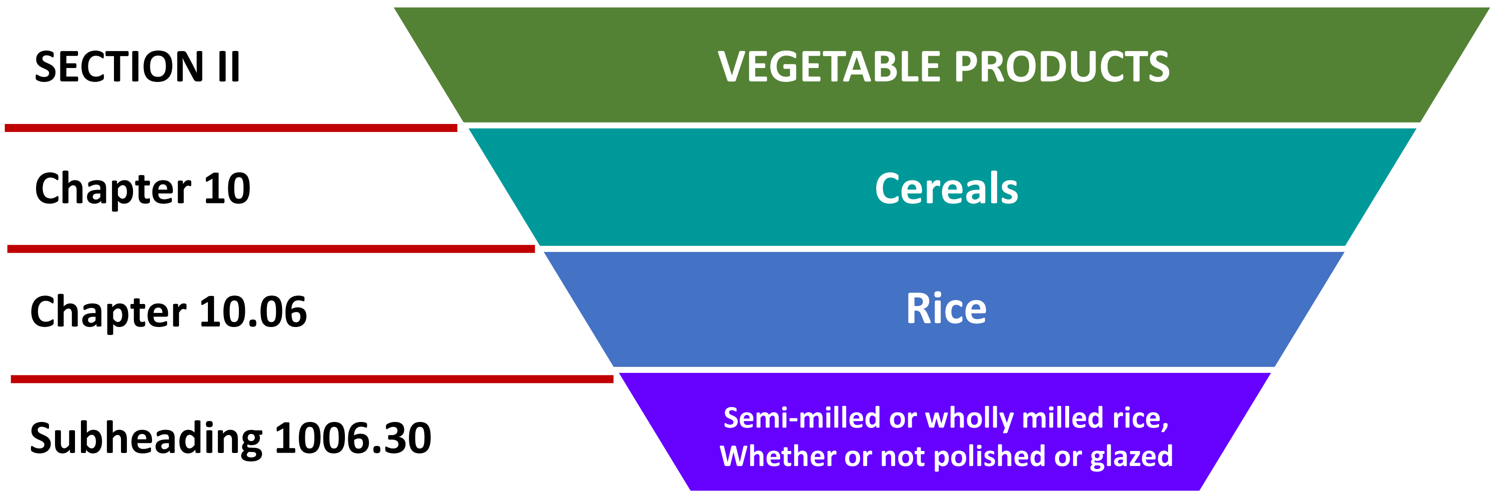
AI HS Code Recommendation Modeling

Dr. Seon Yeong Han

Ⅰ. Overview

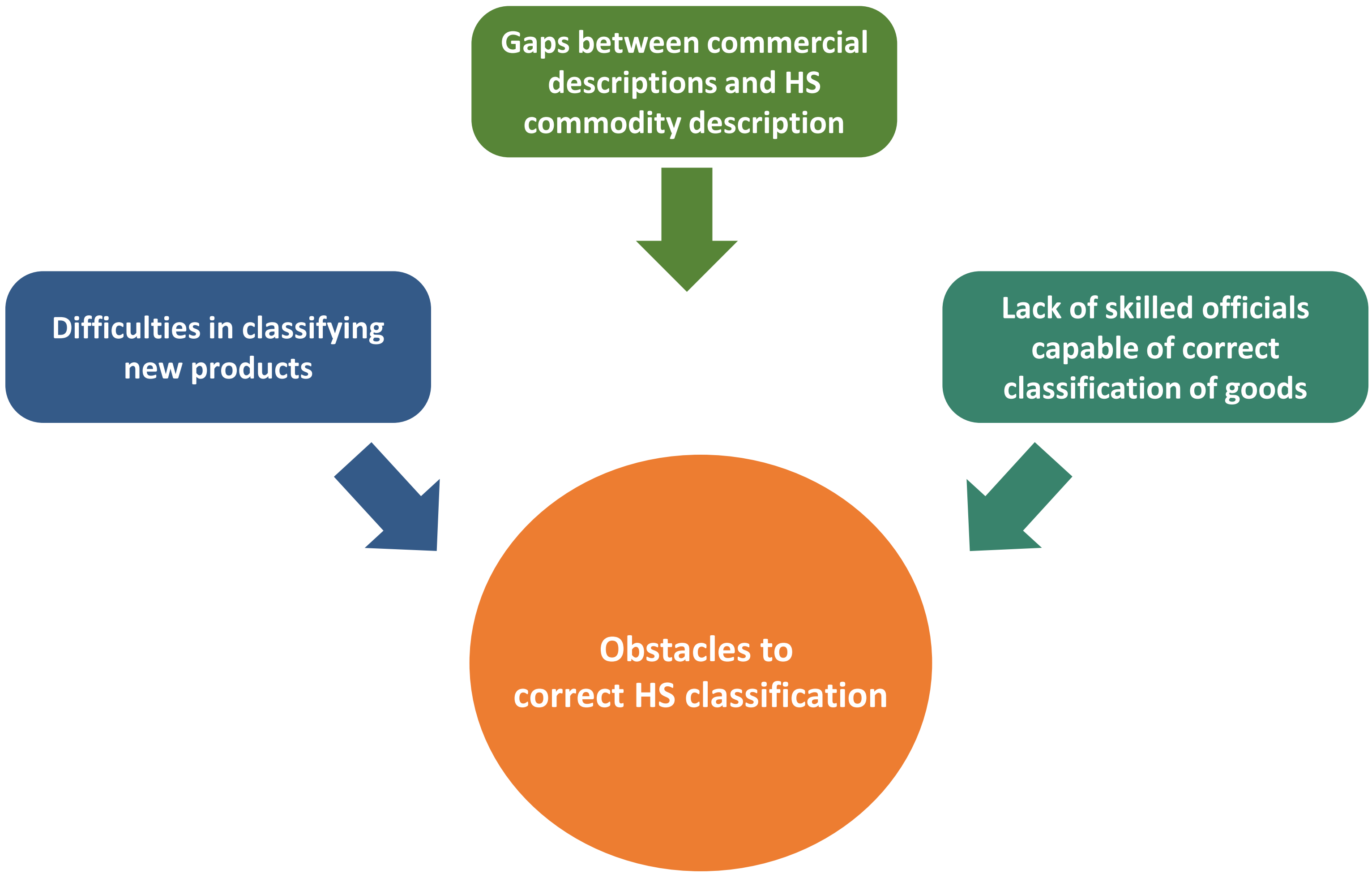
❍ HS Code (Harmonized System code; Harmonized Commodity Description and Coding System)

- Numeric code representing each category of commodities subject to import and export
- HS code basically consists of 6 digits, but most countries use 10~12-digit code system for more detailed classification



Ⅰ. Overview

❍ Issues related to HS classification



Ⅰ. Overview

❍ Researches on HS classification using AI

✓ **Altaheri & Shaalan**

- **Proposed an HS code classifier using the data provided by Dubai Customs for the Artificial Intelligence (AI) hackathon competition**
- **Used TF-IDF, Bag-of-Word techniques**

✓ **Spichakova & Haav**

- **Proposed HS code prediction method using Doc2Vec**
- **Applied a similarity metric that combines text similarity and HS code taxonomy-based semantic similarity.**

Ⅱ. AI model

❍ Issues regarding data to be used for AI modeling

(1) Mislabeled data

- Erroneously declared commodity descriptions

(2) Out of vocabulary

