

AN AUDIO AND SUBTITLE SYNCHRONIZATION METHOD BASED ON AUDIO TRANSLATION AND SIMILARITY CALCULATION

TECHNICAL FIELD

The present invention relates to the technical field of audio processing, in particular, to an audio and subtitle synchronization method based on audio translation and similarity calculation.

BACKGROUND ART

With the continuous development of mobile Internet, the online literature market is gradually popular, and people will read and listen to various types of novels in their spare time. During the process of listening, users may not actually listen for various reasons, but novels continue to play, thus, it is difficult for the users to find last breakpoint positions. In addition, some users like to listen to a novel and read subtitles at the same time, so how to accurately match texts with audio timelines is also a problem to be solved. At present, there are many audio-to-text algorithms, but various reasons such as the environment, dialect, speaking speed and the like of the recorder will cause a model to be unable to accurately translate audios into correct words, including the situations that role names in novels cannot be accurately translated, and a single word is translated into multiple words, and the like.

SUMMARY

In view of the deficiencies of the prior art, the present invention aims to provide an audio and subtitle synchronization method based on audio translation and similarity calculation.

In order to achieve the above purpose, the present invention adopts the following technical solution:

an audio and subtitle synchronization method based on audio translation and similarity calculation is specifically implemented as follows:

S1: acquisition of a source text of a novel and audio files of the novel:

acquiring the source text of the novel through a crawler technology, and storing the whole novel as a unit; and downloading the audio files of the novel locally through a database and storing the audio files by chapters;

S2: data cleaning and conversion:

carrying out data cleaning on the source text of the novel so as to process the source text of the novel into a required text content; and converting the audio files of the novel into wav-format files of 18000Hz;

S3: downloading a trained speech-to-text model and a word correction model; where the speech-to-text model adopts a STT model; and an input of the STT model is the audio file of each chapter of the novel processed in step S2, and outputs thereof are

each character of each chapter obtained by translation and a corresponding timestamp of each character in an audio; and

concatenating output characters into corresponding words, and obtaining a corresponding time in the audio for each word; then, verifying the words concatenated by use of the word correction model to ensure the accuracy of the words; and concatenating the words of each chapter obtained by translation in step S3 so as to obtain a translated text of each chapter;

S4: article dividing:

dividing the source text of the novel processed in step S2 by chapters, so as to obtain a source text of each chapter; and

S5: sentence matching:

further dividing the source text of each chapter of the novel in paragraphs, carrying out area matching and probability calculation on the obtained texts and the translated texts of corresponding chapters to determine a stop position of each paragraph of the source text in the translated text, acquiring a corresponding time of the stop position of each paragraph of the source text in the audio according to a corresponding time of the word at the stop position in the audio, and finally, outputting a result of timeline correspondence between the source text and audio of the novel.

Further, step S5 is specifically implemented as follows:

S5.1: dividing the source text of each chapter of the novel into paragraphs respectively with a full stop; where because the translated text only contains commas and full stops, it is necessary to delete punctuation marks except commas and full stops in the paragraphs obtained by the division of the source text of the novel;

S5.2: initialization:

storing all the paragraphs obtained by the division of the source text of each chapter of the novel in a list L in order, traversing the list L, determining the length of each paragraph in the list L, and preliminarily determining that each paragraph of one chapter is within an interval range of the translated text of a corresponding chapter according to the length of each paragraph;

S5.3: determination of range windows:

determining a start position of a window: the position of a stop word of a previous paragraph + the length of a current paragraph - a window spacing = the start position of the window; where the window spacing is the number of words in the window.

determining a stop position of the window: the position of the stop word in the previous paragraph + the length of the current paragraph + the window spacing = the stop position of the window;

defining a range window by the start position and stop position of the window as A1;

S5.3: post-paragraph positioning:

determining words in the range window A1, then selecting the last three words of the current paragraph to generate a range window A2, and at the same time, generating a sliding window with a spacing of 3 in the range window A1, and sliding the words in the range window A1 to respectively compare with the words in the range window A2; during the process of comparison, initializing three variables i, j and t to respectively record a pointer position of the sliding window of the range window A1, a pointer position of the range window A2 and a counter which are all 0 initially; then, traversing the range window A2 and comparing the word at a position j in the range window A2 with the word at a position i in the sliding window A1, if the two words are the same through comparison, determining that the counter t is equal to t+1, the pointer position i of the sliding window of the range window A1 is equal to i+1 and the pointer position j of the range window A2 is equal to j+1, otherwise, determining that the pointer position j of the range window A2 is equal to j+1 and the rest is unchanged; carrying out comparison repeatedly until the pointer position j is equal to the length of the range window A2, and ending the circulation; and finally, if the probability that the word in the sliding window of the range window A1 is the same as the word in the range window A2 is equal to $\frac{2}{3}$, considering that the stop word of the current paragraph is within the sliding window of the range window A1;

