## CLAIMS

1. A system comprising a computing device, wherein the computing device comprises a processor and a storage device storing computer executable code, and the computer executable code, when executed at the processor, is configured to:

retrieve product descriptions of products and provide a plurality of seed tags, wherein the product descriptions comprise text identical to the seed tags and the seed tags characterize at least one of the products;

train a named-entity recognition (NER) model using the product descriptions and the seed tags;

predict pseudo tags from the product descriptions using the NER model;

calculate confidence scores of the pseudo tags;

compare each of the confidence scores with a threshold, and define one of the pseudo tags as a true tag when the confidence score of the pseudo tag is greater than the threshold;

add the true tags to the seed tags to obtain updated tags; and

repeat the steps of training, predicting, calculating, comparing and adding using the product descriptions and the updated tags, so as to keep updating the updated tags,

wherein the NER model is configured to provide a label for each character of the product descriptions, the labels comprises beginning of a brand ($B_{bra}$), middle of a brand ($M_{bra}$), end of a brand ($E_{bra}$), beginning of a product industrial attributions ($B_{att}$), middle of a product industrial attribution ($M_{att}$), end of a product industrial attribution ($E_{att}$), beginning of a function ($B_{fun}$), middle of a function ($M_{fun}$), end of a function ($E_{fun}$), beginning of a style ($B_{sty}$, middle of a style ($M_{sty}$), end of a style ($E_{sty}$), beginning of a feeling ($B_{fel}$), middle of a feeling ($M_{fel}$), end of a feeling ($E_{fel}$), beginning of a scenario ($B_{sce}$), middle of a scenario ($M_{sce}$), end of a scenario ($E_{sce}$), beginning of a human audience ($B_{hum}$), middle of a human audience ($M_{hum}$), end of a human audience ($E_{hum}$), and no label (O), and the labels are used to generate the tags.

2. The system of claim 1, wherein the NER model comprises a bidirectional encoder representations from transformers (BERT) layer, a bidirectional long short-term memory (BiLSTM) layer, and a conditional random field (CRF) layer.

3. The system of claim 1, wherein the confidence score of each of the pseudo tags is a number of detections of the pseudo tags from the product descriptions.

4. The system of claim 1, wherein the tags have different types, and the types of the tags include brand tags indicating brands of the products, industrial attribution tags indicating important physical features of the products, function tags indicating functions of the products, style tags indicating styles of the products, feeling tags indicating impression of customers to the products, scenario tags indicating suitable scenarios for using the products, and audience tags indicating targeted customers of the products, the threshold is 2-3 for the brand tags, 5-20 for the industrial attribution tags, the style tags, the audience tags, and the scenario tags, and 10-50 for the feeling tags.

5. The system of claim 4, wherein values of the thresholds of the tags increase during different rounds of repeating the steps of training, predicting, calculating, comparing and adding.

6. The system of claim 1, wherein the CRF layer is configured to learn constraint between labels by maximizing a sentence labeling score, and the sentence labeling score for each of the plurality of sentence predictions is calculated by:

$$s(X, y) = \sum_{i=0}^{n} A_{y_i, y_{i+1}} + \sum_{i=1}^{n} P_{i, y_i} ,$$

wherein $X$ is one of the documents and comprises $n$ number of words $x_1, x_2, \ldots, x_n$, $Y$ is predicted label sequence of the one of the documents and comprises $y_1, y_2, \ldots, y_n$, $A_{y_i, y_{i+1}}$ is a matrix of transition score representing transition from the label $y_i$ to the label $y_{i+1}$ and corresponds to the constraint between neighboring labels, and $P_{i, y_i}$ is a possibility of the $i$-th word in the

documents having the label $y_i$.

7. The system of claim 6, wherein the sentence labeling score for each of the plurality of sentence predictions is normalized by:

$$p(y|X) = \frac{e^{s(X,y)}}{\sum_{\tilde{y}\in Y_X} e^{(X,\tilde{y})}},$$

wherein $p(y|X)$ is a normalized possibility for the document $X$ relative to the label $y$, $\tilde{y}$ is any one of the predicted labels of the documents.

8. The system of claim 1, wherein the computer executable code is further configured to: provide the updated tags as keywords of the corresponding products, such that a user can use the keywords to find the corresponding products via a search engine.

9. The system of claim 1, wherein the computer executable code is further configured to: provide the updated tags as keywords of the corresponding products, and displaying the keywords on a homepage of the corresponding products as comment tags or title words.

10. A method, comprising:

retrieving, by a computing device, product descriptions of products, and providing, by the computing device, a plurality of seed tags, wherein the product descriptions comprises text identical to the seed tags and the seed tags characterize at least one of the products;

training, by the computing device, a named-entity recognition (NER) model using the product descriptions and the seed tags;

predicting, by the computing device, pseudo tags from the product descriptions using the NER model;

calculating, by the computing device, confidence scores of the pseudo tags;

comparing, by the computing device, each of the confidence scores with a

threshold, and defining one of the pseudo tags as a true tag when the confidence score of the pseudo tag is greater than the threshold;

adding, by the computing device, the true tags to the seed tags to obtain updated tags; and

repeating, by the computing device, the steps of training, predicting, calculating, comparing and adding using the product descriptions and the updated tags, so as to keep updating the updated tags,