- Newly invented or newly traded commodities

(3) Imbalanced data

- Imbalanced import data between classes (HS codes)

(4) Text dense embedding

- Selection of appropriate training method considering learning speed

Ⅱ. AI model

❍ AI training model

✓ Source data:

US Imports CBP Automated Manifest System(AMS) Shipments 2020

– cargo descriptions collected from January 2002 to September 2020

✓ Only first six digits of HS codes were used

✓ Preprocessing

- Removal of space
- Removal of data solely consisted of numbers
- Removal of descriptions consisted of two or less characters
- Removal of duplicate data

Ⅱ. AI model

❍ AI training model

✓ **Sub-word**

- **Out-Of-Vocaburary problem (about 8% @90% train data)**
- **character n-gram**
- **ex.** cellular smartphone → {ce el ll lu ul la ar sm ma ar rt tp ph ho on ne}

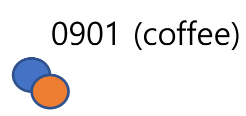
  celular smartphones → {ce el lu ul la ar sm ma ar rt tp ph ho on ne es}

  → 15/16 match → similar embedding

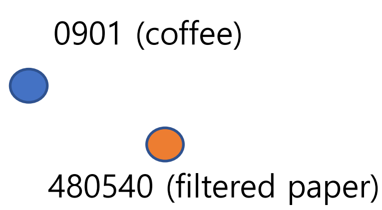
✓ **Word n- gram**

- **word order information**
- **ex.** Coffee filter

  filtered coffee



Vs.

