Luk3Yao's Neo/vim Config

Lean mean Neo/vim machine, 30-45ms startup time.

Best with Neovim or Vim 8 with +python3 extensions enabled.

screen.png

Features

- Fast startup time
- Robust, yet light-weight
- Lazy-load 95% of plugins with [Shougo/dein.vim]
- Custom side-menu (try it out! Leader+l)
- Modular configuration
- Denite (Unite's successor) centric work-flow
- Extensive Deoplete setup (auto-completion)
- Lightweight simple status/tabline
- Easy customizable theme
- Premium color-schemes
- Central location for tags

Pre-requisites

- Python 3 (`brew install python`)

- Python 2 (`brew install python@2`)

- Neovim (Optional, `brew install neovim`)

- virtualenv for both python3 and python2:

  ```
  pip2 install virtualenv
  pip3 install virtualenv
  ```

- Manually config the python3 path but not using detecting automatically to accelerate the startup

- Rust & Cargo (to accelerate the icon render)

  ```
  curl https://sh.rustup.rs -sSf | sh # install the Rust and cargo at the same time
  cargo install bat | brew install bat # preview
  cargo install devicon-lookup # faster render the file notation
  ```

- For coc modules(if you are using yarn to install):

  ```
  npm install -g yarn
  yarn config set https-proxy http://localhost:3128
  ```

  and then exe follow in ex-mode:

  ```
  :call coc#util#build()
  ```

Recommended Tools

- ag (The Silver Searcher): ggreer/the_silver_searcher
- z (jump around): rupa/z
- Universal ctags: ctags.io
- Fuzzy file finders: fzf, fzy, or peco
- Tern: `npm -g install tern`
- bat
- rg
- issw: auto switch input sources on mac os

Structure

- config/ - Configuration
    - basic - Basic config about Behaviour, UI etc.
    - dein - dein config yaml
    - plugins/ - Plugin configurations
    - vimrc - Initialization
- ftplugin/ - Language specific custom settings
- plugin/ - Customized small plugins
- snippets/ - Personal code snippets

Plugin Highlights

- Package management with caching enabled and lazy loading
- Project-aware tabs and label
- Completion for Go, Python, Naive Java, C etc. supported by coc.nvim

Note that 95% of the plugins are **[lazy-loaded]**.

Behaviour

- issw: auto switch input sources on mac os

switch to ABC input in normal mode automatically by following

It's comfortable with markdown-preview.nvim and cVim while cjk mode is used.

```
" auto switch im in macos
" com.apple.keylayout.ABC
" com.baidu.inputmethod.BaiduIM.pinyin
" com.baidu.inputmethod.BaiduIM.wubi
au InsertEnter * call system('issw com.baidu.inputmethod.BaiduIM.wubi')
au InsertLeave * call system('issw com.apple.keylayout.ABC')
```

Plugins

StartUp

- dein - 󠀠 Dark powered Vim/Neovim plugin manager
- spaceline.vim - vim statusline like spacemacs
- vim-buffet - IDE-like Vim tabline

Markdown

- vim-markdown - Markdown Vim Mode http://plasticboy.com/markdown-vim-mode/
- markdown-preview.nvim - markdown preview plugin for (neo)vim
- tagbar-markdown - vim-tagbar extension for markdown
- vim-emoji - 😀 Emoji in Vim

Keybindings

KeyMap

- Default leader SPACE localleader; ,The vim keymap in `mappings.vim`,Plugins keymap in `core/plugins/allkey.vim`
- Insertkey like emacs
- if you find the keymap doesn't work , Or the keymap doesn't show in this keylist but in `allkey.vim` ,you can create a issue. Welcome PR.

Keys	Mode	Description
Ctrl + w	Insert	delete the whole word

Keys	Mode	Description
Ctrl + h	Insert	delete BackSpace
Ctrl + d	Insert	delete a single char
Ctrl + u	Insert	delete to the beginning of the line
Ctrl + b	Insert	move left
Ctrl + f	Insert	move right
Ctrl + a	Insert	move the head of line
Ctrl + e	Insert	move the end of line
Ctrl + p	Command	move up
Ctrl + b	Command	move left
Ctrl + f	Command	move right
Ctrl + a	Command	move the head of line
Ctrl + e	Command	move the end of line
Ctrl + d	Command	move the cursors char
Ctrl + h	Command	delete BackSpace
Ctrl + S	Insert	save
Ctrl + Q	Insert	save and exit
Ctrl + o	Insert	create a new line
] + b	Normal	pre buffer
[+ b	Normal	next buffer
Ctrl + x	Normal	delete buffer
Y	Normal	copy utils the end
j	Normal	gj
k	Normal	gk
j	Visual	gj
k	Visual	gj
-	Normal	ChooseWin
, + Space	Normal	Delete trailing spaces
[+ a	Normal	Ale next
] + a	Normal	Ale pre
[+ c	Normal	coc next
] + c	Normal	coc prev
] +]	Normal	jump to prev go func
[+ [	Normal	jump to next go func
g + c	Normal	caw.vim comment
g + c + c	Normal	caw.vim uncomment
g + a	Normal	EasyAlign
g + a	Visual	EasyAlign
s + a	Normal	vim-sandwich prefix +{iw/is/ip}+{i/f/t}
s + d	Normal	vim-sandwich prefix +{iw/is/ip}+{i/f/t}
s + r	Normal	vim-sandwich prefix +{iw/is/ip}+{i/f/t}
s + d + b	Normal	vim-sandwich prefix +{iw/is/ip}+{i/f/t}
s + r + b	Normal	vim-sandwich prefix +{iw/is/ip}+{i/f/t}
v	Visual	vim-expand-region
V	Visual	vim-expand-region
z + l	Normal	Denite Outline for golang
z + T	Normal	Denite Goroot for golang