where, the stop word should be determined according to the matching of the words in the sliding window of the range window A1 and the words in the range window A2, if the same two words in the sliding window of the range window A1 and the range window A2 are the last two words in the sliding window of the range window A1 and the first two words in the range window A2, the stop word is the next word of the last word in the sliding window of the range window A1; if the same two words in the sliding window of the range window A1 and the range window A2 are the first two words in the sliding window of the range window A1 and the first two words in the range window A2, alternatively, the last two words in the sliding window of the range window A1 and the last two words in the range window A2, the stop word is the last word in the sliding window of the range window A1; and if the same two words in the sliding window of the range window A1 and the range window A2 are the first two words in the sliding window of the range window A1 and the last two words in the range window A2, the stop word is the second word in the sliding window of the range window A1;

S5.4: pre-paragraph positioning:

if in the post-paragraph positioning in S5.3, the stop word of the current paragraph fails to be found in the range window A1, carrying out pre-paragraph positioning, i.e., selecting the next paragraph of the current paragraph, and taking the first three words of the next paragraph of the current paragraph as a range window A3, where the spacing of the sliding window in the range window A1 is also 3;

initializing three variables i, j and t to respectively record the pointer position of the sliding window of the range window A1, a pointer position of the range window A3 and the counter, which are all 0 initially; then, traversing the range window A3 and

comparing a word at a position j in the range window A3 with the word at the position i in the sliding window A1, if the two words are the same through comparison, determining that the counter t is equal to $t+1$, the pointer position i of the sliding window of the range window A1 is equal to $i+1$ and the pointer position j of the range window A3 is equal to $j+1$, otherwise, determining that the pointer position j of the range window A3 is equal to $j+1$ and the rest is unchanged; carrying out comparison repeatedly until the pointer position j is equal to the length of the range window A3, and ending the circulation; continuously circulating the words of the two windows until the window with a probability equal to $2/3$ is found, then determining the position pointer of the first word in the sliding window of the range window A1, and allowing the obtained value to minus 1 so as to find the stop word of the current paragraph;

step S5.5: paragraph merging:

if in the pre-paragraph positioning in S5.4, the stop word of the current paragraph also fails to be found in the range window A1, meaning that the translated text has a large error for the current paragraph and the stop word cannot be found through probability comparison, in this case, directly merging the two paragraphs, and circulating the post-paragraph positioning and the pre-positioning until a stop word of the paragraph satisfying the conditions is found; and

step S5.6: result outputting:

carrying out result outputting once the stop word of each paragraph is found.

Further, in step S5.1, if a paragraph obtained by dividing contains more than 20 words, the paragraph is divided once again with commas to ensure that the number of words in each paragraph obtained by dividing do not exceed 20.

Further, in step S5.3, the window spacing is chosen to be 8.

Further, in step S5.6, the format of the result outputting as follows:

paragraph number

start time of the paragraph --> stop time of the paragraph

source text of the paragraph

the paragraph number starts at 0 and indicates to which paragraph of the chapter is belonged; the second line is a timeline of the paragraph, the left side of the --> indicates a start time of the paragraph and the right side of the --> indicates a stop time of the paragraph, down to milliseconds; and the third line is the source text of the paragraph.

The present invention has the beneficial effects that: the method disclosed by the present invention is a method of how to accurately match an audio having been translated into a text and a timeline through an algorithm with a source text. the timeline of the audio may be further mapped to the source text by matching the text translated from the audio with the source text, which ensures the accuracy of the texts

and may guarantee that the source text has the function of the timeline; and text translation may be performed by use of an open-source model and without using high-precision models, which reduces training samples and time for model training and a great amount of resources, and saves the cost of development.

BRIEF DESCRIPTION OF THE DRAWINGS

FIG. 1 is an overall flowchart of the method according to an embodiment of the present invention; and

FIG. 2 is a specific flowchart of step S5 in the method according to the embodiment of the present invention.