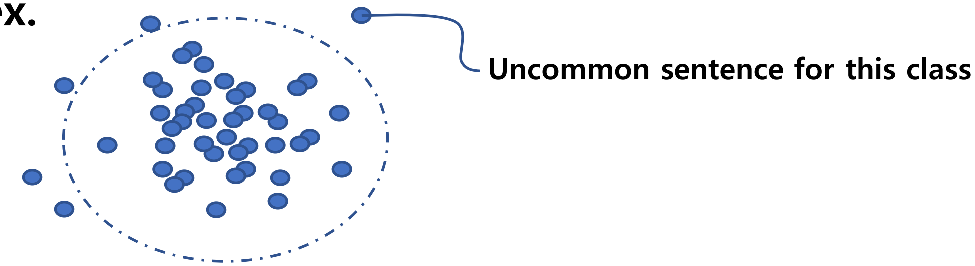


Ⅱ. AI model

❍ AI training model

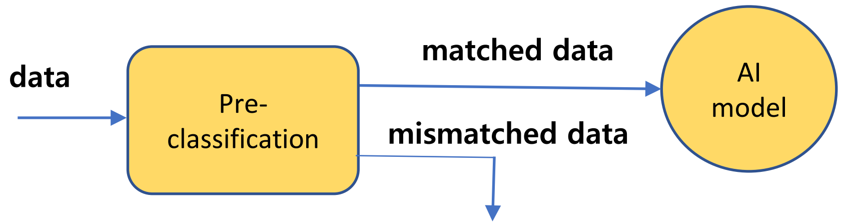
✓ **Distance based outlier detection**

- mislabeled data problem
- same class data embedding → outlier detection
- ex.



✓ **Classification based outlier detection**

- mislabeled data problem
- pre- classification for outlier remove

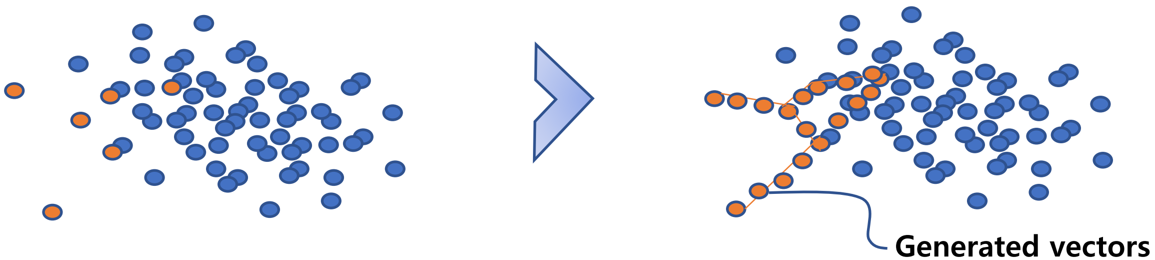


Ⅱ. AI model

❍ AI training model

✓ **Balanced sampling**

- imbalanced data set problem
- how to plausible data generation?
- Synthetic Minority Oversampling Technique (SMOTE)
- ex.



Ⅱ. AI model

❍ AI training model

✓ Word2vec

- convert word into vector maintaining semantic similarity
- Continuous Bag Of Words(CBOW) : neighbor words are trained together
- Skipgram : a word is trained towards its neighbor words

✓ Doc2Vec

- word2vec with document ID
- Paragraph Vector Distributed Memory (PV-DM) : CBOW with document ID
- Paragraph Vector Distributed Bag Of Word (PV-DBOW) : a document ID is trained towards its words

