(mak).Among these structures, algorithm gives «çalı», having different meaning but covered by «çalış(mak)». However it is not big deal that is why nearly all documents including «çalı» related to «çalış(mak)», because «çalı» is not popular word in modern turkish.

This morphological problem in this point is related to computing «larger meaning scope than it should be» , not «narrower than it should be».

# Application (prediction)

- $for\ i = 38296$
- $predict_1(doc_i) = "SPOR"$
- $predict_2(doc_i) = "SANAT"$
- $predict_3(doc_i) = "DÜNYA"$
- $predict_4(doc_i) = "DÜNYA"$
- $predict_5(doc_i) = "SPOR"$



# Application (results)

| | | Confusion Matrix for Model 1 (count) | | | | | Confusion Matrix for Model 1 (percentage) | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | | Prediction | | | | | Prediction (rounded to 2 digits) | | | | |
| | | DÜNYA | SANAT | SPOR | Teknoloji | Total | DÜNYA | SANAT | SPOR | Teknoloji | Total |
| Observed | DÜNYA | 1509 | 20 | 302 | 11 | 1842 | 0.82 | 0.01 | 0.16 | 0.01 | 1 |
| | SANAT | 21 | 21 | 14 | 0 | 56 | 0.38 | 0.38 | 0.25 | 0 | 1 |
| | SPOR | 86 | 0 | 312 | 1 | 399 | 0.22 | 0 | 0.78 | 0 | 1 |
| | Teknoloji | 25 | 0 | 2 | 1 | 28 | 0.89 | 0 | 0.07 | 0.04 | 1 |
| Accuracy Rate For Model 1 | | | | | | | | | | | |
| 0.79 | | | | | | | | | | | |

| | | Confusion Matrix for Model 2 (count) | | | | | Confusion Matrix for Model 2 (percentage) | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | | Prediction | | | | | Prediction (rounded to 2 digits) | | | | |
| | | DÜNYA | SANAT | SPOR | Teknoloji | Total | DÜNYA | SANAT | SPOR | Teknoloji | Total |
| Observed | DÜNYA | 1189 | 196 | 165 | 292 | 1842 | 0.65 | 0.11 | 0.09 | 0.16 | 1 |
| | SANAT | 5 | 39 | 7 | 5 | 56 | 0.09 | 0.7 | 0.13 | 0.09 | 1 |
| | SPOR | 41 | 26 | 319 | 13 | 399 | 0.1 | 0.07 | 0.8 | 0.03 | 1 |
| | Teknoloji | 6 | 3 | 2 | 17 | 28 | 0.21 | 0.11 | 0.07 | 0.61 | 1 |
| Accuracy Rate For Model 2 | | | | | | | | | | | |
| 0.67 | | | | | | | | | | | |

# Application (results)

| | | Confusion Matrix for Model 3 (count) | | | | | Confusion Matrix for Model 3 (percentage) | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | | Prediction | | | | | Prediction (rounded to 2 digits) | | | | |
| | | DÜNYA | SANAT | SPOR | Teknoloji | Total | DÜNYA | SANAT | SPOR | Teknoloji | Total |
| Observed | DÜNYA | 1838 | 1 | 2 | 1 | 1842 | 1 | 0 | 0 | 0 | 1 |
| | SANAT | 55 | 0 | 1 | 0 | 56 | 0.98 | 0 | 0.02 | 0 | 1 |
| | SPOR | 291 | 1 | 107 | 0 | 399 | 0.73 | 0 | 0.27 | 0 | 1 |
| | Teknoloji | 28 | 0 | 0 | 0 | 28 | 1 | 0 | 0 | 0 | 1 |
| Accuracy Rate For Model 3 | | | | | | | | | | | |
| 0.84 | | | | | | | | | | | |

| | | Confusion Matrix for Model 4 (count) | | | | | Confusion Matrix for Model 4 (percentage) | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | | Prediction | | | | | Prediction (rounded to 2 digits) | | | | |
| | | DÜNYA | SANAT | SPOR | Teknoloji | Total | DÜNYA | SANAT | SPOR | Teknoloji | Total |
| Observed | DÜNYA | 1824 | 0 | 16 | 2 | 1842 | 0.99 | 0 | 0.01 | 0 | 1 |
| | SANAT | 44 | 9 | 3 | 0 | 56 | 0.79 | 0.16 | 0.05 | 0 | 1 |
| | SPOR | 153 | 1 | 245 | 0 | 399 | 0.38 | 0 | 0.61 | 0 | 1 |
| | Teknoloji | 28 | 0 | 0 | 0 | 28 | 1 | 0 | 0 | 0 | 1 |
| Accuracy Rate For Model 4 | | | | | | | | | | | |
| 0.89 | | | | | | | | | | | |



# Application (results)

| | | Confusion Matrix for Model 5 (count) | | | | | Confusion Matrix for Model 5 (percentage) | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | | Prediction | | | | | Prediction (rounded to 2 digits) | | | | |
| | | DÜNYA | SANAT | SPOR | Teknoloji | Total | DÜNYA | SANAT | SPOR | Teknoloji | Total |
| Observed | DÜNYA | 963 | 241 | 243 | 395 | 1842 | 0.52 | 0.13 | 0.13 | 0.21 | 1 |
| | SANAT | 8 | 33 | 7 | 8 | 56 | 0.14 | 0.59 | 0.13 | 0.14 | 1 |
| | SPOR | 56 | 33 | 275 | 35 | 399 | 0.14 | 0.08 | 0.69 | 0.09 | 1 |
| | Teknoloji | 6 | 3 | 3 | 16 | 28 | 0.21 | 0.11 | 0.11 | 0.57 | 1 |
| Accuracy Rate For Model 3 | | | | | | | | | | | |
| 0.55 | | | | | | | | | | | |

*End of Episode One*