Features extraction from text data

The problem

How do we transform text data into numeric data to feed Machine Learning algorithm?

Bag of word approach:

- Tokenizing
- Counting
- Normalizing

Preprocessing

We do need to preprocess the data in order to:

- Clean the data
- Reduce the number of words/features
- Have more relevant features

Preprocessing

- To lower case
- Strip accent, repeated characters, url, punctuation
- Create special token for numeric, proper noun, hashtags
- Remove stopwords and rare words
- Keep only the stem or the lemme
- Build bigram/trigram
- Add POS information
- Tokenization

Preprocessing

- **Original**: *Jérôme is doing a data sciences presentation*
- **Stemming**: *jérôme is do a data scienc present*
- **Lemmatization**: *jérôme be do a data science presentation*
- **Bigram**: *jérôme is doing a data sciences presentation jérôme_is is_doing doing_a a_data data_sciences sciences_presentation*

Data Transformation

“This is the first document.”
“This is the second second document.”
“And the third one.”

and	document	first	is	one	second	the	this
0	1	1	1	0	0	1	1
0	1	0	1	0	2	1	1
1	0	0	1	1	0	1	0

Document term matrix

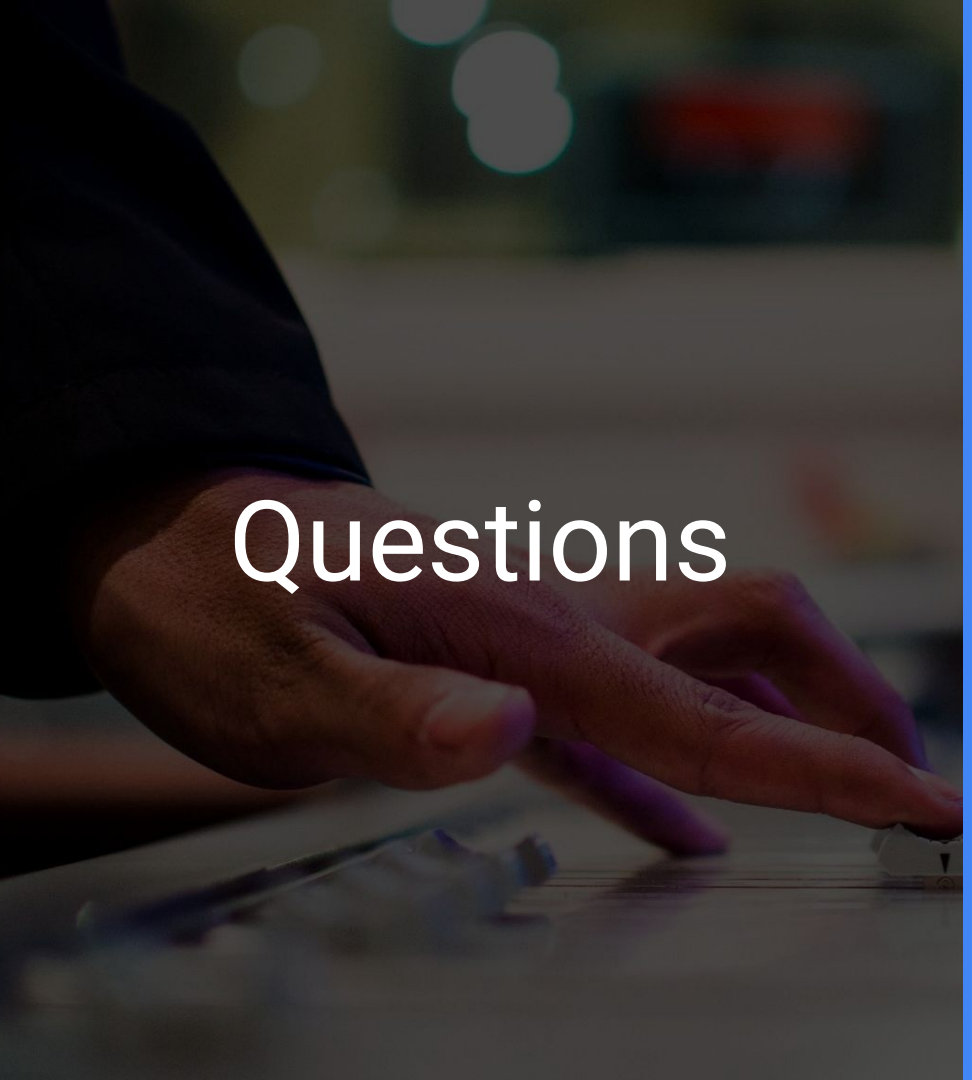
Data Transformation

How to weight the data: term frequency times inverse document frequency:
tf-idf(t, d) = tf(t,d) . idf(t)

Where idf(t)=log(N / df(t,d)) + 1

With :

- N: Number of document
- df(t, d): Number of document d with the term t



Questions