



Application of Natural Language Processing on Electronic Logbooks (Elog)

Jennefer Maldonado

Collider Accelerator Department, Brookhaven National Laboratory

jmaldonad@bnl.gov



@BrookhavenLab

Outline

- Overview
- Data Preprocessing
- Elog Entry Similarity
- Classification
- Workflow, Summary, & Future Work

Overview

What is the elog system?

- The electronic logbook (elog) system is used to record information ranging from meeting notes, to do lists, and critical operations

Elog | Elog List | Jenn | Options | Add Entry | Reload | Last refresh: 09/29 15:26:25

OC: OC
Default: RHIC

08/26/2022

ae Aug 26 08:41 cp Jenn
Training + Testing Results
1. Added layers & 19 tags & 1000 epochs

- training accuracy ~79%
- Testing accuracy ~71%

Aug 26 08:50 Jenn
confusion matrix

Accuracy Score is 71.8814444878375 %
Recall Score is 39.2599768953494 %
Precision Score is 47.1887996985397 %
FScore is 42.8244686231896 %

Seaborn Confusion Matrix with labels

Elog | Elog List | RHIC | Options | Add Entry | Reload | Last refresh: 09/29 15:40:27

OC: OC
Default: RHIC

03/07/2022

ae Mar 07 00:07 cp kad [1 edit]

ae Mar 07 00:13 cp mcr (RhcInjection) Blue Ring Filled
Pattern: 111x111_P5

ae Mar 07 00:14 cp opsver (Polarization Measurement)
Polarization For Blue 1 H Target1: 54.88 ± 2.23
Injection Energy (23.81), Blue Beam Intensity: 212.77×10^{11}

ae Mar 07 00:15 cp opsver (Polarization Measurement)
Polarization For Blue 2 V Target1: 55.63 ± 1.91
Injection Energy (23.81), Blue Beam Intensity: 212.44×10^{11}

ae Mar 07 00:25 cp mcr (RhcInjection) Yellow Ring Filled
Pattern: 111x111_P5

ae Mar 07 00:25 cp mcr (RhcInjection) Both Rings Filled
Pattern: 111x111_P5

ae Mar 07 00:26 cp opsver (Polarization Measurement)
Polarization For Yellow 1 V Target2: 51.5 ± 1.84
Injection Energy (23.81), Yellow Beam Intensity: 228.43×10^{11}

ae Mar 07 00:28 cp mcr Ramp 33164

Mar 07 00:28 mcr (tape)
Acceleration Ramp Started:

Fill 33164	Ramp	TuneMeter	IPM	OrbFB	TuneFB	ChromFB
Blue	pp22-255GeV-e1_1646630887	BBQ	On	On	On	Off
Yellow	-	BBQ	On	On	On	Off

Mar 07 00:33 opsver (scripTrigger) Injected Beam Stats

pp22-255GeV-e1 Injected Beam Statistics for Fill number 33164

Started filling RHIC: Mon Mar 7 00:00:36 2022, Fill complete: Mon Mar 7 00:25:35 2022, Minutes to fill: 24
Newfill time: Sun Mar 6 23:23:59 2022, Minutes from newfill to accramp: 64

Motivation

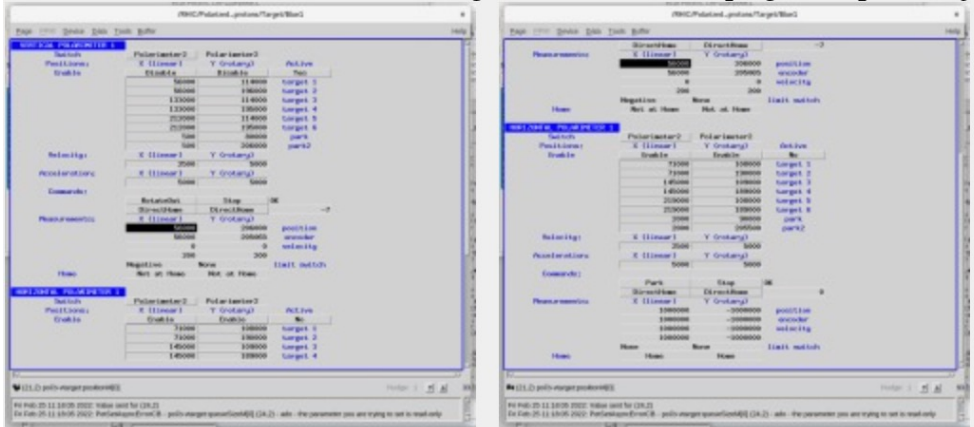
- The search feature only provides exactly what a user enters, what if there are other entries that do not include those exact words BUT also are related to these things
- Can we determine what the user is interested in viewing?
- Eventually provide custom sets of entries based on users' interactions with the system