DETAILED DESCRIPTION OF THE EMBODIMENTS

The following further describes the present invention with reference to the accompanying drawings, and it should be noted that detailed implementations and specific operation processes are given in the embodiment based on the technical solution, but the scope of protection of the present invention is not limited to the embodiment.

The embodiment discloses an audio and subtitle synchronization method based on audio translation and similarity calculation, in which texts translated by a model are matched with a source text of an audio in a mode of combining the model with matching calculation, so that the accuracy of the texts is ensured, and resources required for model training may be reduced. According to the method in the embodiment, the start and stop positions of each paragraph of a source text in a translated text will be determined by matching the source text with the translated text in paragraphs, in such a way, a timeline range of the source text in the audio may be obtained, and then a purpose of matching the source text with the corresponding timeline in the audio is achieved.

As shown in FIG. 1, the audio and subtitle synchronization method based on audio translation and similarity calculation is specifically implemented as follows:

S1: acquisition of a source text of a novel and audio files of the novel:

the source text of the novel is acquired through a crawler technology, and the whole novel is stored as a unit; and the audio files of the novel locally are downloaded through a database and stored by chapters;

S2: data cleaning and conversion:

the source text of the novel contains various symbols and HTML codes, thus, it is necessary to carry out data cleaning on the source text of the novel so as to process the source text of the novel into required text contents; and the audio files of the novel are required to be converted to wav format files of 18000Hz;

S3: a trained speech-to-text model and a word correction model are downloaded; wherein the speech-to-text model adopts a STT model (speech-to-text model is an open-source model); and an input of the STT model is the audio file of each chapter

of the novel processed in step S2, and outputs thereof are each character of each chapter obtained by translation and a corresponding timestamp of each character in an audio; and

since the outputs of the STT models are expressed in characters, it is also necessary to concatenate the words, i.e., concatenate output characters into corresponding words, and obtain a corresponding time in the audio for each word; then, the words concatenated by use of the word correction model are verified to ensure the accuracy of the words; and the words of each chapter obtained by translation in step S3 are concatenated so as to obtain a translated text of each chapter;

S4: article dividing:

the source text of the novel processed in step S2 by chapters is divided, so as to obtain a source text of each chapter; and

S5: sentence matching:

the source text of each chapter of the novel in paragraphs is further divided, area matching and probability calculation are performed on the obtained texts and the translated texts of corresponding chapters to determine a stop position of each paragraph of the source text in the translated text, and a corresponding time of the stop position of each paragraph of the source text in the audio is acquired according to a corresponding time of the word at the stop position in the audio. As shown in FIG. 2, step S5 is specifically implemented as follows:

S5.1: the source text of each chapter of the novel is divided into paragraphs with a full stop respectively; moreover, because the translated text only contains commas and full stops, it is necessary to delete the punctuation marks except commas and full stops, such as ?, " and * and the like, in the paragraphs obtained from the source text of the novel by dividing; in addition, if a paragraph obtained by dividing contains more than 20 words, the paragraph will be divided once again with commas to ensure that the number of words in each paragraph obtained by dividing do not exceed 20, thereby guaranteeing the matching accuracy better;

S5.2: initialization:

all the paragraphs obtained by the division of the source text of each chapter of the novel are stored in a list L in order, the list L is traversed and the length of each paragraph in the list L is determined, and that each paragraph of one chapter is within an interval range of the translated text of a corresponding chapter is preliminarily determined according to the length of each paragraph.

S5.3: determination of range windows:

the translated text of the novel is expressed in chapters, and due to the influence of various factors, the accuracies of translation are also different. For example, a word is translated into two or more words, and punctuation marks such as commas are translated into full stops, and the like. Thus, a mode of carrying out one-by-one matching on words of the source text and translated words may have serious biases,

which causes that the final text is correct but cannot match with the timeline of the audio accurately. Therefore, according to the method in the embodiment, paragraphs obtained by dividing the source text of each chapter of the novel are respectively matched with the translated text of each chapter in paragraphs to find out the position of the stop word of each paragraph of the source text of the novel in the translated text, in such a way, the correspondence between the source text of the novel and the audio timeline of the novel may be accurately obtained, and then that the timeline of each paragraph can be accurately output is ensured.