wherein the NER model is configured to provide a label for each character of the product descriptions, the labels comprises beginning of a brand ($B_{bra}$), middle of a brand ($M_{bra}$), end of a brand ($E_{bra}$), beginning of a product industrial attributions ($B_{att}$), middle of a product industrial attribution ($M_{att}$), end of a product industrial attribution ($E_{att}$), beginning of a function ($B_{fun}$), middle of a function ($M_{fun}$), end of a function ($E_{fun}$), beginning of a style ($B_{sty}$, middle of a style ($M_{sty}$), end of a style ($E_{sty}$), beginning of a feeling ($B_{fel}$), middle of a feeling ($M_{fel}$), end of a feeling ($E_{fel}$), beginning of a scenario ($B_{sce}$), middle of a scenario ($M_{sce}$), end of a scenario ($E_{sce}$), beginning of a human audience ($B_{hum}$), middle of a human audience ($M_{hum}$), end of a human audience ($E_{hum}$), and no label (O), and the labels are used to generate the tags.

11. The method of claim 10, wherein the NER model comprises a bidirectional encoder representations from transformers (BERT) layer, a bidirectional long short-term memory (BiLSTM) layer, and a conditional random field (CRF) layer.

12. The method of claim 10, wherein the confidence score of each of the pseudo tags is a number of detections of the pseudo tags from the product descriptions.

13. The method of claim 10, wherein the tags have different types, and the types of tags comprises brand tags indicating brands of the products, industrial attribution tags indicating important physical features of the products, function tags indicating functions of the products, style tags indicating styles of the products, feeling tags indicating impression of customers to the products, scenario tags

indicating suitable scenarios for using the products, and audience tags indicating targeted customers of the products, the threshold is 2-3 for the brand tags, 5-20 for the industrial attribution tags, the style tags, the audience tags, and the scenario tags, and 10-50 for the feeling tags.

14. The method of claim 10, wherein the CRF layer is configured to learn constraint between labels by maximizing a sentence labeling score, and the sentence labeling score for each of the plurality of sentence predictions is calculated by:

$$s(X, y) = \sum_{i=0}^{n} A_{y_i, y_{i+1}} + \sum_{i=1}^{n} P_{i, y_i} ,$$

wherein *X* is one of the documents and comprises *n* number of words $x_1, x_2, \ldots, x_n$, *Y* is predicted label sequence of the one of the documents and comprises $y_1, y_2, \ldots, y_n$, $A_{y_i, y_{i+1}}$ is a matrix of transition score representing transition from the label $y_i$ to the label $y_{i+1}$ and corresponds to the constraint between neighboring labels, and $P_{i, y_i}$ is a possibility of the *i*-th word in the documents having the label $y_i$.

15. The system of claim 14, wherein the sentence labeling score for each of the plurality of sentence predictions is normalized by:

$$p(y|X) = \frac{e^{s(X, y)}}{\sum_{\tilde{y} \in Y_x} e^{(X, \tilde{y})}} ,$$

wherein $p(y|X)$ is a normalized possibility for the document *X* relative to the label *y*, $\tilde{y}$ is any one of the predicted labels of the documents.

16. The method of claim 10, further comprising:

providing the updated tags as keywords of the corresponding products, such that a user can use the keywords to find the corresponding products via a search bar; or

providing the updated tags as keywords of the corresponding products, and displaying the keywords on a homepage of the corresponding products as comment tags or title words.

17. A non-transitory computer readable medium storing computer executable code, wherein the computer executable code, when executed at a processor of a computing device, is configured to:

retrieve product descriptions of products and a plurality of seed tags, wherein the product descriptions comprises text identical to the seed tags and the seed tags characterize at least one of the products;

train a named-entity recognition (NER) model using the product descriptions and the seed tags;

predict pseudo tags from the product descriptions using the NER model;

calculate confidence scores of the pseudo tags;

compare each of the confidence scores with a threshold, and define one of the pseudo tags as a true tag when the confidence score of the pseudo tag is greater than the threshold;

add the true tags to the seed tags to obtain updated tags; and

repeat the steps of training, predicting, calculating, comparing and adding using the product descriptions and the updated tags, so as to keep updating the updated tags,

wherein the NER model is configured to provide a label for each character of the product descriptions, the labels comprises beginning of a brand ($B_{bra}$), middle of a brand ($M_{bra}$), end of a brand ($E_{bra}$), beginning of a product industrial attributions ($B_{att}$), middle of a product industrial attribution ($M_{att}$), end of a product industrial attribution ($E_{att}$), beginning of a function ($B_{fun}$), middle of a function ($M_{fun}$), end of a function ($E_{fun}$), beginning of a style ($B_{sty}$, middle of a style ($M_{sty}$), end of a style ($E_{sty}$), beginning of a feeling ($B_{fel}$), middle of a feeling ($M_{fel}$), end of a feeling ($E_{fel}$), beginning of a scenario ($B_{sce}$), middle of a scenario ($M_{sce}$), end of a scenario ($E_{sce}$), beginning of a human audience ($B_{hum}$), middle of a human audience ($M_{hum}$), end of a human audience ($E_{hum}$), and no label (O), and the labels are used to generate the tags.

18. The non-transitory computer readable medium of claim 17, wherein the NER model comprises a bidirectional encoder representations from transformers

(BERT) layer, a bidirectional long short-term memory (BiLSTM) layer, and a conditional random field (CRF) layer.