✓ FastText : Word2Vec + subword

✓ FastText Classification : words and sub-words are trained towards class

Ⅱ. AI model

❍ Implementation of AI training model

Scenario	Description
CBOW+SVM	Classify using SVM the embeddings created by Word2Vec-CBOW
Skipgram+SVM	Classify using SVM the embeddings created by Word2Vec-skipgram
PV-DBOW+ms	Predict HS code using cosine similarity after embedding text using Doc2Vec PV-DBOW
PV-DM+ms	Predict HS code using cosine similarity after embedding text using Doc2Vec PV-DM
Doc2Vec+SVM	Classify using SVM the results provided by combined model of PV-DM and PV-DBOW
FastText+SVM	Classify using SVM the embeddings created by FastText model
FastText-cl	Recommend HS code using FastText-classification model
FastText-cl+d-outlier	Apply FastText-classification model after removing distance-based outlier from training data set
FastText-cl+cl-outlier	Apply FastText-classification model after removing classification-based outlier from training data set
FastText-cl+bigram	Consider the order of words by selecting bigram option of FastText-classification
FastText+SVM+b-sampling	Apply SMOTE balance-sampling when creating training data set

Ⅱ. AI model

❍ Training results

- ✓ Embedding and classification models were created for each HS chapter
  Only classes with over 200 declarations were analyzed
- ✓ Performances were evaluated for five randomly selected chapters

❍ Accuracy : high

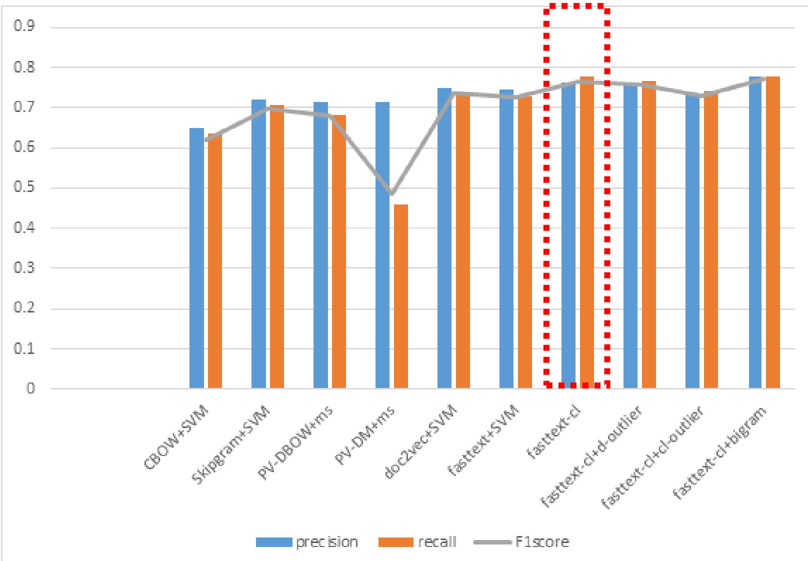
HS	Scenario	precision	recall	F1-score	acc@3top	acc@5top
Chapter 40	**FastText-cl**	0.89	0.89	0.88	0.96	0.97
Chapter 62	**FastText+SVM**	0.71	0.7	0.69	0.91	0.96
Chapter 73	**FastText-cl**	0.75	0.74	0.74	0.87	0.91
Chapter 61	**FastText-cl+bigram**	0.76	0.76	0.76	0.89	0.93
Chapter 33	**FastText-cl+bigram**	0.8	0.8	0.79	0.93	0.96