Motivation

- ML techniques help customize logbook settings and views, including specification of favorite logbooks
- Possibly eliminate the need for manual searching

Feb 25 11:26 cp ph

I'm not convinced that this is the wrong behavior but I dumping some pictures just for documentation purposes.



Feb 25 11:26 ph

Steve called worried that some of the polarimeter planes rdbks are railed at 100000. In the these pictures horiz is updating as expected, vertical is not. This is true for all 4 polarimeter targets (if horiz is updating, vert is not and vice versa).

The HW limits the operations to only a single plane, ie. horiz would need to be at home before vert could be moved. I'm going back through archives and the rdbks are consistent.

Steve is going to do a little more investigating on the HW side but I think things look right on the SW side.

Feb 25 16:17 ph

After making a tunnel access to verify that things looked OK Steve enabled the motion and that triggered the updates.

There can be many kinds of elogs in the database. Ex:
RHIC, Machine Learning, personal elogs
This assumes we have ALL entries from ALL elogs

Removes any punctuation, stop words, and
empty entries. Tokenize each entry & save this
information.

Gensim Doc2Vec model trains 100 epochs.
Returns the top few similar elog entries.

SKLearn MultinomialNB classifier predicts any entry's tag.

By classifying the similar entries output by the D2V model
we can see if any entries are similar by tag. Otherwise,
just use the similar entries in order by most similar.

Elog
Database

NLP
Preprocessing

Doc2Vec
Model

Classification

Similar
Entries

Data Preprocessing

Data

ID	ID	Content	Timestamp	Author	ElogID	Tag	Flag
0	1	<p>bta-th158-ps and bta-qd5-ps both have a sta...	2013-11-18 20:25:48	pdyer	1	bta	0
2	3	NRO wants the same 114 MeV (160 in Booster) se...	2013-11-18 20:06:38	NAK	1		0
3	4	New 114 MeV Au_Ebis file created.	2013-11-18 20:00:04	tape	1		0
4	5	It starts out fine then fades away	2013-11-18 18:00:25	keith	1		0
5	6	Entry deleted	2013-11-18 17:56:49	anonymous	1		0

- The database includes whether entries are a comment, what book they are in, and time entered
- All elog data is stored in a MySQL database

Data Preprocessing

- Preprocessing data improves model performance by focusing on important aspects of our data
- Remove links, numbers, tokenize, lemmatize, lowercase, remove punctuation, also remove any entries with no content in them

Service, building, and equipment tour complete.



[service, building, equipment, tour, complete]

Data Preprocessing

Tokenize

Split each elog entry into individual words

This is an elog entry.



[this, is, an, elog, entry]

Lemmatize

Group different forms of one word into the same form

log logs logging logged



log

Data Preprocessing

Remove Stop Words

Remove commonly used words in the English language

This is an elog entry.