Keys	Mode	Description
Ctrl + h	Normal	move to left window
Ctrl + l	Normal	move to right window
Ctrl + j	Normal	move to down window
Ctrl + k	Normal	move to up window
Ctrl + w + [	Normal	the width of window reduce 3
Ctrl + w +]	Normal	the width of window increase 3
Ctrl + f	Normal	Window scrolling down
Ctrl + b	Normal	Window scrolling up
Ctrl + d	Normal	Window scrolls down half a screen

Keys	Mode	Description
Ctrl + u	Normal	Window scrolls up half a screen

Keys	Mode	Description
Leader + t + n	Normal	create tab
Leader + t + e	Normal	edit tab
Leader + t + m	Normal	move tab

Keys	Mode	Description
Leader + number	Normal	number[0-9] choose buffer
Leader + c + a	Normal	CocDiagnostics
Leader + c + c	Normal	CocCommands
Leader + c + e	Normal	CocExtensions
Leader + c + j	Normal	Cocnext
Leader + c + k	Normal	CocPrev
Leader + c + o	Normal	CocOutline
Leader + c + r	Normal	CocResume
Leader + c + s	Normal	CocIsymbols
Leader + G	Normal	open Goyo
Leader + m	Normal	open mundo
Leader + w	Normal	save file
Leader + -	Normal	ChooseWin
Leader + s	Normal	open Startify
Leader + d	Normal	open Dash
Leader + p	Normal	Edit the file which in plugins folder
Leader + c + f	Normal	coc Format
Leader + f + c	Normal	fzf chage vim colorscheme
Leader + f + f	Normal	fzf find files
Leader + f + r	Normal	fzf rg search
Leader + f + v	Normal	fzf Vista Finder
Leader + f + b	Normal	fzf find buffer
Leader + f + w	Normal	fzf rg search the cursor word
Leader + f + g + c	Normal	fzf gitcommits
Leader + f + b + c	Normal	fzf gitbcommits
Leader + Leader + w	Normal	Easymotion jump to donw
Leader + Leader + b	Normal	Easymotion jump to up
Leader + Leader + f	Normal	Easymotion search for jump
Leader + g + d	Normal	Gdiff
Leader + g + c	Normal	Gcommit
Leader + g + b	Normal	Gblame
Leader + g + B	Normal	GBrowse
Leader + g + S	Normal	GStatus
Leader + g + p	Normal	Git push
Leader + g + l	Normal	Denite show all git log
Leader + g + h	Normal	Denite show git branch

Keys	Mode	Description
LocalLeader + t	Normal	Open Tagbar
LocalLeader + r	Normal	QuickRun
LocalLeader + v	Normal	Open Vista
LocalLeader + gi	Normal	vim-go GoImpl
LocalLeader + gd	Normal	vim-go GoDescribe
LocalLeader + gc	Normal	vim-go GoCallees
LocalLeader + gC	Normal	vim-go GoCallers

Keys	Mode	Description
LocalLeader + gs	Normal	vim-go GoCallstack

Keys	Mode	Description
LocalLeader + m	Normal	Denite Menu
<CR>	Denite	Doaction
i	Denite	Open filter buffer
d	Denite	Delete
p	Denite	Preview
s + t	Denite	tabopen
s + v	Denite	Vsplit
s + i	Denite	Split
'	Denite	QuickMove
q	Denite	Quit
r	Denite	RedRaw
y + y	Denite	Yank
Esc	Denite	Quit
Ctrl + u	Denite	Restore Source
Ctrl + f	Denite	call Defx
Ctrl + x	Denite	Choose Action
Space	Denite	Toggle Select
Esc	Denite Filter	Quit
q	Denite Filter	Quit
Ctrl + c	Denite Filter	Quit filter
k + k	Denite Filter insert	move window
k + k	Denite Filter normal	move window
j + j	Denite Filter insert	move window
j + j	Denite Filter normal	move window

Keys	Mode	Description
C-X	Normal	tab split
C-v	Normal	split
C-t	Normal	vsplit
C-d	Normal	pagedown
C-u	Normal	pageup

Keys	Mode	Description
Leader + e	Normal	open defx
N	Defx	touch new file
K	Defx	touch new folder
<CR>	Defx	open file or folder
l	Defx	open file or folder
d + d	Defx	delete (Send2Trash required)
r	Defx	rename
.	Defx	show ignore files
h	Defx	Return to the parent directory
q	Defx	exit defx
s	Defx	Open in rightwindow vs
i	Defx	Open file sp
y + y	Defx	copy the path of file
</kbd>	Defx	show the path of file
c	Defx	copy
m	Defx	move
p	Defx	paste

Keys	Mode	Description
Space	Defx	multi
g + f	Defx	open in denite
g + r	Defx	search in denite
w	Defx	increase the width

Keys	Mode	Description
Tab	Select	Jump
Tab	Insert	completion donw
Shit + Tab	Insert	completion up
Ctrl + n	Insert	completion down
Ctrl + p	Insert	completion up
Ctrl + Space	Normal	Forced trigger completion
<CR>	Insert/Select	Confirm
K	Normal	Show the function doc
g + d	Normal	jump to defined
g + y	Normal	coc-type-definition
g + i	Normal	coc-implementation
g + r	Normal	coc-references
[+ g	Normal	coc-git-prechunk
] + g	Normal	coc-git-nextchunk

ToDo

☐ Read the flow source to simplify and speed up
☐ Skilled more in Vim Script

Credits

Big thanks to the big guys!

- rafi
- taigacute