❍ Accuracy : low

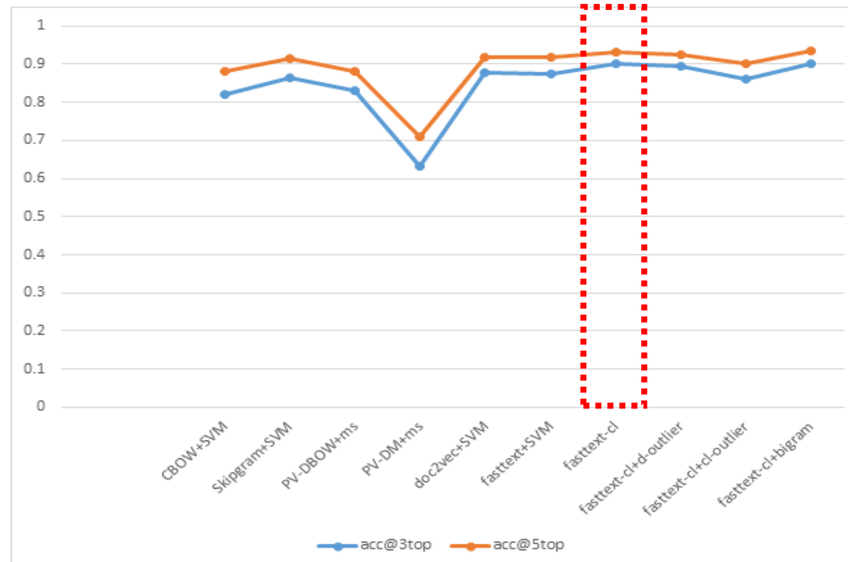
HS	Scenario	precision	recall	F1-score	acc@3top	acc@5top
Chapter 40	**PV-DM+ms**	0.79	0.49	0.52	0.68	0.76
Chapter 62	**CBOW+SVM**	0.56	0.57	0.54	0.76	0.83
Chapter 73	**CBOW+SVM**	0.66	0.6	0.59	0.8	0.86
Chapter 61	**CBOW+SVM**	0.63	0.61	0.6	0.81	0.87
Chapter 33	**CBOW+SVM**	0.58	0.58	0.55	0.8	0.88

❍ Training results

✓ Avrage Precision, Recall, F1-score for 5 Chapters



✓ Average acc@3top, acc@5top Performance for 5 Chapters

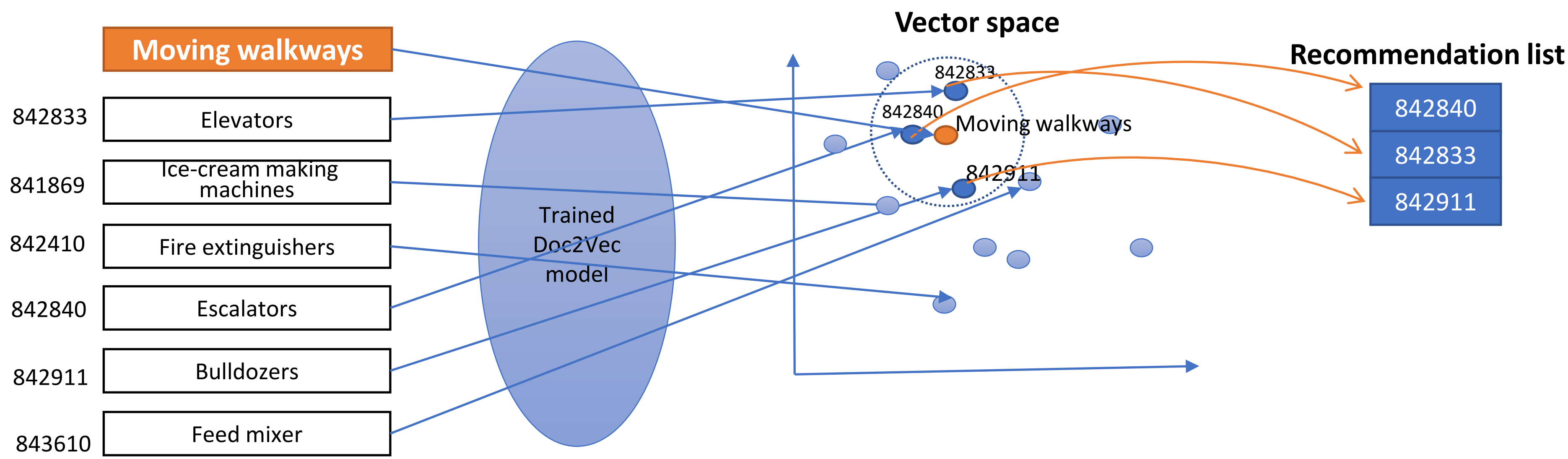


The results of the application of 10 models to five randomly selected chapters show that

FastText-cl+bigram presents great performances in all of the five chapters.

Ⅲ. Conclusion

❍ AI model concept diagram



❍ Expected effects

- ✓ **Decrease of erroneously declared HS codes**
- ✓ **Post-audit for detection of illegal declaration submitted after release of goods**

➔ Improved revenue collection by Customs Administration thanks to the accurate declaration of HS code