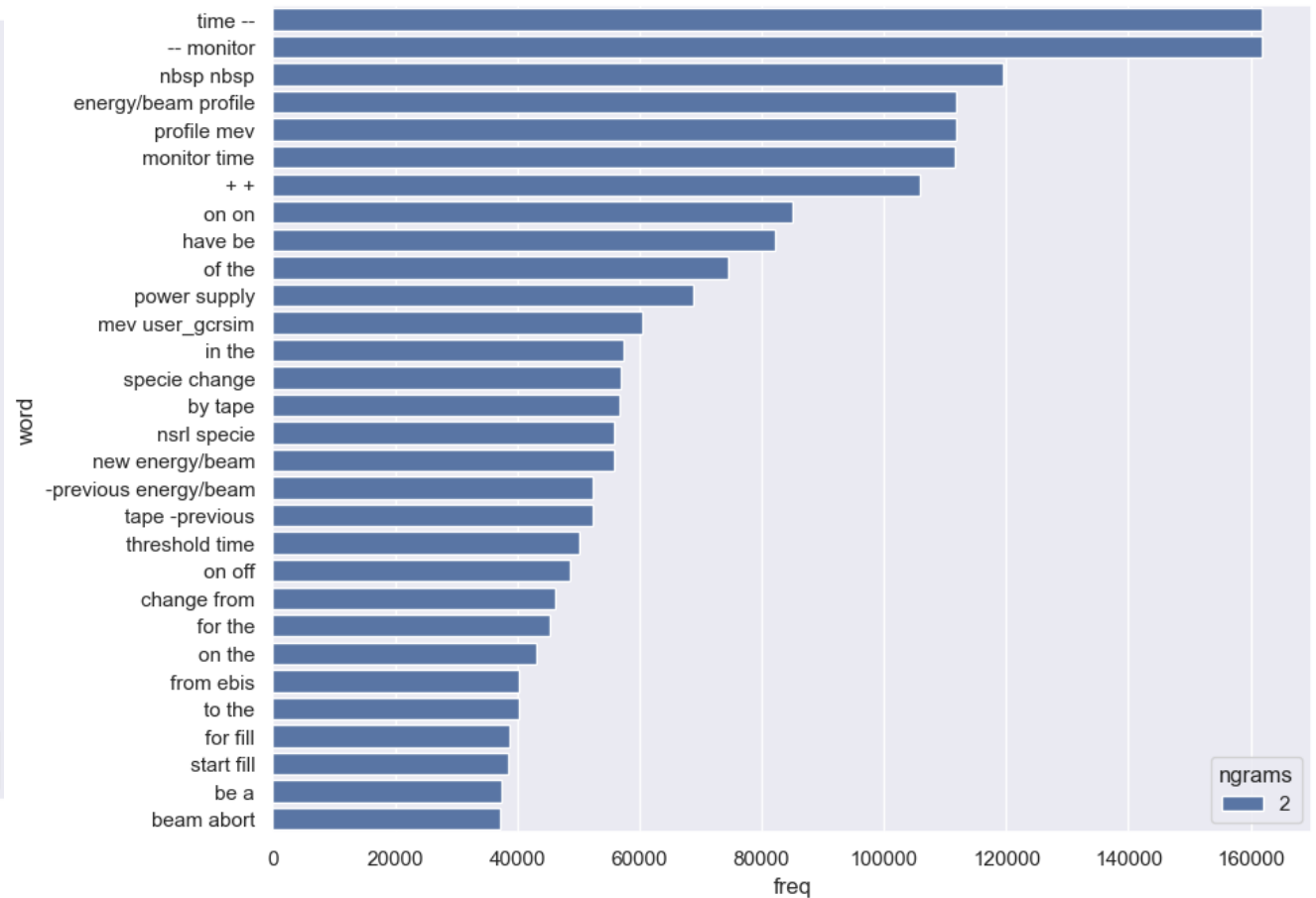