In the embodiment, an interval is determined first to preliminarily screen out the position of the stop word of each paragraph, and then the accurate position of the stop word of one paragraph of the source text of the novel in the translated text is found out by means of probability matching, so that the overall time range of the paragraph can be calculated through the timeline of the stop word of the paragraph.

a start position of a window is determined: the position of a stop word of a previous paragraph + the length of a current paragraph - a window spacing = the start position of the window; where the window spacing is the number of words in the window;

the stop position of the window is determined: the position of the stop word in the previous paragraph + the length of the current paragraph + the window spacing = the stop position of the window;

a range window is defined as A1 by the start position and stop position of the window; in the embodiment, the window spacing is chosen to be 8.

S5.3: post-paragraph positioning:

words in the range window A1 are determined, then the last three words of the current paragraph are selected to generate a range window A2, and at the same time, a sliding window with a spacing of 3 is generated in the range window A1, the words in the range window A1 are slid to respectively compare with the words in the range window A2, the probability of intersection of the two range windows is calculated, and if the probability exceeds 2/3, the stop word of the paragraph is considered to be within the sliding window of the range window A1.

However, in the process of continuous experiments, it is found that there may be a word order deviation in A1 and A2 while the probability exceeds 2/3, for example, A1 is ['to', 'up', 'data'], and A2 is ['up', 'to', 'date'], although the probability is 2/3, it is obvious that the two sentences cannot match.

Therefore, the method in this embodiment is amended based on the above solution as follows:

determining words in the range window A1, then selecting the last three words of the current paragraph to generate a range window A2, and at the same time, generating a sliding window with a spacing of 3 in the range window A1, and sliding the words in the range window A1 to respectively compare with the words in the range window A2; during the process of comparison, initializing three variables i, j and t to

respectively record a pointer position of the sliding window of the range window A1, a pointer position of the range window A2 and a counter which are all 0 initially; then, traversing the range window A2 and comparing the word at a position j in the range window A2 with the word at a position i in the sliding window A1, if the two words are the same through comparison, determining that the counter t is equal to t+1, the pointer position i of the sliding window of the range window A1 is equal to i+1 and the pointer position j of the range window A2 is equal to j+1, otherwise, determining that the pointer position j of the range window A2 is equal to j+1 and the rest is unchanged; carrying out comparison repeatedly until the pointer position j is equal to the length of the range window A2, and ending the circulation; and finally, if the probability that the word in the sliding window of the range window A1 is the same as the word in the range window A2 is equal to 2/3, considering that the stop word of the current paragraph is within the sliding window of the range window A1.

How to determine the position of the stop word of one paragraph in the sliding window is specific as follows:

the stop word should be determined according to the matching of the words in the sliding window and those in the range window A2, if the same two words in the sliding window of the range window A1 and the range window A2 are the last two words in the sliding window of the range window A1 and the first two words in the range window A2, the stop word is the next word of the last word in the sliding window of the range window A1; if the same two words in the sliding window of the range window A1 and the range window A2 are the first two words in the sliding window of the range window A1 and the first two words in the range window A2, alternatively, the last two words in the sliding window of the range window A1 and the last two words in the range window A2, the stop word is the last word in the sliding window of the range window A1; and if the same two words in the sliding window of the range window A1 and the range window A2 are the first two words in the sliding window of the range window A1 and the last two words in the range window A2, the stop word is the second word in the sliding window of the range window A1;

the above three cases are illustrated by examples as follows:

first, if the words in the sliding window of A1 are ['a', 'up', 'to'], and the words in the window A2 are ['up', 'to', 'date'], the stop word is the next word of 'to' in the range window A1;

second, if the words in the sliding window of A1 are ['up', 'to', 'data'], and the words in the window A2 are ['up', 'to', 'date'], the stop position is the position of 'data' in the range window A1; and

third, if the words in the sliding window of A1 are ['up', 'to', 'data'], and the words in the window A2 are ['a', 'up', 'to'], the stop position is the position of 'to' in the range window A1;