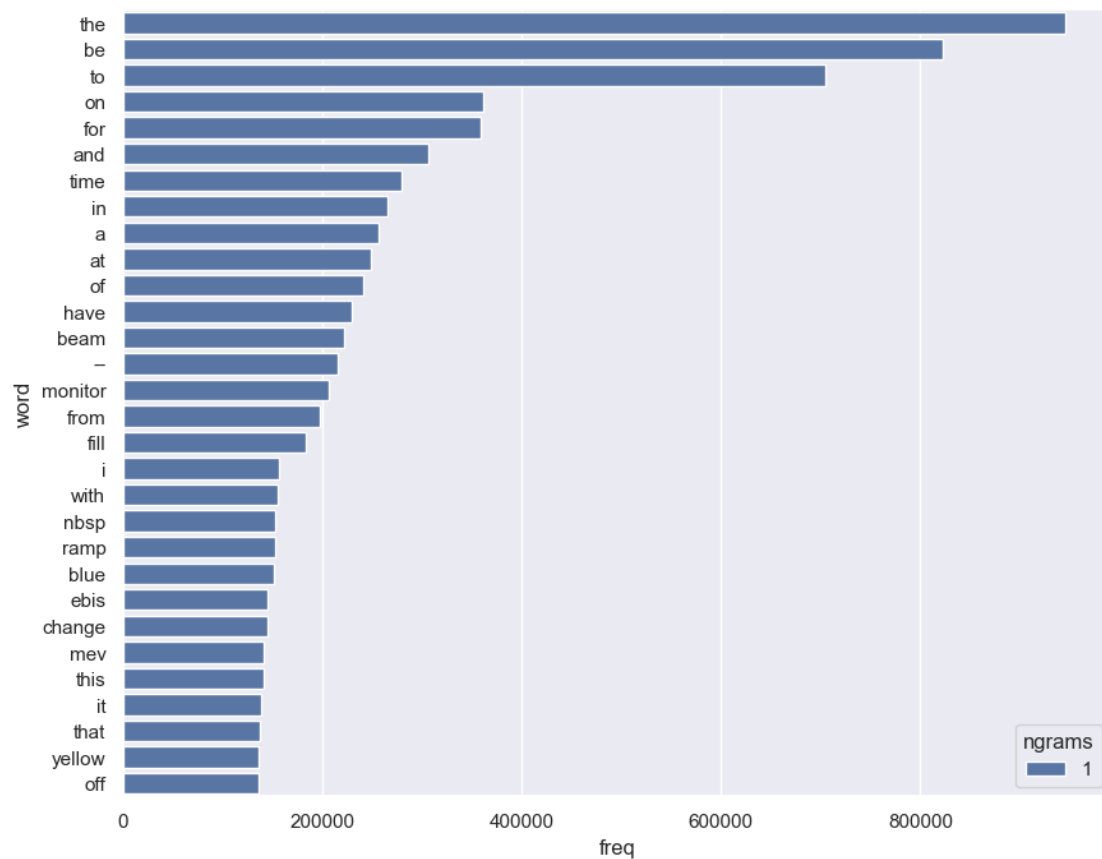
[elog, entry]

Weight

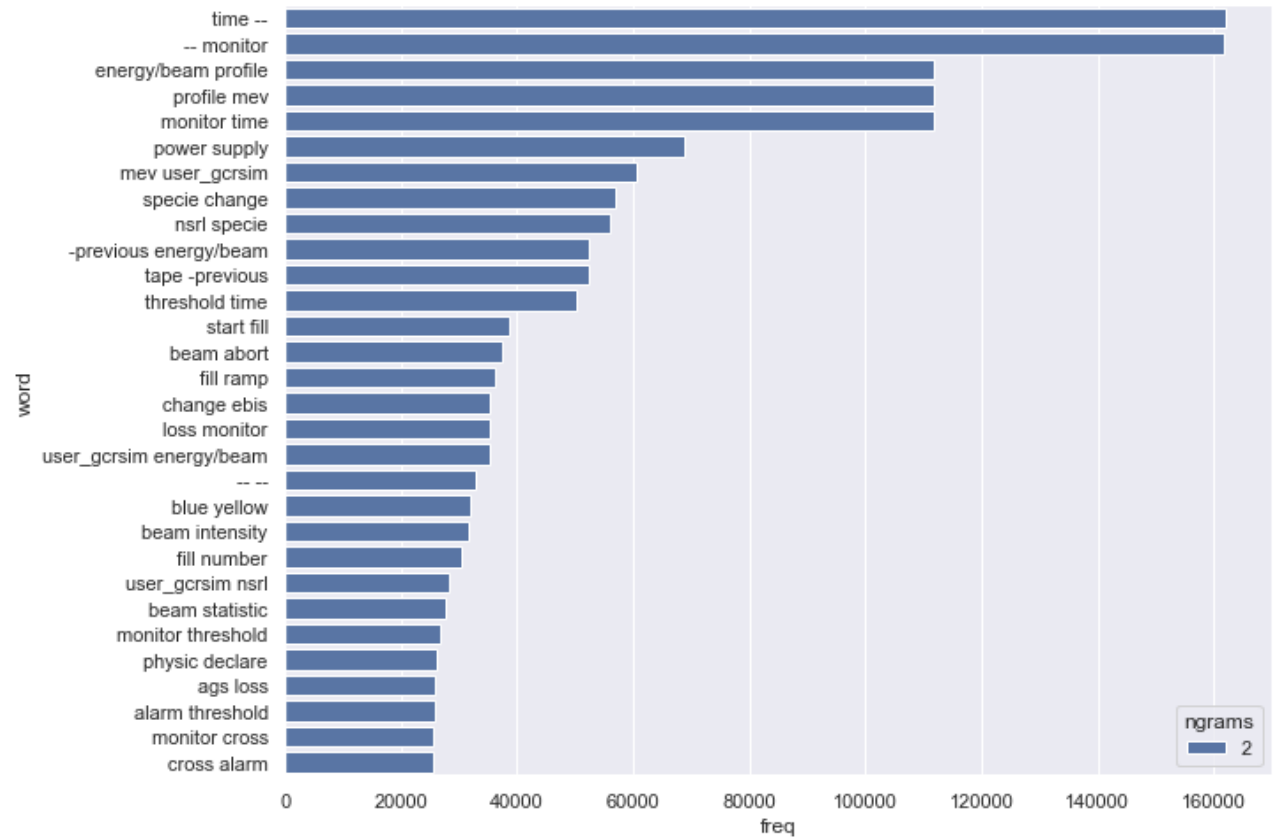
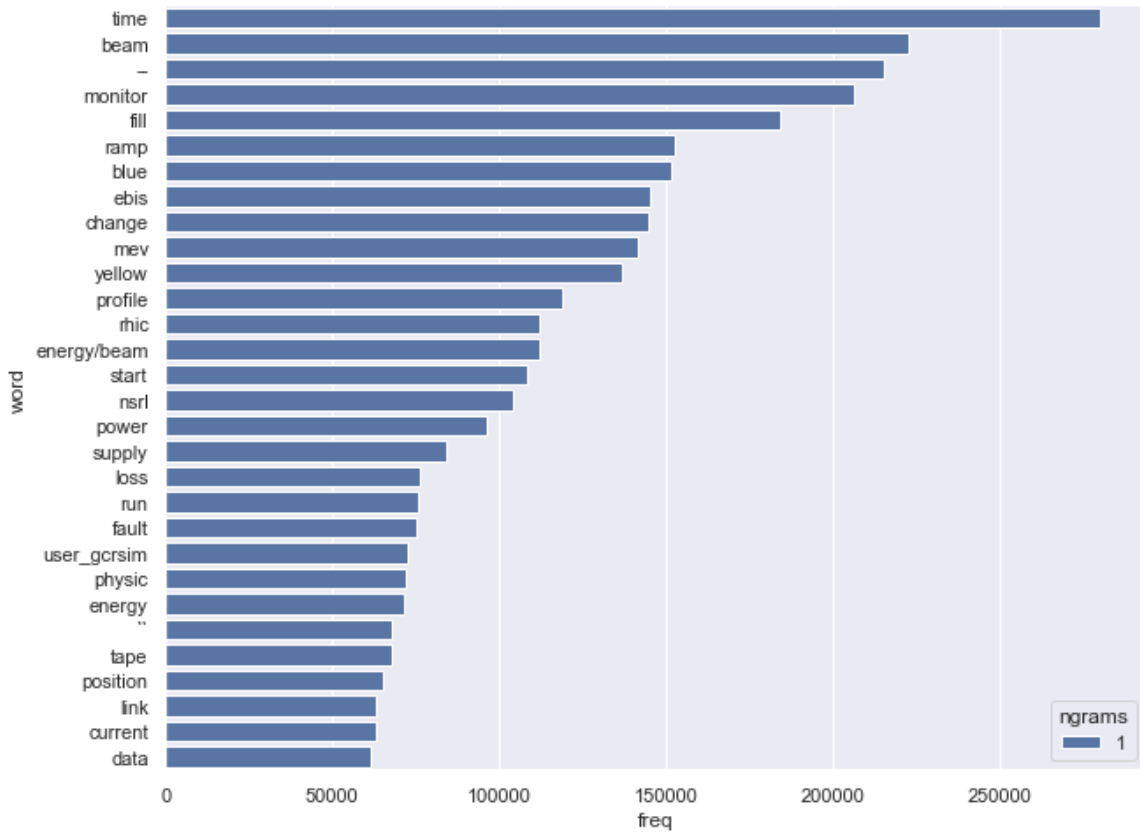
Compute the weight of an entry

- ✓ if an entry is a comment
- ✓ if an entry is flagged
- ✓ if an entry is tagged

Common Elog Words (before preprocessing)



Common Elog Words (after preprocessing)