S5.4: pre-paragraph positioning:

if in the post-paragraph positioning in S5.3, the stop word of the current paragraph fails to be found in the range window A1, pre-paragraph positioning is performed, i.e., the next paragraph of the current paragraph is selected, and the first three words of the next paragraph of the current paragraph is taken as a range window A3, wherein the spacing of the sliding window in the range window A1 is also 3; and

three variables i, j and t are initialized to respectively record the pointer position of the sliding window of the range window A1, a pointer position of the range window A3 and the counter, which are all 0 initially; then, the range window A3 is traversed and a word at a position j in the range window A3 is compared with the word at the position i in the sliding window A1, if the two words are the same through comparison, the counter t is equal to t+1, the pointer position i of the sliding window of the range window A1 is equal to i+1 and the pointer position j of the range window A3 is equal to j+1 is determined, otherwise, the pointer position j of the range window A3 is equal to j+1 and the rest is unchanged; comparison is performed repeatedly until the pointer position j is equal to the length of the range window A3, and the circulation is ended; the words of the two windows are circulated continuously until the window with a probability equal to 2/3 is found, then the position pointer of the first word in the sliding window of the range window A1 is determined, and after the obtained value minus 1, the stop word of the current paragraph is found;

step S5.5: paragraph merging:

if in the pre-paragraph positioning in S5.4, the stop word of the current paragraph also fails to be found in the range window A1, it means that the translated text has a large error for the current paragraph and the stop word cannot be found through probability comparison, in this case, the current paragraph and the next paragraph are directly merged, and the post-paragraph positioning and the pre-positioning are circulated until a stop word of the paragraph satisfying the conditions is found; and

step S5.6: result outputting:

result outputting is performed once the stop word of each paragraph is found, with a format as follows:

3

00:00:10,660 --> 00:00:14,840

She rummaged through her basket, holding up the tin, and I nodded.

The result outputting consists of three parts, each of which is separated by a blank line, the first part is a paragraph number starting from 0 and indicating to which paragraph of the chapter is belonged; the second line is a timeline of the paragraph, the left side of the --> indicates a start time of the paragraph and the right side of the --> indicates a stop time of the paragraph, down to milliseconds; and the third line is the source text of the paragraph. The stop time of the paragraph is a time corresponding to the stop word of the paragraph.

For those skilled in the art, various corresponding changes and deformations can be

given according to the above technical solutions and conceptions, and all these changes and deformations should be included in the scope of protection of the claims of the present invention.

CLAIMS

1. An audio and subtitle synchronization method based on audio translation and similarity calculation, comprising the following specific steps:

S1: acquisition of a source text of a novel and audio files of the novel:

acquiring the source text of the novel through a crawler technology, and storing the whole novel as a unit; and downloading the audio files of the novel locally through a database and storing the audio files by chapters;

S2: data cleaning and conversion:

carrying out data cleaning on the source text of the novel so as to process the source text of the novel into a required text content; and converting the audio files of the novel into wav-format files of 18000Hz;

S3: downloading a trained speech-to-text model and a word correction model; where the speech-to-text model adopts a STT model; and an input of the STT model is the audio file of each chapter of the novel processed in step S2, and outputs thereof are each character of each chapter obtained by translation and a corresponding timestamp of each character in an audio; and

concatenating output characters into corresponding words, and obtaining a corresponding time in the audio for each word; then, verifying the words concatenated by use of the word correction model to ensure the accuracy of the words; and concatenating the words of each chapter obtained by translation in step S3 so as to obtain a translated text of each chapter;

S4: article dividing:

dividing the source text of the novel processed in step S2 by chapters, so as to obtain a source text of each chapter; and

S5: sentence matching:

further dividing the source text of each chapter of the novel in paragraphs, carrying out area matching and probability calculation on the obtained texts and the translated texts of corresponding chapters to determine a stop position of each paragraph of the source text in the translated text, acquiring a corresponding time of the stop position of each paragraph of the source text in the audio according to a corresponding time of the word at the stop position in the audio, and finally, outputting a result of timeline correspondence between the source text and audio of the novel.

2. The method according to Claim 1, wherein step S5 is specifically implemented as follows:

S5.1: dividing the source text of each chapter of the novel into paragraphs respectively with a full stop; where because the translated text only contains commas and full stops, it is necessary to delete punctuation marks except commas and full stops in the paragraphs obtained by the division of the source text of the novel;

S5.2: initialization:

storing all the paragraphs obtained by the division of the source text of each chapter of the novel in a list L in order, traversing the list L, determining the length of each paragraph in the list L, and preliminarily determining that each paragraph of one chapter is within an interval range of the translated text of a corresponding chapter according to the length of each paragraph;

S5.3: determination of range windows:

determining a start position of a window: the position of a stop word of a previous paragraph + the length of a current paragraph - a window spacing = the start position of the window; where the window spacing is the number of words in the window;

determining a stop position of the window: the position of the stop word in the previous paragraph + the length of the current paragraph + the window spacing = the stop position of the window; and

defining a range window by the start position and stop position of the window as A1;