Elog Entry Similarity

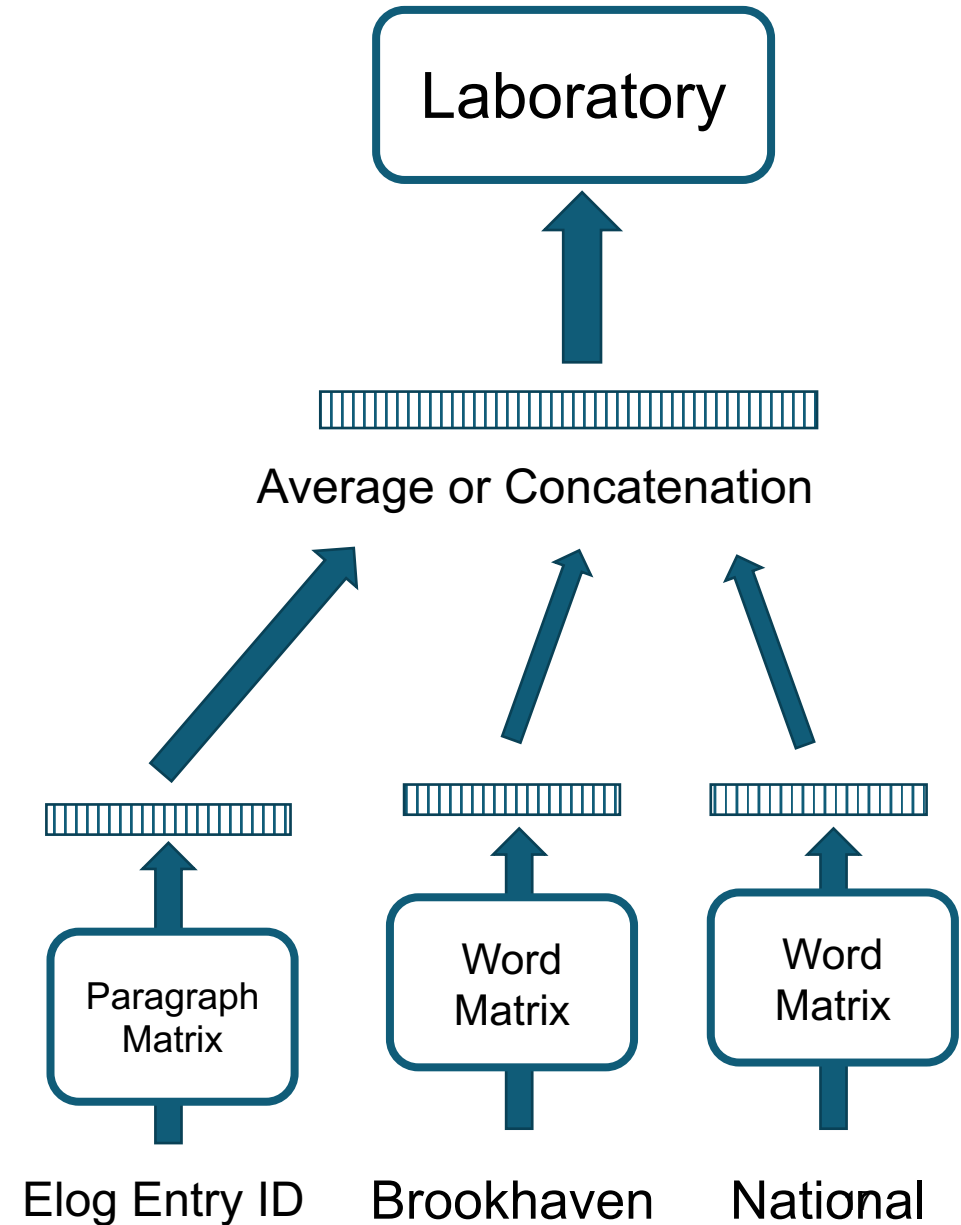
Word2Vec & Doc2Vec

Why Word2Vec?