S5.3: post-paragraph positioning:

determining words in the range window A1, then selecting the last three words of the current paragraph to generate a range window A2, and at the same time, generating a sliding window with a spacing of 3 in the range window A1, and sliding the words in the range window A1 to respectively compare with the words in the range window A2; during the process of comparison, initializing three variables i, j and t to respectively record a pointer position of the sliding window of the range window A1, a pointer position of the range window A2 and a counter which are all 0 initially; then, traversing the range window A2 and comparing the word at a position j in the range window A2 with the word at a position i in the sliding window A1, if the two words are the same through comparison, determining that the counter t is equal to t+1, the pointer position i of the sliding window of the range window A1 is equal to i+1 and the pointer position j of the range window A2 is equal to j+1, otherwise, determining that the pointer position j of the range window A2 is equal to j+1 and the rest is unchanged; carrying out comparison repeatedly until the pointer position j is equal to the length of the range window A2, and ending the circulation; and finally, if the probability that the word in the sliding window of the range window A1 is the same as the word in the range window A2 is equal to 2/3, considering that the stop word of the current paragraph is within the sliding window of the range window A1;

where, the stop word should be determined according to the matching of the words in the sliding window of the range window A1 and the words in the range window A2, if the same two words in the sliding window of the range window A1 and the range window A2 are the last two words in the sliding window of the range window A1 and the first two words in the range window A2, the stop word is the next word of the last word in the sliding window of the range window A1; if the same two words in the sliding window of the range window A1 and the range window A2 are the first two words in the sliding window of the range window A1 and the first two words in the

range window A2, alternatively, the last two words in the sliding window of the range window A1 and the last two words in the range window A2, the stop word is the last word in the sliding window of the range window A1;

and if the same two words in the sliding window of the range window A1 and the range window A2 are the first two words in the sliding window of the range window A1 and the last two words in the range window A2, the stop word is the second word in the sliding window of the range window A1;

S5.4: pre-paragraph positioning:

if in the post-paragraph positioning in S5.3, the stop word of the current paragraph fails to be found in the range window A1, carrying out pre-paragraph positioning, i.e., selecting the next paragraph of the current paragraph, and taking the first three words of the next paragraph of the current paragraph as a range window A3, where the spacing of the sliding window in the range window A1 is also 3; and

initializing three variables i, j and t to respectively record the pointer position of the sliding window of the range window A1, a pointer position of the range window A3 and the counter, which are all 0 initially; then, traversing the range window A3 and comparing a word at a position j in the range window A3 with the word at the position i in the sliding window A1, if the two words are the same through comparison, determining that the counter t is equal to t+1, the pointer position i of the sliding window of the range window A1 is equal to i+1 and the pointer position j of the range window A3 is equal to j+1, otherwise, determining that the pointer position j of the range window A3 is equal to j+1 and the rest is unchanged; carrying out comparison repeatedly until the pointer position j is equal to the length of the range window A3, and ending the circulation; continuously circulating the words of the two windows until the window with a probability equal to 2/3 is found, then determining the position pointer of the first word in the sliding window of the range window A1, and allowing the obtained value to minus 1 so as to find the stop word of the current paragraph;

step S5.5: paragraph merging:

if in the pre-paragraph positioning in S5.4, the stop word of the current paragraph also fails to be found in the range window A1, meaning that the translated text has a large error for the current paragraph and the stop word cannot be found through probability comparison, in this case, directly merging the two paragraphs, and circulating the post-paragraph positioning and the pre-positioning until a stop word of the paragraph satisfying the conditions is found; and

step S5.6: result outputting:

carrying out result outputting once the stop word of each paragraph is found.

3. The method according to Claim 2, wherein in step S5.1, if a paragraph obtained by dividing contains more than 20 words, the paragraph is divided once again with commas to ensure that the number of words in each paragraph obtained by dividing do not exceed 20.

4. The method according to Claim 2, wherein in step S5.3, the window spacing is chosen to be 8.

5. The method according to Claim 2, wherein in step S5.6, the format of the result outputting as follows:

paragraph number

start time of the paragraph --> stop time of the paragraph

source text of the paragraph

the paragraph number starts at 0 and indicates to which paragraph of the chapter is belonged; the second line is a timeline of the paragraph, the left side of the --> indicates a start time of the paragraph and the right side of the --> indicates a stop time of the paragraph, down to milliseconds; and the third line is the source text of the paragraph.