- One Hot Encoding assigns value to word as they appear in text
- Word2vec creates a representation based on a word's relations with others
- Word2vec combines two models to represent contexts
 - Continuous Bag of Words (CBOW) – predicts context based on surrounding words. The vectors produced represent similar words by different distance metrics.
 - SkipGram – uses one word to predict the context of all surrounding words

Doc2Vec

- Gensim D2V model based on *Distributed Representations of Sentences and Documents* by Le and Mikolov
- Paragraph vectors predict the next word given a sample of words from the paragraph
- Every paragraph is mapped to unique vector which is a column in the paragraph matrix
- Every word in each unique vector also gets mapped to a unique column vector in the word matrix



Doc2Vec

- Take processed entries and create a list of d2v tagged documents
- The model will build a vocabulary from this list
- Trains the model on the list of tagged documents
- D2V most_similar computes the cosine similarity between entries

Doc2Vec

- Used 100 epochs for the model to learn
 - ~1.5 hours on GPU for 1.5+ million entries
- Used 100 epochs for each prediction
- For finding similar documents, 0.3s-1s on GPU
- Daily Training: 100 epochs

Number of Entries per Day	Time Taken to Train (seconds)
50	0.25
100	0.39
700	2.91
1300	4.71

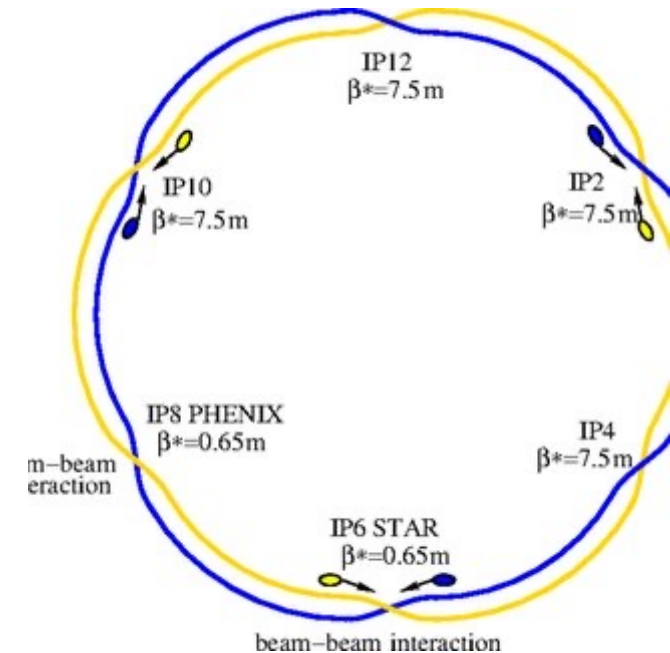
Similar by Words and Most Similar

Polarization

- Background ~ 70%
- Beta ~ 69%
- Polarimeter ~ 64%
- Bunch ~ 63%
- Excitation ~ 63%
- Coherence ~ 62%
- Emittances ~ 60%

Blue

- Yellow ~ 94%
- RHIC ~ 87%
- Ramp ~ 84%
- Power ~ 84%
- Run ~ 82%
- Fill ~ 80%
- Delay ~ 79%



Similar Documents

polarization for yellow 2h target1 store energy
before physics declared yellow beam intensity

1. Yellow 1 V6 Polarization: -51.53 6.05% **78%**
Yellow 2 H6 Polarization: -51.28 10.83%
2. Polarization For Yellow 1 V Target2: 51.44 ± 1.94 Store
Energy (254.21) Before Physics Declared, Yellow Beam **77%**
Intensity: 208.3×10^{11}
3. Yellow 1 V5 Polarization: -56.67 4.87% Yellow 2 H5
Polarization: -59.46 6.09% **76%**

Classification

Metrics

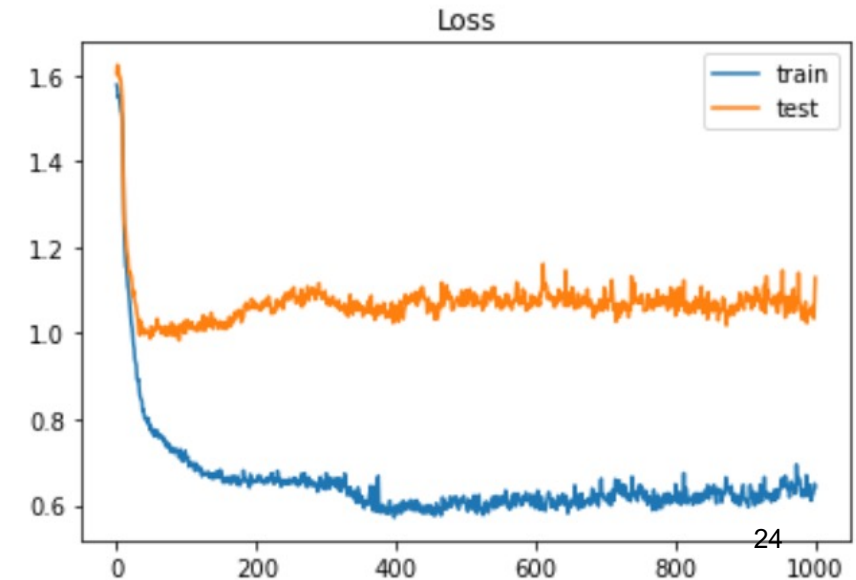
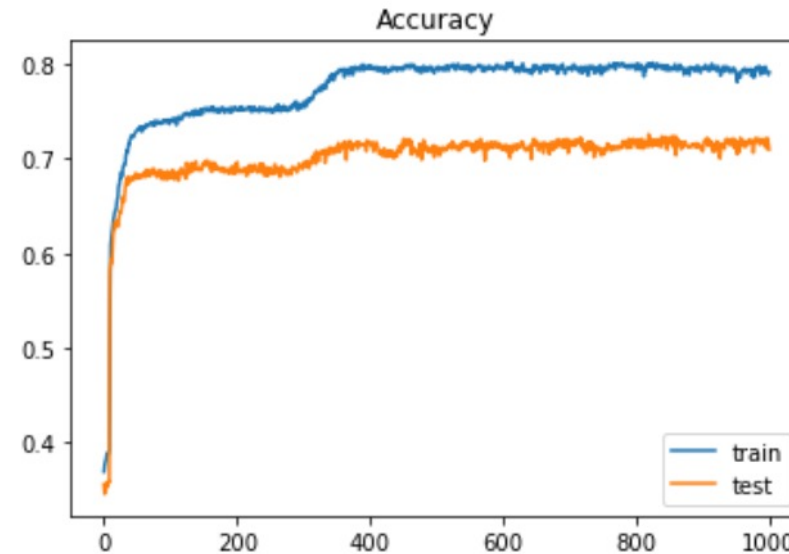
- **Accuracy:** correctly predicted tags / all possible entries to be tagged ($TP+TN / TP+FP+FN+TN$)
- **Recall:** correctly predicted tags / all entries that have that tag ($TP / TP+FN$)
- **Precision:** correctly predicted tag of a specific tag / all entries that were labeled with that tag ($TP / TP+FP$)
- **F-Score:** weighted average of precision & recall ($2 * (\text{recall} * \text{precision}) / (\text{recall} + \text{precision})$)

<https://blog.exsilio.com/all/accuracy-precision-recall-f1-score-interpretation-of-performance-measures/>

Multiclass Classifier

- 19 Tags, 1000 epochs
- Training Accuracy: ~79%
- Testing Accuracy: ~72%
- Recall Score: ~39%
- Precision Score: ~47%
- F Score: ~0.42

<https://www.jeansnyman.com/posts/multi-class-text-classification-with-tensorflow/>



Multinomial Naïve Bayes

- Naïve Bayes for multinomially distributed data
- Naïve Bayes variant for text classification
- Data is word vector counts using SKLearn CountVectorizer
- Accuracy Score: ~78%
- Recall Score: ~74%
- Precision Score: ~66%
- F-Score: ~0.70

Classification Example

R-Failure

Alarm cleared by access control personnel. Might have been related to power lost or the fire alarm testing.

F-MachineSetup

Received call to inform us that the work at the Booster argon station #1 is completed. It is now back to normal operations.

Workflow, Summary & Future Work

Workflow

Once

- Collect the data from the database
- Preprocess and save the data
- Train D2V and multinomial classifier on all data
- Save trained models

Daily

- Collect days entries, varies from 50-1,300
- Preprocess day's data
- Load and train D2V model
- Apply count vectorization
- Load, fit, and predict multinomial classifier
- Sort entries by most similar, considering tags

Future work

- System in place to determine what entries each user interacts with
- During run time, elog entries are often automated. Should this impact the suggested entries?
- Custom model for each user with input data of their written and commented on entries?
- Should typos be dealt with?
- Web interface implemented into the elog system

Thank you!

Special thanks to Dale Yu & Prerana Kankiya for their contributions over the summer.

Thanks to Kevin Brown, Sam Clark, Wenge Fu, and Seth Nemesure for their knowledge and support.